www.modaresanesharif.ac.ir



# فصل اول

# «پیچیدگی زمانی الگوریتم‌ها»

## تست‌های طبقه‌بندی شده کنکوری فصل اول

### «نمادهای مجانبی»

۱ـ گزینهٔ درست را انتخاب کنید. (علوم کامپیوتر ـ سراسری ۸۰)

۱) $f(n)\in O(g(n)), h(n)\in \Omega(g(n)) \Rightarrow f(n)\in \theta(h(n))$

۲) $f(n)\in \theta(g(n)), g(n)\in O(h(n)) \Rightarrow h(n)\in O(f(n))$

۳) $f(n)\in \Omega(g(n)), h(n)\in O(g(n)) \Rightarrow f(n)\in \theta(g(n))$

۴) $f(n)\in \Omega(g(n)), g(n)\in \theta(h(n)) \Rightarrow f(n)\in \Omega(h(n))$

۲ـ گزینه صحیح را انتخاب کنید. (علوم کامپیوتر ـ سراسری ۸۰)

۱) $n^{1/10}\in \Omega(\log n)$

۲) $\log(n!)\in O(\log n)$

۳) $8n^2+3n-4\in O(n\log n)$

۴) $1^n+2^n+\cdots+n^n\in O(n^n)$

۳ـ کدام یک از عبارات زیر غلط است؟ (مهندسی کامپیوتر ـ سراسری ۸۱)

۱) $10^n+n^{20}\notin \theta(n^n)$

۲) $\log_2^n\in \theta(\log_{10} n)$

۳) $(\log_2 n)!\in \Omega(n!)$

۴) $4n^3+5n^2+7n\in \Omega(\log_2 n)$

۴ـ فرض کنید $f$ و $g$، دو تابع دلخواه به شکل $f,g:N\to R^+$ باشند. بعلاوه فرض کنید: $\lim_{n\to\infty}\frac{f(n)}{g(n)}=+\infty$. کدام یک از گزینه‌های زیر درست است؟ (مهندسی کامپیوتر ـ سراسری ۸۲)

۱) $f(n)\in O(g(n))\ ,\ g(n)\notin \Omega(f(n))$

۲) $g(n)\in O(f(n))\ ,\ f(n)\in O(g(n))$

۳) $f(n)\in \Omega(g(n))\ ,\ f(n)\notin \theta(g(n))$

۴) $f(n)\in \theta(g(n))\ ,\ g(n)\notin \Omega(f(n))$

۵ـ کدام عبارت صحیح است؟ (علوم کامپیوتر ـ سراسری ۸۲)

۱) $(n+1)(n^2-2n+1)\in O(2^n)$

۲) $(n+1)(n^2-2n+1)\in \theta(n)$

۳) $(n+1)(n^2-2n+1)\in \Omega(n^4)$

۴) $(n+1)(n^2-2n+1)\in O(n^2\log n)$

۶ـ کدام گزینه صحیح است؟ (علوم کامپیوتر ـ سراسری ۸۲)

۱) $n^a=O((\log n)^b)$ اگر $a>b$

۲) $n^a\neq O((\log n)^b)$ اگر $a>b$

۳) $n^a=O((\log n)^b)\forall b\ ,\ a>0$

۴) $n^a\neq O((\log n)^b)\forall b\ ,\ a>0$

۷ـ فرض کنید $f(n)=O(g(n))$ کدام‌یک از گزینه‌های زیر برای هر تابع با مقادیر مثبت مانند $f(n)$ ، $g(n)$ برقرار است؟ (مهندسی کامپیوتر ـ سراسری ۸۳)

۱) $g(n)=\theta(f(n))$

۲) $2^{f(n)}=O(2^{g(n)})$

۳) $2^{g(n)}=\Omega(2^{f(n)})$

۴) $\log(g(n))=O(\log(f(n)))$ با فرض $\log(g(n))\geq 1$

۸ـ کدام یک از گزینه‌های زیر صحیح است؟ (مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۴)

۱) $n^2\sin n\in \Omega(n)$

۲) $n^2\sin n\in O(n)$

۳) $n\in \Theta(n^2\sin n)$

۴) $n\in \Omega(n^2\sin n)$

۹ـ کدام گزینه صحیح می‌باشد؟ (علوم کامپیوتر ـ سراسری ۸۴)

۱) $0<\varepsilon<0.1\quad n^3\log n=O(n^{3+\varepsilon})$

۲) $\sqrt{n}=O(\log n)$

۳) $0<\varepsilon<0.1\quad n^{1+\varepsilon}=O(n\log n)$

۴) $n^2=O(\frac{n^2}{\log n})$

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



۱۰ـ توابع $f(n)=4^{\log n}$ و $g(n)=(\log n)^{\log n}$ و $h(n)=\log^2 n$ را در نظر می‌گیریم، کدام یک از گزینه‌های زیر صحیح است؟ (مهندسی کامپیوتر ـ سراسری ۸۵)

۱) $f(n)\in O(g(n)), f(n)\in \Omega(h(n))$

۲) $g(n)\in \Omega(h(n))\ ,\ h(n)\in \Omega(f(n))$

۳) $f(n)\in \theta(h(n))\ ,\ g(n)\in \Omega(f(n))$

۴) $h(n)\in O(g(n))\ ,\ f(n)\in \theta(g(n))$

۱۱ـ کدام یک از عبارات زیر درست‌اند: (مهندسی کامپیوتر ـ سراسری ۸۵)

I. $e^{c\sqrt{n}}=O(e^{\sqrt{n}})\ \ c\geq 1$

II. $n^2=O(n\lg n)$

III. $n=O(n\lg n)$

۱) فقط II

۲) فقط III

۳) II,I

۴) III,I

۱۲ـ کدام یک از عبارت‌های زیر نادرست است؟ (مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۵)

۱) $O(n!)=O(n^n)$

۲) $O(10^6)<O(n)<O(n.\log_2 n)$

۳) هر الگوریتم از مرتبه $\theta(n)$، از مرتبه $O(n^2)$ نیز هست.

۴) حذف عنصر آخر در یک لیست زنجیره‌ای یک طرفه، با داشتن اشاره‌گرهای last,first از مرتبه $O(1)$ است.

۱۳ـ در رشد توابع زیر کدام ترتیب صحیح می‌باشد؟ (علوم کامپیوتر ـ سراسری ۸۵)

۱) $O(n\log n), O(1+\varepsilon)^n, O(\frac{n^2}{\log n})$

۲) $O((1+\varepsilon)^n, O(n\log n)^n, O(\frac{n^2}{\log n})$

۳) $O(\frac{n^2}{\log n}), O(n\log n), O(1+\varepsilon)^n$

۴) $O(n\log n), O(\frac{n^2}{\log n}), O(1+\varepsilon)^n$

۱۴ـ کدام‌یک از تساوی‌های زیر درست است؟ (مهندسی کامپیوتر ـ آزاد ۸۵)

۱) $n^2\log n=\theta(n^2)$

۲) $6n^3/(\log n+1)=O(n^3)$

۳) $n^2/\log n=\theta(n^2)$

۴) $n^3 2^n+6n^2 3^n=O(n^3 2^n)$

۱۵ـ کامپیوتری در واحد زمان مسئله‌ای به اندازه 16 را که الگوریتم آن از مرتبه زمانی $n2^n$ است حل می‌کند. اگر سرعت کامپیوتر 131072 برابر گردد این کامپیوتر همان مسئله را با چه اندازه‌ای در واحد زمان حل خواهد کرد؟ (مهندسی کامپیوتر ـ سراسری ۸۶)

۱) $16+17+\log 17$

۲) $16\times 17\times \log 17$

۳) 32

۴) $16+\log 131072$

۱۶ـ کدام یک از موارد زیر نادرست است؟ (مهندسی کامپیوتر ـ آزاد ۸۶)

۱) $6*2^n+n^2=\theta(n^2)$

۲) $6*2^n+n^2=0(2^n)$

۳) $6*2^n+n^2=\Omega(2^n)$

۴) $6*2^n+n^2=\Omega(n^2)$

۱۷ـ پیچیدگی زمانی دو الگوریتم A و B به‌ترتیب برابر $A(n)=\frac{n^2}{50}$ و $B(n)=20n$ می‌باشد. حداکثر به ازای کدام n استفاده از پیچیدگی زمانی $A(n)$ مناسب‌تر است؟ (مهندسی کامپیوتر ـ آزاد ۸۷)

۱) $n<=500$

۲) $n<=100$

۳) $n<=1000$

۴) $n<=250$

۱۸ـ پیچیدگی زمانی دو الگوریتم A و B به‌ترتیب برابر $A(n)=\frac{n^2}{4}$ و $B(n)=25n$ می‌باشد، حداکثر به ازای کدام n استفاده از پیچیدگی زمانی $A(n)$ کاراتر خواهد بود؟ (مهندسی کامپیوتر ـ آزاد ۸۷)

۱) $n<=50$

۲) $n<=120$

۳) $n<=100$

۴) $n<=150$

۱۹ـ با فرض مثبت بودن توابع f و g، تعداد گزاره‌های درست از میان گزاره‌های زیر چند است؟ (مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۸)

گزاره ۱: $f(n)=O((f(n))^2)$، گزاره ۲: $f(n)+O(f(n))=\theta(f(n))$، گزاره ۳: $f(n)=O(g(n))\Rightarrow 2^{f(n)}=O(2^{g(n)})$، گزاره ۴: $f(n)=\theta(f(n/2))$

۱) یک گزاره صحیح است.

۲) دو گزاره صحیح است.

۳) سه گزاره صحیح است.

۴) گزاره‌ای صحیح نیست.

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۲۰ـ** توابع $f(n)=4^{\lg n}, g(n)=\lg^{\lg n} n$ و $h(n)=\lg^2 n$ را در نظر بگیرید. کدام یک از گزاره‌های زیر صحیح است؟ (مهندسی کامپیوتر ـ سراسری ۸۹)

۱) $f(n)\in O(g(n)), f(n)\in \Omega(h(n))$

۲) $f(n)\in \Theta(h(n)), g(n)\in \Omega(f(n))$

۳) $g(n)\in \Omega(h(n)), h(n)\in \Omega(f(n))$

۴) $h(n)\in O(g(n)), f(n)\in \Theta(g(n))$

**۲۱ـ** کدام‌یک از گزاره‌های زیر غلط است؟ (مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۹)

۱) $f(n)+O(f(n))=\Theta(f(n))$

۲) $\Omega(f(n))+O(f(n))=\Theta(f(n))$

۳) $g(n)=\Omega(f(n))\Rightarrow f(n)=O(g(n))$

۴) $g(n)=\Omega(f(n))\Rightarrow g(n)=\Omega(O(f(n))$

**۲۲ـ** کدام گزینه صحیح است؟ (علوم کامپیوتر ـ سراسری ۸۹)

۱) $\frac{n}{\lg n}=O(n^{1-x})\ ,\ 3^n=\Omega(n2^n)$

۲) $n(\lg n)^5=\Omega(n^{\frac{1}{2}})\ ,\ \sqrt{n}=O(\lg n)^5$

۳) $\frac{n}{\lg n}=\Omega(n^{1-x})\ \ 0<x<1\ ,\ 3^n=O(n2^n)$

۴) $n(\lg n)^5=O(n^{\frac{1}{2}})\ ,\ \sqrt{n}=\Omega((\lg n)^5)$

**۲۳ـ** کدام یک از گزینه‌های زیر صحیح نمی‌باشد؟ (مهندسی کامپیوتر و مهندسی فناوری اطلاعات (IT) ـ آزاد ۸۹)

۱) $\frac{n^2}{\log n}=\theta(n^2)$

۲) $(1+\frac{3}{\log n})n^3=O(n^3)$

۳) $10n^3+3n^2=\Omega(n^2)$

۴) $10n^2 2^n+n\log n=\theta(n^2 2^n)$

**۲۴ـ** کدامیک از گزاره‌های زیر غلط است؟ (مهندسی فناوری اطلاعات (IT) ـ سراسری ۹۰)

۱) $f(n)+O(f(n))=\Theta(f(n))$

۲) $g(n)\neq O(f(n))\ \Rightarrow g(n)=\Omega(f(n))$

۳) $f(n)+g(n)=O(\max\{f(n),g(n)\})$

۴) $g(n)=\Omega(f(n))\Rightarrow\ g(n)=\Omega(O(f(n)))$

**۲۵ـ** زمان اجرای الگوریتمی برابر با $\sum_{i=1}^{n} i^P$ است. مرتبه زمانی آن کدام‌یک از موارد زیر است؟ (مهندسی کامپیوتر ـ آزاد ۹۰)

۱) $\theta(n^P)$ ۲) $\theta(n^{P-1})$ ۳) $\theta(n^{P+1})$ ۴) $\theta(n^P\log n)$

**۲۶ـ** کدام گزینه درست است؟ (مهندسی کامپیوتر ـ آزاد ۹۰)

۱) $f(n)+g(n)\in o(\max(f(n),g(n))$ ، $f^2(n)\in\Omega(f(n))$

۲) $f(n)+g(n)\notin o(\max(f(n),g(n))$ ، $f^2(n)\in\Omega(f(n))$

۳) $f(n)+g(n)\in o(\max(f(n),g(n))$ ، $f^2(n)\notin\Omega(f(n))$

۴) $f(n)+g(n)\notin o(\max(f(n),g(n))$ ، $f^2(n)\notin\Omega(f(n))$

**«مرتبه زمانی حلقه‌ها»**

**۲۷ـ** یک آرایه از اعداد صحیح به صورت **A[1..m]** مفروض است، به طوری که $\sum_{i=1}^{m} A[i]=s$ می‌باشد. در این صورت مرتبه اجرای الگوریتم زیر کدام یک از گزینه‌های زیر است؟ (مهندسی کامپیوتر ـ سراسری ۷۶)

```
T := 0 ;
for i := 1 to m do
   for j := 1 to  A[i] do
      T := T + 1;
```

۱) $O(s^2)$

۲) $O(m+s)$

۳) $O(m^2)$

۴) $O(m\times s)$

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



۲۸ـ در برنامهٔ زیر تعداد دفعات تکرار دستورالعمل:

```
1 for (k = 0; k <= n − 1, k ++)
2 for (i = 0; i <= n − k; i ++)
3       a[i] [i + k] = k;
```

شمارهٔ ۳ برابر است با ..... (علوم کامپیوتر ـ سراسری ۸۰)

۱) $n^2$ ۲) $\frac{n(n+1)}{2}$ ۳) $\frac{n^2}{2}$ ۴) $\frac{n(n-1)}{2}$

۲۹ـ N یک عدد صحیح 16 بیتی است که در متغیر صحیح A قرار دارد. الگوریتم α که هدف آن بررسی اول (Prime) بودن متغیر A می‌باشد دارای چه درجه‌ای است؟ (مهندسی کامپیوتر ـ سراسری ۸۱)

۱) N ۲) $N^2$ ۳) Log N ۴) نمایی

۳۰ـ پیچیدگی زمانی الگوریتم زیر کدام است؟ (علوم کامپیوتر ـ سراسری ۸۱)

```
sum = 0
for (i = 0; i < n; i ++)
for (j = 0; j < i; j++)
    for (k = 0; k < 3; k ++)
            sum ++;
```

۱) $O(n^3)$
۲) $O(n)$
۳) $O(n \log n)$
۴) $O(n^2)$

۳۱ـ الگوریتم زیر را در نظر بگیرید، مقدار y در شروع هر عبارت **while** برابر است با: (علوم کامپیوتر ـ سراسری ۸۲)

```
y ← 0
i ← n
while i ≥ 0 do
      y ← a_i + xy
      i ← i − 1
```

۱) $y = \sum_{k=0}^{n-i} a_k x^k$ ۲) $y = \sum_{k=0}^{i} a_k x^k$
۳) $y = \sum_{k=0}^{n-(i+1)} a_{k+i+1} x^k$ ۴) $y = \sum_{k=0}^{n} a_k x^k$

۳۲ـ کدام گزینه تعداد مراحل برنامهٔ زیر را به درستی بیان می‌کند: (مهندسی کامپیوتر ـ سراسری ۸۳)

```
void sum (int m , int n , float s [ ] [ ])
{     int  i , j ;
      for j=0 ; j<m;j + +)
      {    s[n-1][j]= 0;
           for (i=0 ; i<n-1;i + +)
           s[n-1][j]+ = s[i][j];
       }
}
```

۱) $2m + 2n$
۲) $mn^2 + m^2n$
۳) $m(2n+1)+1$
۴) $m(2n+1)-1$

۳۳ـ میزان زمان لازم برای اجرای قطعه برنامه زیر چگونه برآورد می‌شود؟ (مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۳)

```
for i ← 1 to  n
  for j ← n to  i
    for k ←1 to   n^2
      sum ← sum + A
```

۱) $o(n^3)$ ۲) $\theta(n^3)$
۳) $\omega(n^4)$ ۴) $\theta(n^4)$

۳۴ـ مرتبه زمانی شبه کُد زیر چیست؟ (مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۴)

```
for(i=1;i≤ n;i ++)
{
  for(j=1;j≤n;j++)
      x++;
  n - - ;
}
```

۱) n
۲) $n^2$
۳) log n
۴) n log n

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۳۵ـ** پس از اجرای قطعه کد زیر، مقدار نهایی x چه مقداری خواهد بود؟ (مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۵)

```
x=0;
for(i=1;i<=n;i++){
      for(j=1;j<=n;j++)
        x++;
      j=1;
      while(j<n){
          x++;j=j*2;
      }
}
```

۱) $n.\lceil \log_2 n \rceil$

۲) $n^2 + \lceil \log_2 n \rceil$

۳) $n^2 + n\lceil \log_2 n \rceil$

۴) $n(1+\lfloor \log_2 n \rfloor)$

**۳۶ـ** مرتبه زمانی قطعه کد زیر چیست؟ (مهندسی فناوری اطلاعات (IT) ـ آزاد ۸۵)

```
j:=n
while   j≥1   do
begin
for i:=1   to   n   do
x := x − 1
j := ⌊j/2⌋
end
```

۱) $\theta(n \log n)$

۲) $\theta(\log n)$

۳) $\theta(n)$

۴) $\theta(n^2)$

**۳۷ـ** n عدد در آرایهٔ T ذخیره شده‌اند. برای T الگوریتم ذکر شده را اجرا می‌کنیم. اگر t تعداد دفعات اجرای حلقه repeat باشد کدام یک از گزاره‌های زیر صحیح است با فرض این که $K = \lfloor \log n \rfloor$. (مهندسی کامپیوتر ـ سراسری ۸۶)

```
Procedure M(T[1..n])
    for i ← ⌊n/2⌋ downto 1 do
   i ← R
      repeat
         j ← R
       if 2j ≤ n and T[2j]>T[R] then
                  R ← 2j
       if 2j+1 ≤ n and T[2j+1]>T[R] then
                  R ← 2j+1
       swap(T[j], T[R])
until j = R
```

۱) $t \le 2\times 2^{k-2} + 3\times 2^{k-3} + \ldots + k$

۲) $t \le 3\times 2^{k-2} + 4\times 2^{k-3} + \ldots + k$

۳) $t \le 3\times 2^{k-2} + 4\times 2^{k-3} + \ldots + (k+1)$

۴) $t \le 2\times 2^{k-1} + 3\times 2^{k-2} + \ldots + (k+1)$

**۳۸ـ** مرتبه زمانی شبه کد زیر چیست؟ (مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۶)

```
for(i=1;i<=n;i++)
    for(j=1;j<=n;j++)
    {
        x++;
        n - -;
}
```

۱) n

۲) n ^ 2

۳) Logn

۴) nLogn

**۳۹ـ** با توجه به قطعه برنامه زیر کدام گزینه صحیح است؟ (علوم کامپیوتر ـ سراسری ۸۶)

```
y = m; i = n; x = 1;
while (i ≠ 0){
      if ⌊i/2⌋ ≠ i  then  x = x*y;
      y = y*y;
      i = ⌊i/2⌋;
}
```

۱) $x = m^n$

۲) $y = m^n$

۳) $m = y^n$

۴) $m = x^n$

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۴۰ـ** یک بردار از اعداد صحیح به‌صورت $V[1..n]$ مفروض می‌باشد. و رابطه $\sum_{i=1}^{n} V[i] = Q$ برقرار است. در این‌صورت پیچیدگی زمانی الگوریتم زیر برابر خواهد بود با:

(مهندسی کامپیوتر ـ آزاد ۸۶)

```
Tmp := -1;
For r := 1 to n do
    For S := 1 to V[r] do
        Tmp := Tmp + 10;
```

۱) $O(10n+Q)$
۲) $O(Q+n)$
۳) $O(n \times Q)$
۴) $O(n^2+10)$

**۴۱ـ** کدام گزینه مرتبه زمانی کد زیر را نشان می‌دهد؟

(مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۷)

```
for (i=l;i<=n;i++)
    for(j=1;j<=n;j=j+i)
        x + +;
```

۱) $\Theta(n)$
۲) $\Theta(n^2)$
۳) $\Theta(n\text{Log}n)$
۴) $\Theta(n^2\text{Log}n)$

**۴۲ـ** در کد زیر دستور x++ چند بار تکرار می‌شود/ فرض کنید $n >= 3$ است.

(مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۷)

```
for(i=3;i<=n;i=i*2)
    x++;
```

۱) $\left\lceil \text{Log}\frac{n}{3} \right\rceil$
۲) $\left\lceil \frac{(\text{Log}n)}{3} \right\rceil$
۳) $\left\lfloor \text{Log}\frac{n}{3}+1 \right\rfloor$
۴) $\left\lfloor \frac{\text{Log}(n+1)}{3} \right\rfloor$

**۴۳ـ** با فرض آن که $n = m$ باشد، پیچیدگی زمانی الگوریتم زیر کدام است؟

(مهندسی فناوری اطلاعات (IT) ـ آزاد ۸۷)

```
for    i := 1 to n do
for  j := 1 to m do
for k := 1 to  j do
a := a + 3;
```

۱) $O(\frac{m(m+1)}{2})$
۲) $O(\frac{n+1}{2})$
۳) $O(n^3)$
۴) $O(n^2)$

**۴۴ـ** مرتبه زمانی شبه کد زیر چیست؟

(مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۸)

```
for(i=1;i≤n;i=i*2)
    for(j=1;j≤n;j=j*2)
        for (k=1;k≤j;k++)
            x++;
```

۱) $\theta(n)$
۲) $\theta(n^2)$
۳) $\theta(n\text{Log}n)$
۴) $\theta(n(\text{Log}n)^2)$

**۴۵ـ** با اجرای قطعه کد زیر چند بار کلمه Test نمایش داده می‌شود؟

(مهندسی کامپیوتر ـ آزاد ۸۹)

```
for(m = 1;m <= 4;m + +)
    for(n = 1;n <= 4;n + +)
        for(k = m;k <= n;k + +)
            cout ≪ "Test \n";
```

۱) 25
۲) 16
۳) 20
۴) 26

**۴۶ـ** مرتبه زمانی شبه کد زیر کدام گزینه است؟

(مهندسی فناوری اطلاعات (IT) ـ سراسری ۹۰)

```
for(i = 1;i ≤ n;i + +)
    for (j = 1; j ≤ n; j = j + i)
        x + +;
```

۱) $n$
۲) $n^2$
۳) $\log n$
۴) $n\log n$

**۴۷ـ** تابع زیر را در نظر بگیرید. فرض کنید که $T(n)$ نشان‌دهنده تعداد عملیات ++ (هر سه) در تابع فوق باشد. اگر تابع فوق با پارامتر n فراخوانده شود، کدام رابطه‌ی بازگشتی زیر درست است؟

(مهندسی کامپیوتر ـ سراسری ۹۱)

```
int test (int n) {
  int i, j, count = 0
  for (i = 0; i < n; i + +)
    for (j = 0; j < i; j + +)
      count + +;
  return count;
}
```

۱) $T(n) = T(n-1) + 2n + 1$
۲) $T(n) = nT(n-1) + n - 1$
۳) $T(n) = T(n-1) + 2n - 1$
۴) $T(n) = nT(n-1) + n + 1$

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



## «روابط بازگشتی»

**۴۸ـ** در زیر برنامه زیر مقدار $F(3,6)$ برابر است با: (مهندسی کامپیوتر ـ سراسری ۷۵)

```
int F (int m, int n)
    {if (m= =1 | |n = =0 | | m= = n)
        return 1;
      else
        return F(m-1,n)+F(m-1,n-1)
}
```

۱) 20
۲) 10
۳) 18
۴) 4

**۴۹ـ** تابع بازگشتی زیر را در نظر بگیرید، زمان اجرای تابع فوق برابر است با: (مهندسی کامپیوتر ـ سراسری ۷۵)

```
int test (int  n)
{if (n <= 2)
    return  1;
else
    return test (n – 2) * test (n – 2);
}
```

۱) $O(n^2)$
۲) $O(n\log n)$
۳) $O(2^{\frac{n}{2}})$
۴) $O(2^n)$

**۵۰ـ** الگوریتم زیر $\binom{n}{m}$ را برای اعداد صحیح $n \geq m$ محاسبه می‌کند: (مهندسی کامپیوتر ـ سراسری۷۷)

```
Function  combination ( n , m : integer ) : integer ;
begin
   if (n=m) or (m=0) then
            return (1)
   else
            return (combination (n-1,m)+Combination (n-1, m-1) )
end;
```

برای $n > m$ دلخواه، عمل + چند بار اجرا می‌شود؟

۱) $\binom{n}{m}$
۲) $nm$
۳) $\binom{n}{m}-1$
۴) $n(n-m)$

**۵۱ـ** تابع ACK به صورت زیر تعریف می‌شود. مقدار $ACK(1,1)$ برابر است با: (مهندسی کامپیوتر ـ سراسری ۷۷)

```
int ACK (int m , int n)
{if (m< 0 or n<0)
    return 0;
else if (m = = 0)
    return (n+1);
else if (n= = 0)
    return (ACK (m-1 , 1));
else
    return  ACK(n-1 ,  ACK(m,n-1)) ;}
```

۱) 5
۲) 4
۳) 3
۴) 6

**۵۲ـ** در فراخوانی تابع زیر برای $n = 8$ چند عمل ضرب انجام می‌شود؟ فرض کنید هر عمل square نیز یک عمل ضرب نیاز دارد. (مهندسی کامپیوتر ـ سراسری ۷۸)

```
Function  count (n)
        if n ≤ 0 Then return 1
        if n = 1 Then return 2
        if n = 2 Then return 3
        return (count (n-2)*square (count (n-4)))
```

۱) 4
۲) 8
۳) 10
۴) 9

**۵۳ـ** در روال (routine) بازگشتی زیر مقدار $Rec(5,3)$ کدام است؟ (مهندسی کامپیوتر ـ سراسری ۸۰)

```
int  Rec(int p, int q)
   { int R;
    if (q <=0) return 1;
    R=Rec (p,q/2);
    R=R*R;
    if (q  mod  2 = = 0)
          return  R;
      else
         return  R*p;
     }
```

۱) 15
۲) 25
۳) 75
۴) 125

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



۵۴ـ تابع ذیل برای محاسبه بزرگ‌ترین مقسوم‌علیه دو عدد نوشته شده است. (مهندسی کامپیوتر ـ سراسری ۸۰)

```
int gcd (int a , int b)
{
    if (b= =0)
        return a;
    else
        return gcd (b,a mod b)
}
```

در این صورت می‌توان گفت: مرتبه زمانی الگوریتم ..... است.

۱) $O(\log_2 a / b)$ ۲) $O(\log_b a)$ ۳) $O(\log_2 a)$ ۴) $O(\log_2 (a-b))$

۵۵ـ برنامه زیر برای انتقال **n** حلقه از برج هانوی (برهما) استفاده می‌شود. کدام گزینه دستور حذف شده را مشخص می‌کند: (علوم کامپیوتر ـ سراسری ۸۰)

```
Procedure Tower (n:integer; from, help, to: char);
begin
   if n = 1 then
      writeln ('move a disk from', from, 'to', to);
   else begin
      Tower (n-1, from, to, help);
      writeln ('move & disk from, 'to', to);
                        دستور حذف شده
      end; {else}
end;
```

۱) Tower ( n −1, from, to, help);
۲) writeln ('move a disk from', help, 'to', to);
۳) Tower ( n −1, help, from, to);
۴) writeln ('move a disk from', from, 'to' help);

۵۶ـ مسئله برج هانوی را با سه میلهٔ **C,B,A** در نظر می‌گیریم. در اینجا هیچ حلقه‌ای را نمی‌توان مستقیماً از **A** به **B** یا از **B** بـه **A** منتقـل کـرد. یعنی چنین انتقال‌هایی تنها به کمک میلهٔ **C** انجام پذیرند. اگر در ابتدا **n** حلقه در میلهٔ **A** داشته باشیم و **T(n)** حداقل تعداد عملیات لازم برای انتقال **n** حلقه از **A** به **B** باشد، **T(n)** با کدام رابطه مشخص می‌شود؟ (فرض کنید **T(1) = 2**) (یادآوری می‌شود که در هیچ مرحله‌ای نمـی‌تـوان حلقـه‌ای را روی حلقه‌ای کوچکتر از خودش قرار داد و در هر مرحله فقط یک حلقه جابه‌جا می‌شود.) (مهندسی کامپیوتر ـ سراسری ۸۱)

۱) $T(n) = 6T(n-1) + 3$
۲) $T(n) = 3T(n-1) + 2$
۳) $T(n) = T(n-1) + T(n-2) + 1$
۴) $T(n) = T(n-1) + T(n-2) + 2$

۵۷ـ تابع زیر را در نظر بگیرید، کدام گزینه، بهترین نمایش برای تابع رشد **f(n)** برحسب **n** است؟ (مهندسی کامپیوتر ـ سراسری ۸۱)

```
int f (int n)
{
  if n < 1 return 1;
  else return f (n − 1) + g(n)
}
int g(int n)
  if n < 2 return 1;
  else return f (n − 1) + g(n/2)
}
```

۱) خطی
۲) نمایی
۳) درجه دو
۴) لگاریتمی

۵۸ـ با توجه به دو تابع زیر، خروجی $F_1(4)$ و $F_2(4)$ چیست؟ (مهندسی کامپیوتر ـ سراسری ۸۲)

```
void  F1(int x)
{
  if(x)  F2(x-1);
  printf(x);
}
```

```
void  F2(int y)
{
  if(y)  {
      printf(y+1);
      F1(y-1);
}}
```

۱) به ترتیب از چپ به راست برای $F_1(4)$ خروجی 44220 و برای $F_2(4)$ خروجی 53113 است.
۲) به ترتیب از چپ به راست برای $F_1(4)$ خروجی 20244 و برای $F_2(4)$ خروجی 1335 است.
۳) به ترتیب از چپ به راست برای $F_1(4)$ خروجی 42204 و برای $F_2(4)$ خروجی 5331 است.
۴) به ترتیب از چپ به راست برای $F_1(4)$ خروجی 42024 و برای $F_2(4)$ خروجی 5313 است.

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



۵۹ـ الگوریتم زیر را در نظر بگیرید:

```
P(X) {
    if !x  return (0)
       else return (1+P(left (x))+P(right (x))
    }
```

پیچیدگی الگوریتم برای درخت باینری x با n گره برابر است با: (علوم کامپیوتر ـ سراسری ۸۲)

۱) $O(n^2)$ ۲) $O(\log n)$ ۳) $O(n\log n)$ ۴) $O(n)$

۶۰ـ جواب فرمول بازگشتی $T(n)=2T(\frac{n}{2})+\log n!$ برابر است با: (علوم کامپیوتر ـ سراسری ۸۲)

۱) $\theta(n^2)$ ۲) $\theta(n\log n)$ ۳) $\theta(n\log^2 n)$ ۴) $\theta(n^2\log n)$

۶۱ـ کدام‌یک از شرایط زیر برای مقادیر صحیح و مثبت $a_1$ تا $a_k$ تضمین می‌کند که $\theta(n)$ حل رابطه بازگشتی $T(n)=\sum_{i=1}^{k}T(\frac{n}{a_i})+\theta(n)$ است. (مهندسی کامپیوتر ـ سراسری ۸۳)

۱) $\sum_{i=1}^{k}a_i=1$ ۲) $\sum_{i=1}^{k}a_i=k$ ۳) $\sum_{i=1}^{k}a_i<k$ ۴) $\sum_{i=1}^{k}a_i>k$

۶۲ـ مقدار تابع زیر به ازای $n\geq 2$ کدام است؟ (مهندسی کامپیوتر ـ سراسری ۸۳)

```
function g(n)
begin .
    if n≤1 then  g := n;
    else g := 5* g(n−1)−6*g(n−2)
end
```

۱) $5^n-6^n$
۲) $3^n-2^n$
۳) $3^n+2^n$
۴) $5^n+6^n$

۶۳ـ رابطه‌ی بازگشتی روبرو داده شده است: $G_n=G_{n-1}+2G_{n-2}+G_{n-3}\ ,\ G_2=4,G_1=2,G_0=1$

برای $n\geq 3$ کدام یک از گزینه‌های زیر بهترین جواب این رابطه است؟ (مهندسی کامپیوتر ـ سراسری ۸۴)

۱) $G_n\leq 2^n$ ۲) $G_n\leq 4^n$ ۳) $G_n\leq 2^{n+1}$ ۴) $G_n\leq 4^{n+1}$

۶۴ـ جواب رابطه‌ی بازگشتی $T(n)=8T(\frac{n}{9})+n\lg n$ کدام‌یک از گزینه‌های زیر است؟ (مهندسی کامپیوتر ـ سراسری ۸۴)

۱) $\theta(\lg n)$ ۲) $\theta(n)$ ۳) $\theta(n^2\lg n)$ ۴) $\theta(n\lg n)$

۶۵ـ $f(n)$ زمان اجرای الگوریتمی مطابق رابطه بازگشتی زیر محاسبه شده است. کدام گزینه صحیح است؟ (مهندسی کامپیوتر ـ سراسری ۸۴)

$$f(n)=\begin{cases}a & n=1 \quad \text{اگر}\\ bn^2+n.f(n-1) & \text{در غیر این صورت}\end{cases}$$

۱) $f(n)=\theta(2^n)$ ۲) $f(n)=\theta(n!)$ ۳) $f(n)=\theta(2^{n!})$ ۴) $f(n)=\theta(n!)^2$

۶۶ـ فرض کنید که $n$ توانی از $2$ باشد. الگوریتم زیر چه عددی را به عنوان جواب بر می‌گرداند؟ (مهندسی کامپیوتر ـ سراسری۸۴)

```
function  Test(n)
      if  n=1  Then
          return  1
      return    2*Test + 6 * n -1;
```

۱) $6n\log_2 n$
۲) $6n\log_2 n+1$
۳) $6n\log_2 n-1$
۴) $6n$

۶۷ـ مرتبه زمانی الگوریتمی با تابع زمانی زیر برابر کدام گزینه می‌باشد؟ (مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۴)

$$T(n)=\begin{cases}0 & n=0\\ 1 & n=1\\ 3T(n-1)+4T(n-2) & \text{other wise}\end{cases}$$

۱) $n^2$ ۲) $4^n$ ۳) $2^n\log n$ ۴) $n^4\log n$

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



۶۸ـ فرض کنید زمان اجرایی الگوریتمی روی $n$ ورودی، $T(n)$ بوده که به‌صورت روبرو تعریف می‌شود:

$$T(n)=\begin{cases}1 & n=1\\ T(n-1)+n & n\geq 2\end{cases}$$

زمان اجرای الگوریتم مزبور برابر کدام گزینه است؟ (مهندسی کامپیوتر ـ آزاد ۸۴)

۱) $O(n)$ ۲) $O(n^{\frac{3}{2}})$ ۳) $O(n\log n)$ ۴) $O(n^2)$

۶۹ـ در زیر برنامه زیر مقدار $F(6,3)$ برابر است با: (مهندسی کامپیوتر و مهندسی فناوری اطلاعات (IT) ـ آزاد ۸۴)

```
int F(int m , int n)
{if (m==1|| n==0|| m==n)
return 1;
else
return F(m-1,n)+F(m-1,n-1)}
```

۱) 20
۲) 10
۳) 18
۴) 15

۷۰ـ تابع برگشتی زیر را در نظر بگیرید: (مهندسی کامپیوتر و مهندسی فناوری اطلاعات (IT) ـ آزاد ۸۴)

```
Function recursive (n);
begin
if  n=1  then
recursive:=1
else
recursive:= recursive (n-1)+ recursive (n-1)
end;
```

مقدار برگشتی **recursive (5)** برابر است با:

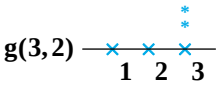
۱) 16 ۲) 8 ۳) 23 ۴) 14

۷۱ـ کدام یک از عبارات زیر جواب رابطه باز گشتی $T(n)\leq T(\frac{n}{5})+T(\frac{7n}{10})+\theta(n)$ است ؟ (مهندسی کامپیوتر ـ سراسری ۸۵)

۱) $T(n)\leq \sum_{i=0}^{2\log_5^n}(\frac{9}{10})^i n$ ۲) $T(n)\leq \sum_{i=0}^{\log_{10}^n}(\frac{1}{5})^i n$ ۳) $T(n)\leq \sum_{i=0}^{2\log_5^n}(\frac{7}{10})^i n$ ۴) $T(n)\leq \sum_{i=0}^{\log_{10}^n}(\frac{5}{10})^i n$

۷۲ـ الگوریتم روبه‌رو را در نظر بگیرید. فرض کنید الگوریتم $g$، با تعریف $g(x,y)$ در محل $x$، به تعداد $y$ علامت ستاره را می‌نویسد.

مثال:



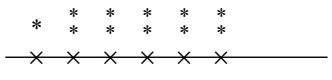

خروجی الگوریتم $f(0,8,3)$ کدامیک از موارد ذیل است؟ (مهندسی کامپیوتر ـ سراسری ۸۵)

```
f(int a, int b,intc)
{
int m =(a+b)/2;
if (c >0)
{
g(m,c);
f(a,m,c−1);
f(m,b,c−1);
}
```

۱)

۲)

۳)

۴)


۷۳ـ تابع $T(n)=2T(\lfloor\sqrt{n}\rfloor)+\log n$ را در نظر بگیرید. کدام‌یک از روابط زیر برای $T(n)$ درست است؟ (مهندسی کامپیوتر ـ سراسری ۸۵)

۱) $T(n)=O(\log\log n)$ ۲) $T(n)=O(\log n\log\log n)$ ۳) $T(n)=O(n\log\log n)$ ۴) $T(n)=O(n\log n\log n)$

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۷۴ـ** زیر برنامهٔ زیر را در نظر بگیریم. اگر $t(n)$ تعداد دفعات چاپ پیغام **Message** روی صفحه برای فراخوانی **time(n)** باشد آنگاه کدام یک از موارد زیر درست است؟ (مهندسی کامپیوتر ـ سراسری ۸۵)

```
Procedure Time (n) {
    if (n < 8) write("Message")
     else{
       Time(⌊n/2⌋)
       Time(⌊n/4⌋)
       Time(⌊n/8⌋)
      }
}
```

۱) $t(9)=4\ ;\ t(16)=5\ ;\ t(36)=8$

۲) $t(9)=3\ ;\ t(16)=5\ ;\ t(36)=9$

۳) $t(9)=3\ ;\ t(16)=6\ ;\ t(36)=8$

۴) $t(9)=4\ ;\ t(16)=6\ ;\ t(36)=9$

**۷۵ـ** یک شمارنده **K** بیتی در ابتدا مقدار صفر دارد. این شمارنده را $2^k$ بار و هر بار به مقدار **1** واحد افزایش می‌دهیم. مجموع تعداد تغییرات بیت‌های این شمارنده (از **0** به **1** و برعکس) چند تاست؟ (مهندسی کامپیوتر ـ سراسری ۸۵)

۱) $\sum_{i=1}^{k}\frac{n}{2^i}$

۲) $\theta(n^2)$

۳) $\sum_{i=1}^{k}\frac{n}{2^{i-1}}$

۴) $\theta(n\log n)$

**۷۶ـ** در مورد رابطه بازگشتی روبرو کدام گزینه صحیح است؟

$$T(n)=4T(\sqrt{n})+1 \qquad T(2)=1$$

(مهندسی کامپیوتر ـ سراسری ۸۵)

۱) $T(n)=\frac{1}{3}(\log n)^2-\frac{4}{3}$

۲) $T(n)=\frac{4}{3}(4)^n-\frac{1}{3}$

۳) $T(n)=\frac{4}{3}(\log n)^2-\frac{1}{3}$

۴) $T(n)=\frac{4}{3}(n)^2-\frac{1}{3}$

**۷۷ـ** تابع بازگشتی زیر را درنظر بگیرید (**n** توانی از **2** است). زمان اجرای تابع زیر چیست؟ (مهندسی کامپیوتر ـ آزاد ۸۵)

```
int Fun2(int n)
  {
    if (n<=1)
       return 1;
    else
       return Fun2(n/2)+n/2
  }
```

۱) $O(\log_2^n)$

۲) $O(n)$

۳) $O(n\log n)$

۴) $O(\frac{n}{2})+1$

**۷۸ـ** شمار فراوانی تابع بازگشتی زیر چیست؟ (مهندسی کامپیوتر نرم‌افزار ـ آزاد ۸۵)

```
Type     Element List = array [1 .. m] of integer;
Function rsum (var a: Element List ; n : integer) : real ;
  Begin
    if n < = 0 then rsum : = 0
    else rsum : = rsum (a , n – 1) + a[n];
  end ;
```

۱) $n(m+1)+2$

۲) $2n+2$

۳) $2n$

۴) $n(m+1)$

**۷۹ـ** حالت خاصی از مسئله برج هانوی را در نظر بگیرید. **100** سکه با شماره‌ها (و اندازه‌های) **1** تا **100** موجودند. از این تعداد $n_1$ عدد در میله‌ی شمارهٔ **1** (با شماره‌های دلخواه) به ترتیب از بزرگ به کوچک (از پایین به بالا)، و بقیه در میله شماره **3** با همین نظم و ترتیب قرار دارند. می‌خواهیم با حفظ همه مقررات برج هانوی، همه سکه‌ها را به میله شماره **3** منتقل کنیم. حداقل تعداد حرکت‌ها در بدترین حالت حرکت سکه‌ها چند تاست؟ (مهندسی کامپیوتر ـ سراسری ۸۶)

۱) $2^{100}-1$

۲) $2^{100}+1$

۳) $2^{101}-1$

۴) $2^{101}+1$

**۸۰ـ** رابطه‌های بازگشتی زیر برای اعداد صحیح $n>2$ تعریف شده‌اند و داریم $T(0)=T(1)=1$. کدام یک از این روابط جواب چند جمله‌ای **(polynomial)** ندارد؟ (مهندسی کامپیوتر ـ سراسری ۸۶)

۱) $T(N)=2T(N-2)+1$

۲) $T(N)=T(\lfloor\frac{7N}{8}\rfloor)+8N+1$

۳) $T(N)=3T(\lfloor\frac{N}{2}\rfloor)+N^2$

۴) $T(N)=T(N-1)+N^2$

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۸۱ـ جواب تابع بازگشتی روبرو کدام است؟**

$T(n) = 100T(\frac{n}{99}) + \lg(n!)$

(علوم کامپیوتر ـ سراسری ۸۶)

۱) $T(n) = \theta(n^2)$ ۲) $T(n) = \theta(n^{\log_{99}^{100}})$ ۳) $T(n) = \theta(n \lg n)$ ۴) $T(n) = \theta(n \lg^2 n)$

**۸۲ـ تابع پیچیدگی یک الگوریتم به‌صورت روبرو تعریف شده است:**

$T(n) = T(n-1) + \frac{3}{n}\ ,\ n > 1$

**در حالتی‌که n مقدار کوچکی نباشد، پیچیدگی زمانی این الگوریتم برابر است با:**

(مهندسی کامپیوتر ـ آزاد ۸۶)

۱) $O(3\ln(n))$ ۲) $O(\frac{\ln(n)}{3})$ ۳) $O(3\log_2^n)$ ۴) $O(n\log 3)$

**۸۳ـ در روال بازگشتی زیر مقدار Fun (7,5) کدام است؟**

(مهندسی فناوری اطلاعات (IT) ـ آزاد ۸۶)

```
int Fun (int x, int y)
{
int Tmp;
if (x <= 0) return 2;
Tmp = Fun(x – 3, y);
Tmp = 2 × Tmp :
if (x mod 2 == 0)
return Tmp;
else
return Tmp * y;
}
```

۱) 400
۲) 200
۳) 1000
۴) 40

**۸۴ـ فرض کنید که مجموعه متغیرها $\sum = \{a,b,c\}$ و عمل «ضرب» بر روی این متغیرها به صورت زیر انجام می‌شود. اگر یک رشته از متغیرها داده شده باشد، می‌توان آن را به صورت‌های مختلف پرانتزگذاری کرد و حاصل آن را براساس جدول فوق به دست آورد. مثلاً abcc را می‌توان به صورت ((a(bc))c) دید که حاصل آن a است ویا ((a(b(cc))) که حاصل b است. کدام‌یک از گزینه‌های زیر درست است؟**

(مهندسی کامپیوتر ـ سراسری ۸۷)

| عمل‌وند سمت چپ \ سمت راست | a | b | c |
|---|---|---|---|
| a | b | b | a |
| b | c | b | a |
| c | a | c | c |

۱) bbbba را می‌توان طوری پرانتزگذاری کرد که حاصل آن a شود.
۲) cabbbb را می‌توان طوری پرانتزگذاری کرد که حاصل آن a شود.
۳) abbbbca را می‌توان طوری پرانتزگذاری کرد که حاصل آن a شود.
۴) abbbbb را می‌توان طوری پرانتزگذاری کرد که حاصل آن a شود.

**۸۵ـ اگر معادله زمانی بازگشتی برای یک الگوریتم برابر با $T(n) = T(2n/3) + (\log_2 n)^2$ باشد، زمان مصرفی الگوریتم برابر است با:**

(مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۷)

۱) $\Theta(n^2)$ ۲) $\Theta(n^{2/3})$ ۳) $\Theta(n \lg n)$ ۴) $\Theta((\lg n)^3)$

**۸۶ـ تابع زیر برای $n > 1$ چه مقداری را به عنوان جواب تولید می‌کند؟**

(مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۷)

```
Function g(n)
        if    n<2   then
              return n
        end  if
        return 5×g(n-1)-6×g(n-2)
end g
```

۱) $5^n - 6^n$
۲) $3^n + 2^n$
۳) $5n + 6^n$
۴) $3^n - 2^n$

**۸۷ـ تابع زیر چه عملی را انجام می‌دهد؟**

(مهندسی کامپیوتر ـ آزاد ۸۷)

```
int  fun (int  x,int y)
 {
if (y= = 0)  return  x;
else  return fun (y,x%y);
 }
```

۱) بزرگ‌ترین مقسوم‌علیه مشترک دو عدد x و y را محاسبه می‌نماید.
۲) باقی‌مانده x بر y را محاسبه می‌نماید.
۳) باقی‌مانده y بر x را محاسبه می‌نماید.
۴) این تابع باقی‌مانده x بر y را محاسبه می‌نماید، لیکن در انتها مقدار y را بر می‌گرداند.

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۸۸ـ** مرتبه اجرایی الگوریتم زیر کدام است؟ (مهندسی کامپیوتر ـ آزاد ۸۷)

```
int f (int n)
 {
  if   (n= =1)  return 1;
  else return f (n-1) + f (n-1) +1;
 }
```

۱) $O(n)$
۲) $O(2^n)$
۳) $O(n\log n)$
۴) $O(2^{\frac{n}{2}})$

**۸۹ـ** خروجی تابع زیر به ازای $m = 20$ چه می‌باشد؟ (مهندسی فناوری اطلاعات (IT) ـ آزاد ۸۷)

```
float fun (int m)
{
if (m == 1) return sqrt (20);
else return sqrt (20 + fun(m – 1));
}
```

۱) 4
۲) 5
۳) 6
۴) 16

**۹۰ـ** زیر برنامه بازگشتی زیر را در نظر بگیرید. در این صورت حاصل **fun (5)** کدام است؟ (مهندسی فناوری اطلاعات (IT) و مهندسی کامپیوتر ـ آزاد ۸۷)

```
int fun (int k)
{
if (k <= 2)
return (k + 1);
else
return (fun (k – 1) + fun (k – 2));}
```

۱) 12
۲) 14
۳) 11
۴) 13

**۹۱ـ** جواب رابطه‌ی بازگشتی $T(n) = 4T(\frac{\sqrt{n}}{3}) + \log^2 n$ کدام است؟ (مهندسی کامپیوتر ـ سراسری ۸۸)

۱) $\Theta(\sqrt{n})$
۲) $\Theta(\log^2 n)$
۳) $\Theta(\log^3 n)$
۴) $\Theta(\log^2 n \log\log n)$

**۹۲ـ** با توجه به دو تابع زیر خروجی **fun1(8)** و **fun2(8)** چیست؟ (مهندسی کامپیوتر ـ آزاد ۸۸)

```
void fun1(int  a)
{
  if  ((a – 1) >= 0)
  { fun2(a – 2);}
 cout << a;
}
```

```
void fun2(int  b)
{
if (b)
  { cout << b;
    fun1(b – 2);
  }
}
```

۱) به ترتیب از چپ به راست برای fun1(8) خروجی 6248 و برای fun2(8) خروجی 8424 می‌باشد.
۲) به ترتیب از چپ به راست برای fun1(8) خروجی 62048 و برای fun2(8) خروجی 8426 می‌باشد.
۳) به ترتیب از چپ به راست برای fun1(8) خروجی 62048 و برای fun2(8) خروجی 8424 می‌باشد.
۴) به ترتیب از چپ به راست برای fun1(8) خروجی 6248 و برای fun2(8) خروجی 8424 می‌باشد.

**۹۳ـ** مرتبهٔ اجرایی الگوریتم زیر چه می‌باشد؟ (مهندسی کامپیوتر ـ آزاد ۸۸)

```
int  F(int  a)
{
  if (a == 1)return 1;
  else return  F(a – 1) + F(a – 1) + 1;
}
```

۱) $O(2^a)$
۲) $O(a)$
۳) $O(a\log a)$
۴) $O(2^{\frac{a}{2}})$

**۹۴ـ** تابع زیر چه فرمولی را محاسبه می‌نماید؟ (مهندسی کامپیوتر و مهندسی فناوری اطلاعات (IT) ـ آزاد ۸۸)

```
Function N(a,b: integer ): integer;
begin
if  a < b  then N := a
else N := N(a – b,b)
end;
```

۱) $a^b$
۲) a mod b
۳) a div b
۴) (a – b)div b

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



۹۵ـ تابع بازگشتی زیر را در نظر بگیرید. در این صورت مقدار **recursive(5)** کدام است؟ (مهندسی فناوری اطلاعات (IT) ـ آزاد ۸۸)

```
function recursive (n : integer)
begin
if (n = 1)then
recursive := 1;
else
recursive := recursive(n − 1) + (n − 3);
end;
```

۱) 4
۲) 3
۳) 5
۴) 6

۹۶ـ رابطه‌ی بازگشتی زیر را در نظر بگیرید: (مهندسی کامپیوتر ـ سراسری ۸۹)

$$F(x,0) = F(x+1,0) + F(x+1,1), if x < n$$
$$F(x,1) = 2F(x+1,0) + F(x+1,1), if\ x < n$$
$$F(n,0) = 1$$
$$F(n,1) = 0$$

اگر از این رابطه بخواهیم مقدار $F(1,1)$ را به صورت کارا حساب کنیم. چند بار عمل «جمع» (همان + در رابطه‌های فوق) را باید انجام دهیم؟

۱) $O(2^n)$ ۲) $O(n^2)$ ۳) $O(n)$ ۴) $O(2^{n-1})$

۹۷ـ در یک زمستان سرد، خرس قطبی $n$ قطعه گوشت دقیقاً به اندازه‌های 2,1 ، تا $n$ را در غاری ذخیره کرده است. او هر روز یکی از این قطعه گوشت‌ها را به صورت تصادفی انتخاب می‌کند. اگر اندازه‌ی گوشت عدد فردی بود، آن را کاملاً می‌خورد. اگر زوج بود، آن را دقیقاً نصف می‌کند. یک نصف آن را می‌خورد و نصف دیگر را مجدداً در غار قرار می‌دهد. اگر گوشتی موجود نباشد، خرس می‌میرد. با این الگوریتم، برای $n$های خیلی بزرگ روزهای باقیمانده از عمر خرس ما تابع کدام یک از گزینه‌ها خواهد بود؟ (مهندسی کامپیوتر ـ سراسری ۸۹)

۱) $\theta(n)$ ۲) $\theta(\log n)$ ۳) $\theta(n\log n)$ ۴) $\theta(n^2)$

۹۸ـ کم رشدترین حد بالای تابع بازگشتی $f(n) = 4f(\frac{n}{2}) + n^2\log n, f(1) = 1$ کدام است؟ (مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۹)

۱) $O(n^3)$ ۲) $O(n^2\log n)$ ۳) $O(n^2(\log n)^2)$ ۴) $O(n^2\log(\log n))$

۹۹ـ پیچیدگی زمانی چند جمله‌ای روبرو کدام است؟

$$P(n) = 5n^2 + 100n + 200000$$

(فرض کنید $n$ بسیار بزرگ است). (مهندسی کامپیوتر ـ آزاد ۸۹)

۱) $O(5n^2)$ ۲) $O(n^2)$ ۳) $O(100n)$ ۴) $O(200000)$

۱۰۰ـ زمان اجرای یک الگوریتم بر روی یک آرایه یک بعدی شامل $n$ عنصر به‌صورت مقابل مشخص شده است:

$$\begin{cases} T(n) = T(n-1) + \frac{2}{n} \\ T(1) = 0 \end{cases}$$

در این‌صورت کدام گزینه صحیح است؟ (مهندسی کامپیوتر ـ آزاد ۸۹)

۱) $T(n) = 2\ln(n)$ ۲) $T(n) = 2\log_2^n$ ۳) $T(n) = \frac{2}{n}\log_2^n$ ۴) $T(n) = 2n\log_2^n$

۱۰۱ـ فرض می‌گردد $n$ و $m$ مقادیری مثبت و بزرگ‌تر از یک را دارا می‌باشند. تابع زیر چه عملی را انجام می‌دهد؟ (مهندسی کامپیوتر ـ آزاد ۸۹)

```
int fun(int n,int m)
{ if (m == 2)
    return n;
else
  return n*fun(n , m − 1);
}
```

۱) $n^m$
۲) $n^{m-1}$
۳) $n^{m+1}$
۴) $m^n$

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۱۰۲ـ** مقدار **fun(8)** برای برنامه زیر کدام یک از گزینه‌ها می‌باشد؟ (مهندسی کامپیوتر ـ آزاد ۸۹)

```
int  fun(int a)
{  if (a <= 1)
      return(a * a − 5);
   return(3 * fun(a − 2) + 4);
}
```

۱) −245
۲) 160
۳) 260
۴) −270

**۱۰۳ـ** تابع زیر چه مقداری را باز می‌گرداند؟ (مهندسی کامپیوتر و مهندسی فناوری اطلاعات (IT) ـ آزاد ۸۹)

```
int  fun(int x)
{if   (x < 0) return  fun(−x) :
else if (x < 10) return  x;
else return fun(x%10);
}
```

۱) رقم پر ارزش (اولین رقم سمت چپ عدد)
۲) رقم کم ارزش (اولین رقم سمت راست عدد)
۳) فقط اعداد مثبت بین 1 تا 9
۴) اعداد مثبت بزرگ‌تر از 1

**۱۰۴ـ** پیچیدگی زمانی الگوریتم خاصی به صورت چند جمله‌ای زیر معین شده است، در این صورت زمان اجرای آن کدام است؟

(فرض کنید **n** بسیار بزرگ) $P(n) = 2^n + 100n^3 + n\log_2^n$ (مهندسی کامپیوتر و مهندسی فناوری اطلاعات (IT) ـ آزاد ۸۹)

۱) $O(100n^3)$
۲) $O(n\log_2^n)$
۳) $O(2^n + n^3)$
۴) $O(2^n)$

**۱۰۵ـ** مرتبه‌ی **T(n)** کدام است؟ (مهندسی کامپیوتر ـ سراسری ۹۰)

$$T(n) = \begin{cases} n, & n = 1 \text{ , or } n = 0 \\ \sqrt{1/2[T^2(n-1) + T^2(n-2)] + n}, & n > 1 \end{cases}$$

۱) $T(n) = \Theta(\sqrt{n})$
۲) $T^2(n) = \Theta(n^2)$
۳) $T(n) = \Theta(n^2)$
۴) $T^2(n) = \Theta(n^2 \lg n)$

**۱۰۶ـ** تعداد ستاره‌هایی که توسط الگوریتم زیر با فراخوانی **A(n)** چاپ می‌شود چقدر است؟ (علوم کامپیوتر ـ سراسری ۹۰)

```
A(int i)
{
   if (i > 0){
      A(i/2);
      A(i/2);
   }
   Print("*");
}
```

۱) n
۲) $2^{\log_2^n}$
۳) $2^n$
۴) $2^{\lfloor \log_2^n \rfloor + 1}$

**۱۰۷ـ** تابع بازگشتی زیر چه عملیاتی انجام می‌دهد؟ (مهندسی کامپیوتر ـ آزاد ۹۰)

$$T(a,b) = \begin{cases} a & : \text{اگر } a < b \\ T(a-b,b) & : \text{اگر } a \geq b \end{cases}$$

۱) a div b
۲) a mod (b − 1)
۳) (a − 1) mod b
۴) a mod b

**۱۰۸ـ** پاسخ معادله بازگشتی $F(n) = nF(n-1) + n!$ به ازای چه مقدار برای $F(1)$ برابر است با $F(n) = (n+1)!$؟ (مهندسی نرم‌افزار ـ آزاد ۹۱)

۱) صفر
۲) یک
۳) دو
۴) منفی یک

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



## پاسخنامه تست‌های طبقه‌بندی شده کنکوری فصل اول

**«نمادهای مجانبی»**

**۱ـ گزینه «۴»** تمام موارد را بررسی می‌کنیم:

$1:\begin{cases} f(n)\le Cg(n) \\ h(n)\ge Cg(n)\end{cases}\rightarrow f(n)\in O(h(n))$

بنابراین گزینه، نتیجه غلط است.

$2:\begin{cases} C_1g(n)\le f(n)\le C_2g(n) \\ g(n)\le Ch(n)\end{cases}\rightarrow \begin{matrix} h(n)\in\Omega(g(n)) \\ h(n)\in\Omega(f(n))\end{matrix}$

باز هم نتیجه ادعا شده نادرست است.

$3:\begin{cases} f(n)\ge Cg(n) \\ h(n)\le Cg(n)\end{cases}\rightarrow$

از این موارد نتیجه ادعا شده به هیچ وجه حاصل نمی‌شود.

$4:\begin{cases} f(n)\ge Cg(n) \\ g(n)\approx h(n)\end{cases}\rightarrow f(n)\in\Omega(h(n))$

بنابراین جواب گزینه ۴ است.

---

**۲ـ گزینه «۱»** دلیل نادرست بودن گزینه‌های دیگر در زیر آمده است:

$2:\text{Log}\,n!\approx n\text{Log}n$ $\qquad 3:n^2\notin O(n\log n)$ $\qquad 4:(n-1)^{n-1}+n^n\in\Omega(n^n)$

---

**۳ـ گزینه «۳»** چون داریم: $\lim\limits_{n\to\infty}\dfrac{(\log_2^n)!}{n!}=0$ در نتیجه خواهیم داشت: $n!\in\Omega(\log_2^n)!$ و یا $(\log_2^n)!\in O(n!)$

و بنابراین گزینه سوم غلط است.

---

**۴ـ گزینه «۳»** با توجه به فرض مسئله $\lim\limits_{n\to\infty}\dfrac{f_{(n)}}{g_{(n)}}=+\infty$ و با استفاده از قضیهٔ (۲) داریم: $g(n)\in O(F(n))$ و با استفاده از قضیهٔ (۱) نتیجه می‌گیریم:

$$g(n)\in O(F(n))-\Omega(F(n))\Rightarrow g(n)\in O(F(n)),g(n)\notin\Omega(F(n))$$
$$\Rightarrow f(n)\in\Omega(g(n)),f(n)\notin O(g(n))\Rightarrow f(n)\in\Omega(g(n)),f(n)\notin\theta(g(n))$$

---

**۵ـ گزینه «۱»** در هر کدام از چهارگزینه داریم:

$$(n+1)(n^2-2n+1)=n^3-2n^2+n+n^2-2n+1=n^3-n^2-n+1$$

که می‌توان به راحتی اثبات کرد:

$n^3-n^2-n+1\in\theta(n^3)$ و $n^3-n^2-n+1\notin\theta(n)$ و $n^3-n^2-n+1\notin\Omega(n^4)$ و $n^3-n^2-n+1\notin O(n^2\log n)$

ولی می‌توان با $c=10$ نشان داد که برای هر $n>10$ داریم: $n^3-n^2-n+1\le c2^n$

بنابراین می‌توان نوشت: $n^3-n^2-n+1\in O(2^n)$

---

**۶ـ گزینه «۴»** اگر $a$ عددی مثبت و $b$ عددی مثبت و ثابت باشد، مرتبه $n^a$ از $O((\log n)^b)$ بیشتر خواهد بود.

---

**۷ـ هیچ‌کدام از گزینه‌ها صحیح نیست.** اگر قرار دهیم: $f(n)=n^2,g(n)=n^3$ ، داریم: $f(n)=o(g(n))$. بنابراین گزینهٔ یک برقرار نیست. در مورد گزینهٔ دوم داریم $(2^{n^2})=O(2^{n^3})$، بنابراین $2^{f(n)}=O(2^{g(n)})$. ولی می‌بینیم که $2^{g(n)}\neq\Omega(2^{f(n)})$ و همچنین با انتخاب $g(n)=2^{n^2}$ و $f(n)=2^n$ اشتباه بودن گزینهٔ چهارم نیز مشخص خواهد شد. در نهایت اگر $g(n)=n$ و $f(n)=2n$ باشد گزینه دوم نیز نادرست خواهد بود.

---

**۸ـ هیچ‌کدام از گزینه‌ها صحیح نیست.** اگر در گزینه اول از $|\sin n|$ استفاده شده بود می‌توانست درست باشد.

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۹ـ گزینه «۱»** در مورد گزینه‌های ۲ و ۳ و ۴ نمی‌توان یک c پیدا کرد که به ازاء $\forall n \geq n_0$ داشته باشیم $n^{1+3} \leq Cn\log n(0<\varepsilon<0.1)$

و $\sqrt{n} \leq C\log n$ و $n^2 \leq C\frac{n^2}{\log n}$، ولی در مورد گزینه ۳ می‌توان $c=10$ را در نظر گرفت که به ازاء $\forall \ n \geq 5$ داریم:

$$n^{1+\varepsilon} \leq 10n\log n(0 \leq \varepsilon \leq 0.1)$$

---

**۱۰ـ گزینه «۱»** بین سه تابع داده شده رابطهٔ $h(n)<f(n)<g(n)$ برقرار است. بنابراین خواهیم داشت: $h(n)\in O(f(n))\ ,\ f(n)\in O(g(n))$

و $h(n)\in O(g(n))$ و همچنین $g(n)\in \Omega(f(n))\ ,\ g(n)\in \Omega(h(n))$ و $f(n)\in \Omega(h(n))$.

دقت کنید که رابطهٔ $\theta$ نمی‌تواند بین هیچ‌کدام از تابع‌ها برقرار باشد.

---

**۱۱ـ گزینه «۲»** با توجه به تعریف $O(g(n))$ باید داشته باشیم:

$$\exists d, n_\circ > 0 \quad \forall n > n_\circ \quad F(n) \leq dg(n) \Rightarrow F(n) \in O(g(n))$$

برای I $\quad F(n)=e^{c\sqrt{n}}, g(n)=e^{\sqrt{n}} \Rightarrow e^{c\sqrt{n}} \not\leq de^{\sqrt{n}} \ (c \geq 1)$

برای II $\quad F(n)=n^2, g(n)=O(n\log n) \Rightarrow n^2 \not\leq dn\log n$

برای III $\quad F(n)=n, g(n)=n\log n \Rightarrow n \leq d \times n\log n (d=1)(n_\circ=10)$

---

**۱۲ـ گزینه «۴»** برای حذف عنصر آخر یک لیست پیوندی یک طرفه نیاز به داشتن آدرس عنصر ماقبل آخر می‌باشد و به این دلیل باید پیمایش انجام شود که در زمان $O(n)$ امکان‌پذیر است. بقیه گزینه‌ها صحیح است.

---

**۱۳ـ گزینه «۴»** $(1+\varepsilon)^n$ نمایی به حساب می‌آید و از همه کلاس‌ها بزرگتر است، بین $\frac{n^2}{\text{Log}\, n}$ و $n\,\text{Log}\, n$ نیز اولی بزرگتر است.

---

**۱۴ـ گزینه «۲»** در مورد گزینه‌ها روابط زیر برقرار است:

در مورد گزینه اول می‌توان نوشت: $n^2\log n > n^2 \Rightarrow n^2\log n \in \Omega(n^2)\ ,\ n^2\log n \notin \theta(n^2)$

در مورد گزینه سوم داریم: $\frac{n^2}{\log n} < n^2 \Rightarrow \frac{n^2}{\log n} \in O(n^2)\ ,\ \frac{n^2}{\log n} \notin \theta(n^2)$

در مورد گزینه دوم داریم: $\frac{6n^3}{\log n+1} < n^3 \Rightarrow \frac{6n^3}{\log n+1} \in O(n^3)$

در مورد گزینه چهارم داریم: $n^3n^2+6n^23^n > n^32^n \Rightarrow n^32^n+6n^23^n \in \Omega(n^32^n)\ ,\ n^32^n+6n^23^n \notin \theta(n^32^n)$

---

**۱۵ـ گزینه «۳»** این کامپیوتر در واحد زمان الگوریتمی با مرتبه زمانی $n2^n (n=16)$ را حل می‌کند، حال اگر سرعت کامپیوتر 131072 برابر گردد، داریم:

$$16\times 2^{16}\times 131072 = 16\times 2^{16}\times 2^{17} = 32\times 2^{32}$$

بنابراین همان الگوریتم با $n=32$ را حل خواهد کرد.

---

**۱۶ـ گزینه «۱»** با توجه به تعریف $\theta(f(n))$، گزینه ۱، یعنی $6*2^n+n^2 \in \theta(2^n)$ است، بقیه گزینه‌ها صحیح می‌باشد.

---

**۱۷ـ گزینه «۳»** به‌ازای $n=500$ و $n=250$ و $n=100$ داریم: $A(n)<B(n)$

و بنابراین $A(n)$ سریع‌تر است اما به ازای nهای بزرگ‌تر از 1000 رابطه $A(n)>B(n)$ برقرار می‌گردد.

---

**۱۸ـ گزینه «۳»** به ازای اعداد کوچک‌تر از 100 داریم: $A(n)<B(n)$

بنابراین $A(n)$ سریع‌تر است اما به ازای nهای بزرگ‌تر $B(n)$ سریع‌تر است.

---

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۱۹ـ گزینه «۱»** گزاره 2 همواره برقرار است زیرا جمع یک تابع مانند $f(x)$ ایجاد خواهد نمود.

برای رد سایر گزینه‌ها به مثال‌های زیر دقت نمایید:

اگر $f(n)=\frac{1}{n} \Rightarrow f(n) \notin O(f^2(n))$

اگر $f(n)=n, g(n)=10n \Rightarrow 2^n \neq O(2^{10n})$

اگر $f(n)=4^n \Rightarrow f(\frac{n}{2})=4^{\frac{n}{2}}=2^n \neq O(f(n))$

---

**۲۰ـ گزینه «۱»** اگر فرض کنیم $\log n = x$ آنگاه داریم:

$f(n)=4^{\log n}=4^x$

$g(n)=\log^{\log n} n = x^x$

$h(n)=\log^2 n = x^2$

و به راحتی می‌توان نتیجه گرفت: $h(n)<f(n)<g(n)$ یا $x^2<4^x<x^x$

بنابراین $f(n)\in O(g(n))$ و $f(n)\in \Omega(h(n))$ می‌باشد و گزینهٔ (۱) صحیح می‌باشد.

---

**۲۱ـ گزینه «۲»** دقت کنید که:

$\Omega(f(n))+O(f(n)) \geq f(n) \Rightarrow \Omega(f(n))+O(f(n)) \in \Omega(f(n))$

---

**۲۲ـ گزینه «۴»** ایرادهای موجود در سایر گزینه‌ها به صورت مقابل است: گزینه ۳: $3^n > n2^n$ گزینه ۲: $\sqrt{n} > (\log n)^5$ گزینه ۱: $\frac{n}{\log n} > n^{1-x}$

---

**۲۳ـ گزینه «۱»** برای آنکه $\frac{n^2}{\log n} \in \theta(n^2)$ باشد باید $\frac{n^2}{\log n} \in \Omega(n^2)$ باشد، اما نمی‌توان $C$ پیدا کرد که برای آن $n_0$ وجود داشته باشد به قسمی که برای هر $n>n_0$، $\frac{n^2}{\log n} \geq cn^2$ بقیه گزینه‌ها صحیح می‌باشد.

---

**۲۴ـ گزینه «۲»** اگر قرار دهیم $f(n)=n$ , $g(n)=|n^2 \sin n|$ آنگاه $g(n)\neq \Omega(f(n))$ , $g(n)\neq O(f(n))$ و توابعی وجود دارند که لزوماً با نمادهای $O$ , $\Omega$ با یکدیگر رابطه‌ای ندارند، بنابراین گزینه دوم نادرست می‌باشد. درستی سایر گزینه‌ها به سادگی قابل بررسی است.

---

**۲۵ـ گزینه «۳»** در حالت کلی داریم:

$$\sum_{i=1}^{n} i^k \in \theta(n^{k+1})$$

---

**۲۶ـ گزینه «۱»** اگر فرض کنیم $\max(f(n),g(n))=h(n)$ باشد، آنگاه داریم:

$f(n)+g(n) \leq 2h(n) \Rightarrow f(n)+g(n)\in O(h(n)) \Rightarrow f(n)+g(n)\in O(\max(f(n),g(n)))$

و اگر $f(n)$ تابع صعودی باشد همواره داریم: $f^2(n)\geq f(n)$ و در نتیجه $f^2(n)\in \Omega(f(n))$.

---

**«مرتبه زمانی حلقه‌ها»**

**۲۷ـ گزینه «۲»** در الگوریتم داده شده حلقه $i$، $m$ بار و به ازاء $i=k(k=1\cdots m)$ حلقهٔ دوم از 1 تا $A[k]$ یعنی $A[k]$ بار تکرار می‌شود. بنابراین حلقهٔ $j$، $s=\sum_{i=1}^{m} A[i]$ بار تکرار خواهد شد. $A[1]+A[2]+\cdots+A[m]$ البته این جواب به شرطی صحیح است که $A[i](i=1..m)$ اعداد صحیح مثبت باشند.

---

**۲۸ـ گزینه «۲»** به عنوان مثال به ازای $n=3$ خواهیم داشت:

```
n = 3
for (k = ∘; k <= 2; k++)
for (i = 1; i ≤ 3 − k; i++)
    a[i][i + k] = k;
```

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



جدول اجرا به صورت زیر خواهد بود:

| $k=0$ | $k=1$ | $k=2$ |
|---|---|---|
| $i=1,2,3$ | $i=1,2$ | $i=1$ |
| اجرای دستور ۳ | اجرای دستور ۳ | اجرای دستور ۳ |

بنابراین داریم $1+2+3=6$، حال در گزینه‌ها تنها گزینه ۲ یعنی $\frac{n(n+1)}{2}$ (که در اینجا $\frac{3\times4}{2}=6$ است) صحیح می‌باشد.

---

**۲۹ـ گزینه «۴»** بررسی اول بودن یک عدد برحسب تعداد بیت‌های کد کننده عدد از مرتبه نمایی می‌باشد.

---

**۳۰ـ گزینه «۴»** توجه کنید که for سوم یک حلقه for با تکرار ثابت 3 است و در کلاس پیچیدگی بی‌تأثیر است. آنچه مهم است ارتباط دو حلقه for اولی بوده که به صورت روبرو به دست می‌آید. $1+2+\ldots+n=\frac{n(n+1)}{2}=O(n^2)$

---

**۳۱ـ گزینه «۳»** مقدار y در ابتدای حلقه (با توجه به مقادیر i) به صورت زیر می‌باشد:

$i=n: \quad y=0$

$i=n-1: y=a_n \quad =a_{n-i-1}x^{n-i-1}=a_{n-(i+1)+(i+1)}x^{n-(i+1)}$

$i=n-2: y=a_{n-1}+xa_n \quad =a_{n-(i+1)+(i+1)}x^{n-(i+1)}+a_{n-(i+1)+(i+1)+1}x^{n-(i+1)+1}$

$i=n-3: y=a_{n-2}+xa_{n-1}+x^2a_n=\sum_{k=\circ}^{n-(i+1)}a_{k+(i+1)}x^k$

البته با عددگذاری نیز می‌توان به جواب رسید: $n=2\ ,\ i=1\Rightarrow y=a_2$

که فقط در گزینه سوم صادق است.

---

**۳۲ـ گزینه «۳»** با استفاده از شیوهٔ جدول گام‌ها داریم:

```
void sum(int m,int n,Float s[][]
{
int i,j;
for(j=0; j<m; j++)
}
s[n-1][j]=0;
For(i=0; i<n-1;i++)
s[n-1][j]+=s[i][j];
}
}
```

| $\frac{s}{e}$ | تکرار | کل مراحل |
|---|---|---|
| 0 | 1 | 0 |
| 0 | 1 | 0 |
| 0 | 1 | 0 |
| 1 | $m+1$ | $m+1$ |
| 0 | $m$ | 0 |
| 1 | $m$ | $m$ |
| 1 | $m(n)$ | $mn$ |
| 1 | $m(n-1)$ | $mn-m$ |
| 0 | $m$ | 0 |
| 0 | 1 | 0 |

مجموع همهٔ مراحل : $2mn+m+1$

---

**۳۳ـ گزینه «۴»** داخلی‌ترین حلقه باعث می‌شود که دستور $sum \leftarrow sum+A$ به تعداد $n^2$ بار تکرار شود و دو حلقهٔ اول باعث می‌شوند که حلقه داخلی به تعداد $\theta(n^2)=n-1+n-2+\ldots+1$ بار اجرا شود. پس زمان کل عبارتست از $\theta(n^2)\times n^2=\theta(n^4)$.

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۳۴ـ گزینه «۲»** در کد مورد نظر، برای $i=1$ دستور $x++$، $n$ بار اجرا می‌شود و برای $i=2$، $(n-1)$ بار و برای $i=3$، $(n-2)$ بار تا $i=\frac{n}{2}$ که دستور $x++$، $\frac{n}{2}$ بار اجرا خواهد شد و داریم:

$$n+(n-1)+(n-2)+....+\frac{n}{2}=\frac{n(n+1)}{2}-\frac{\frac{n}{2}(\frac{n}{2}-1)}{2}\in O(n^2)$$

**۳۵ـ گزینه «۳»** در حلقه for(i) که از 1 تا $n$ تکرار می‌شود، دو حلقه وجود دارد؛ یکی حلقه for(j) از 1 تا $n$ و دیگری حلقه while که از $j=1$ شروع شده و با شمارندهٔ $j=j*2$ ادامه می‌یابد تا به $n$ برسد، بنابراین به ازاء هر $i$ حلقه for(j)، $n$ بار و حلقه while $\lceil Log_2^n \rceil$ بار تکرار می‌شود و در کل مقدار $x$ برابر $n^2+n\lceil Log_2^n \rceil$ خواهد شد.

**۳۶ـ گزینه «۱»** در قطعه کد مذکور دو حلقه متداخل داریم که در حلقهٔ for، $i$ از 1 تا $n$ یک واحد یک واحد افزایش می‌یابد که مرتبهٔ زمانی این حلقه $\theta(n)$ است و در حلقهٔ while، $j$ از $n$ تا یک با شمارندهٔ $j=\lfloor j/2 \rfloor$ کاهش می‌یابد که مرتبهٔ زمانی این حلقهٔ $\log_2 n$ می‌باشد. بنابراین مرتبه زمانی قطعه کد داده شده $\theta(n\log n)$ می‌باشد.

**۳۷ـ گزینه «۴»** با فرض $n=2$ حلقهٔ repeat حداکثر 2 بار اجرا خواهد شد و فقط رابطهٔ زیر صحیح است:

$$t\le 2\times 2^{\log_2^2-1}+\log_2^2+1=2+1+1=4$$

(این الگوریتم، ساخت درخت Maxheap با روش جوان‌ترین پدر است).

**۳۸ـ گزینه «۱»** حلقه for (i) از ۱ تا $n$ اجرا شده و حلقه for(j) نیز از ۱ تا $n$ اجرا می‌شود. امّا در حلقه for(j) به اجزاء هر بار اجرا یک واحد از $n$ کم می‌شود، بنابراین برای $i=1$ حلقه for(j) $\frac{n}{2}$ بار و برای $i=2$ حلقه for(j)، $\frac{n}{4}$ بار تکرار شده و با ادامه این روند، داریم:

$$\frac{n}{2}+\frac{n}{4}+\frac{n}{8}+\cdots+0=n(\frac{1}{2}+\frac{1}{4}+\frac{1}{8}+\cdots+0)\in\theta(n)$$

**۳۹ـ گزینه «۱»** شرط $[i/2]\neq i$ زمانی برقرار نیست که $i=0$ باشد و خروجی برنامه همان $x=m^n$ می‌باشد.

**۴۰ـ گزینه «۲»** داخلی‌ترین دستور به تعداد $Q$ بار اجرا می‌شود و اگر $Q=0$ باشد، آنگاه حلقه for خارجی به اندازه $n$ بار تکرار می‌شود. بنابراین مرتبه زمانی کل الگوریتم عبارتست از: $O(Q+n)$

**۴۱ـ گزینه «۳»** در قطعه کد سؤال حلقه $i$ از 1 تا $n$ یکی یکی و حلقه $j$ از 1 تا $n$ براساس مقدار $i$ افزایش می‌یابد، بنابراین مرتبه زمانی الگوریتم برابر است با:

$$n+\frac{n}{2}+\frac{n}{3}+\frac{n}{4}+\cdots+1=n(1+\frac{1}{2}+\frac{1}{3}+\cdots+\frac{1}{n})\in\theta(nLogn)$$

**۴۲ـ گزینه «۳»** اگر $n=12$ باشد، دستور $x++$، 3 بار اجرا می‌شود و برای $n=12$ فقط گزینه ۳ صحیح می‌باشد.

**۴۳ـ گزینه «۳»** در این الگوریتم، سه حلقهٔ متداخل داریم که تعداد تکرار حلقهٔ $k$ به $j$ وابستگی دارد و در نهایت بر حسب پارامتر $m$ می‌باشد و چون $m=n$، پس پیچیدگی زمانی این الگوریتم $O(n^3)$ است.

**۴۴ـ گزینه «۳»** در حلقه $i$ شمارنده $i=i*2$ است، پس مرتبه اجرایی این حلقه $\theta(Logn)$ می‌باشد و در مورد دو حلقه $k,j$، چون حلقه $k$ به $j$ وابسته است و شمارنده حلقه $j$ برابر $j=j*2$ است، تعداد تکرار دستور $x++$ برای دو حلقه $j$ و $k$ برابر است با:

$$1+2+4+8+16+\cdots+2^m \qquad 2^m <= n$$

$$=\frac{1(1-2^{m+1})}{1-2}=2^{m+1}-1 \qquad \in\theta(n)$$

پس زمان اجرایی شبه کد مورد سؤال برابر $\theta(nLogn)$ خواهد شد.

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۴۵ـ گزینه «۳»** در این قطعه برنامه و در حلقه for m، m از یک تا 4 و در حلقه for n، n از یک تا 4 و در حلقه for k، k از m تا n اجرا می‌شود یعنی اگر $m=1$ باشد حلقه for k برای $n=1$، یک بار، برای $n=2$، 2 بار و برای $n=3$، 3 بار و برای $n=4$، 4 بار اجرا می‌شود و داریم:

$$\begin{pmatrix} m=1,n=1,\dots,4 \\ k=1+2+3+4 \end{pmatrix}+\begin{pmatrix} m=2,n=1,\dots,4 \\ k=1+2+3 \end{pmatrix}+\begin{pmatrix} m=3,n=1,\dots,4 \\ k=1+2 \end{pmatrix}+\begin{pmatrix} m=4,n=1,\dots,4 \\ k=1 \end{pmatrix}$$

$$\Rightarrow k=(1+2+3+4)+(1+2+3)+(1+2)+1=10+6+3+1=20$$

---

**۴۶ـ گزینه «۴»**

وقتی $i=1$ حلقه داخلی $n$ بار اجرا می‌شود.

وقتی $i=2$ حلقه داخلی $\lfloor n/2 \rfloor$ (سقف) بار اجرا می‌شود.

وقتی $i=3$ حلقه داخلی $\lfloor n/3 \rfloor$ (سقف) بار اجرا می‌شود.

$\vdots$

وقتی $i=n$ حلقه داخلی 1 بار اجرا می‌شود.

بنابراین مرتبه‌ی زمانی: $n+\frac{n}{2}+\frac{n}{3}+\cdots+1=n(1+\frac{1}{2}+\frac{1}{3}+\cdots+\frac{1}{n})=n\log n$ می‌شود.

---

**۴۷ـ گزینه «۳»** در حلقه‌ی for i، عملیات $++$، دقیقاً $n$ بار و در حلقه‌ی for j برای مقادیر مختلف $i$ تعداد عملیات $++$، به صورت زیر است:

$$i=0,1,2,\cdots,n-1$$

$$\text{for j} \rightarrow 0+1+2+\cdots+n-1=\frac{n(n-1)}{2}$$

و برای دستور $\text{Count}++$ نیز به ازای مقادیر مختلف $i$، تعداد عملیات $++$، به صورت زیر می‌باشد:

$$i=0,1,2,\cdots,n-1$$

$$\text{Count}++ \rightarrow 0+1+2+\cdots+n-1=\frac{n(n-1)}{2}$$

بنابراین $T(n)$ را می‌توان به صورت مقابل نوشت: $T(n)=n+\frac{n(n-1)}{2}+\frac{n(n-1)}{2}=n^2$

که در آن $T(0)=0$ و $T(1)=1$ و $T(2)=4$ است، که در این صورت معادله‌ی بازگشتی $T(n)=T(n-1)+2n-1$ صحیح می‌باشد.

---

**«روابط بازگشتی»**

**۴۸ـ گزینه «۴»** ضابطهٔ تابع بازگشتی F، به صورت زیر است:

$$F=(m,n)=\begin{cases} 1 & \text{اگر } m=1 \text{ یا } n=0 \text{ یا } m=n \\ F(m-1,n)+F(m-1,n-1) & \text{در غیر این صورت} \end{cases}$$

$$F(3,6)=F(2,6)+F(2,5)=2+2=4$$

$$F(2,6)=F(1,6)+F(1,5)=1+1=2$$

$$F(2,5)=F(1,5)+F(1,4)=1+1=2$$

$$F(1,6)=1$$

$$F(1,5)=1$$

$$F(1,4)=1$$

---

**۴۹ـ گزینه «۳»** تابع test به صورت زیر تعریف شده است:

$$\text{test}(n)=\begin{cases} 1 & n<=2 \\ 2\times \text{test}(n-2) & n>2 \end{cases}$$

با توجه به ضابطهٔ تابع test اگر $n>2$ باشد، دو مرتبه تابع در خودش فراخوانی می‌شود. پس تعداد فراخوانی‌ها $2^k$ خواهد بود و چون هر مرتبه تابع test با پارامتر $n-2$ فراخوانی می‌شود پس بازهٔ 2 تا $n$، نصف شده و $k=\frac{n}{2}$ خواهد بود.

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۵۰ـ گزینه «۳»** در تابع Combination که به صورت بازگشتی تعریف شده است هر گاه m=n یا $m=0$ باشد مقدار یک و در غیر این صورت دو بار تابع فوق به صورت بازگشتی فراخوانی می‌شود و این عملیات تکرار شده تا اینکه به مقدار یک برسد. بنابراین return (1) به تعداد $\binom{n}{m}$ بار تکرار می‌شود.

و با $\binom{n}{m}-1$ عملِ جمع، بین این مقادیرِ یک بازگشتی، مقدار $\binom{n}{m}$ به دست می‌آید.

---

**۵۱ـ گزینه «۳»** ضابطهٔ ریاضی تابع ACK عبارت است از:

$$ACK(m,n)=\begin{cases}0 & \text{اگر } m<0 \text{ یا } n<0\\ n+1 & \text{اگر } m=0\\ ACK(m-1,1) & \text{اگر } n=0\\ ACK(m-1,ACK(m,n-1)) & \text{در غیر این صورت}\end{cases}$$

با استفاده از ضابطهٔ فوق داریم:

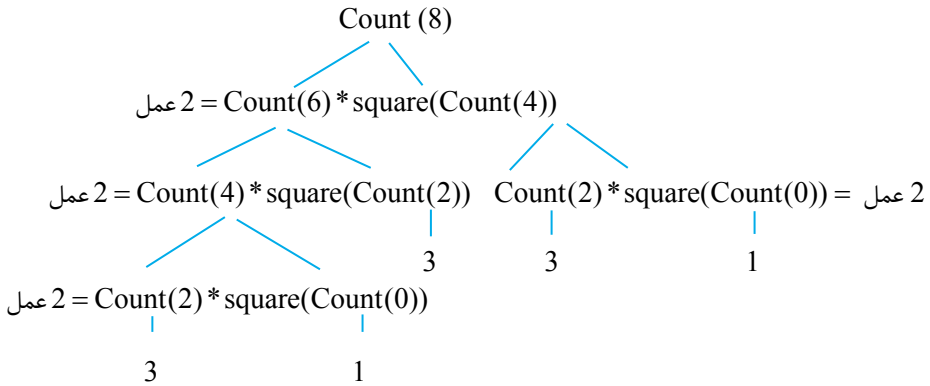
$$ACK(1,1)=ACK(0,ACK(1,0))=ACK(0,2)=2+1=3$$
$$ACK(1,0)=ACK(0,1)=1+1=2$$

---

**۵۲ـ گزینه «۲»** ضابطهٔ ریاضی تابع Count عبارت است از:

$$Count(n)=\begin{cases}1 & \text{اگر } n\le 0\\ 2 & \text{اگر } n=1\\ 3 & \text{اگر } n=2\\ Count(n-2)*square(Count(n-4)) & \text{در غیر این صورت}\end{cases}$$

حال با استفاده از ضابطهٔ ریاضی تابع، درخت فراخوانی‌های تابع Count به ازاء $n=8$ را رسم می‌کنیم. داریم:




با توجه به فرض مسئله که عمل square یک عمل ضرب نیاز دارد، تابع Count برای $n=8$ کلاً ۸ ضرب انجام می‌دهد.

---

**۵۳ـ گزینه «۴»** با استفاده از روال بازگشتی ذکر شده، ضابطهٔ ریاضی تابع Rec به این صورت است:

$$Rec(p,q)\begin{cases}1 & \text{اگر } q<=0\\ Rec^2\ (p,\frac{q}{2}) & \text{اگر } q,q>0 \text{ زوج باشد}\\ p*Rec^2(p,\frac{q}{2}) & \text{اگر } q,q>0 \text{ فرد باشد}\end{cases}$$

بنابراین داریم:

$$Rec(5,3)=5\times Rec^2\ (5,1)=5\times 5^2=5\times 25=125$$
$$Rec(5,1)=5\times Rec^2\ (5,0)=5\times 1=5$$
$$Rec(5,0)=1$$

---

**۵۴ـ گزینه «۲»** تابع gcd(a،b) یک تابع بازگشتی است. در این تابع تا زمانی که $b\neq 0$ است، gcd(b،a mod b) فراخوانی می‌شود. در واقع هر بار جای b با a mod b عوض می‌شود. این تکرار مانند یک حلقه بوده با شمارندهٔ a mod b و با توجه به اینکه اگر شمارندهٔ حلقه‌ای تقسیمات متوالی بر k باشد، تابع پیچیدگی آن حلقه، $\log_k$ خواهد بود، بنابراین مرتبهٔ زمانی الگوریتم gcd، $O(\log_b^a)$ است.

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



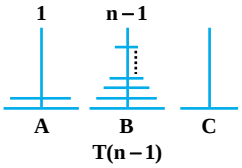
**۵۵ـ گزینه «۳»** به دستور اصلی $Tower(n-1,from,to,help)$ دقت کنید. در اینجا از میله اول (که با from مشخص شده) $n-1$ حلقه به میله دوم (که با to مشخص شده) توسط میله سوم (که با help مشخص شده) منتقل می‌شود. سپس حلقه آخر از میله اول به میله سوم که کمکی بود منتقل می‌شود. اکنون با استفاده از دستور: $('moue\ adish\ from', from, 'to', to)$

$n-1$ حلقهٔ مانده، به میله سوم منتقل می‌شود، بنابراین باید میله اول را به عنوان میله کمکی بگیریم. در نتیجه باید دستور زیر اضافه شود:

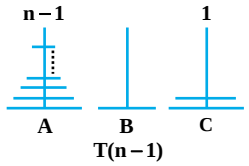
$Tower\ (n-1, help, from, to);$

---

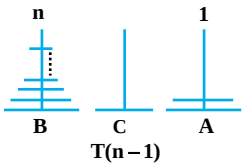
**۵۶ـ گزینه «۲»** مراحل انتقال در زیر آمده است:




انتقال $n-1$ دیسک با رعایت قانون $\Rightarrow$ انتقال دیسک بزرگ از A به C $\Rightarrow$ انتقال $n-1$ دیسک از میله B به A با رعایت قوانین بیان شده




$\Rightarrow$ انتقال دیسک بزرگ از C به B




$\Rightarrow T(n-1)$ انتقال $n-1$ دیسک با رعایت قوانین بیان شده از A به B

بنابراین رابطه بازگشتی مورد نظر به صورت روبرو خواهد بود: $T(n)=3T(n-1)+2$

---

**۵۷ـ گزینه «۲»** زیرا اگر $f(n)$ را در حالت کلی به دست آوریم، خواهیم داشت:

$$f(n)=f(n-1)+g(n)=f(n-1)+f(n-1)+g(\frac{n}{2})=2f(n-1)+g(\frac{n}{2})$$

جواب رابطهٔ بازگشتی $f(n)=2f(n-1)$ برابر است با $2^n$. بنابراین این تابع نیز، نمایی خواهد بود.

---

**۵۸ـ گزینه «۴»** توابع، $F_2, F_1$ دو تابع بازگشتی غیرمستقیم هستند، در $F_1$، $F_2$ و در $F_2$، $F_1$ فراخوانی می‌شود. چون در $F_1$ ابتدا شرط بررسی شده و سپس عمل چاپ اجرا می‌شود، بنابراین اعمال چاپ از انتها به ابتدا انجام می‌شود، این روند در $F_2$ برعکس خواهد بود. پس در $F_1(4)$، ابتدا $F_2(3)$ فراخوانی شده و در $F_2(3)$، عدد 4 چاپ و سپس $F_1(2)$ فراخوانی می‌شود. در $F_1(2)$، $F_2(1)$ فراخوانی شده و در $F_2(1)$ عدد 2 چاپ شده و $F_1(0)$ فراخوانی می‌شود، در $F_1(0)$ چون شرط غلط است پس فقط چاپ عدد صفر را داریم. بنابراین فقط گزینهٔ چهار صحیح است. البته ادامهٔ این روند، $F_1(4)$، عدد 42024 و $F_2(4)$ عدد 5313 را در خروجی چاپ می‌کند.

---

**۵۹ـ گزینه «۴»** (علامت ! را به معنای Not بگیرید.) پیچیدگی این الگوریتم، خطی و معادل گره‌های درخت خواهد بود. لازم نیست که معادله بازگشتی را حل کنید.

---

**۶۰ـ گزینه «۳»** با روش جایگذاری و تکرار، جواب را به دست می‌آوریم.

$$T(n)=2T(\frac{n}{2})+Logn!=2^2T(\frac{n}{2^2})+Log(n-1)!+Logn!=\ldots=2^{Logn}T(1)+\sum_{i=1}^{n}Log(i!)$$

به کمک رابطه $Log\,ab=Log\,a+Log\,b$ داریم:

$$n+\sum_{i=1}^{n}\sum_{j=1}^{n}Log(j)=n+\sum_{i=1}^{n}\frac{Logn\times Logn+1}{2}=n+(Log\,n)^2\times n\rightarrow T(n)\in\theta(n\,Log^2 n)$$

البته می‌توان با استفاده از تبصره بیان شده در قضیه اصلی نیز تست را حل نمود.

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۶۱ـ هیچ‌کدام از گزینه‌ها صحیح نیست.** شرط کافی به‌صورت $\sum_{i=1}^{k}\frac{1}{a_i}<1$ خواهد بود. در این حالت با توجه به صحیح و مثبت بودن مقادیر ضرایب، از قضیه akra-bazzi می‌توان تضمین کرد مرتبه رابطه به صورت $\theta(n)$ خواهد بود.

---

**۶۲ـ گزینه «۲»** می‌توان این تابع را به شکل یک رابطهٔ بازگشتی به صورت زیر نوشت:

$$g(n)=5g(n-1)-6g(n-2), g(0)=0\ ,\ g(1)=1$$

$$\Rightarrow \text{معادلهٔ مشخصه: } r^2-5r+6=0 \Rightarrow r_1=2, r_2=3 \Rightarrow g(n)=\alpha(2^n)+\beta(3)^n$$

و فقط گزینه دوم این شکل را دارد.

---

**۶۳ـ گزینه «۲»**

$$G_n=G_{n-1}+2G_{n-2}+G_{n-3} \quad ; \quad G_n<4G_{n-1}\Rightarrow G_n<4^n$$

زیرا جواب معادله بازگشتی $G_n=4G_{n-1}$ برابر $4^n$ است.

---

**۶۴ـ گزینه «۴»** از روش قضیه اصلی استفاده می‌کنیم، داریم:

$$\text{Log}\frac{8}{9}<1\Rightarrow n^{\log\frac{8}{9}}\in O(n\log n)\ \&\ n^{\log 8},\notin\theta(n\log n)$$

پس:

$$T(n)\in\theta(n\log n)$$

---

**۶۵ـ گزینه «۲»**

**روش اول:** می‌دانیم رابطه $f(n)=nf(n-1)$ و $f(1)=1$ تابع فاکتوریل را مشخص می‌کند و در اینجا شرط اولیه $f(1)=a$ و $bn^2$ باز هم از لحاظ مجانبی تغییری ایجاد نمی‌کنند پس:

$$f(n)\in\theta(n!)$$

**روش دوم:** اگر طرفین رابطهٔ $T(n)=nT(n-1)+bn^2$ را بر $n!$ تقسیم کنیم، داریم:

$$\frac{T(n)}{n!}=\frac{T(n-1)}{(n-1)!}+\frac{bn^2}{n!}$$

که در حد بی‌نهایت می‌توان $\frac{bn^2}{n!}$ را صفر در نظر گرفت. بنابراین خواهیم داشت:

$$\frac{T(n)}{n!}=\frac{T(n-1)}{(n-1)!}$$

حال اگر قرار دهیم $A(n)=\frac{T(n)}{n!}$، می‌توان نوشت:

$$A(n)=\begin{cases}A(n-1) & \text{اگر } n>1\\ a & \text{اگر } n=1\end{cases}$$

بنابراین $A(n)=a$ است، در نتیجه:

$$T(n)=n!\times a\Rightarrow T(n)\in O(n!)$$

---

**۶۶ـ گزینه «۲»** ضابطهٔ ریاضی $test(n)$ عبارت است از:

$$test(n)=\begin{cases}1 & n=1 \text{ اگر}\\ 2\times test(\frac{n}{2})+6*n-1 & n<>1 \text{ اگر}\end{cases}$$

به عنوان مثال $test(2^3)$ را محاسبه می‌کنیم، داریم:

$$test(2^3)=2\times test(2^2)+6\times 2^3-1=98+48-1=145$$
$$test(2^2)=2\times test(2)+6\times 2^2-1=26+24-1=49$$
$$test(2)=2\times test(1)+6\times 2-1=2+12-1=13$$
$$tes(1)=1$$

بنابراین $test(2^3)=145$ که برابر $6\times 2^3\times\log_2^{2^3}+1=48\times 3+1=145$ خواهد شد.

---

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۶۷ـ گزینه «۲»** معادله مشخصه برای رابطه بازگشتی داده شده را می‌توان به صورت مقابل در نظر گرفت: $r^2-3r-4=0 \Rightarrow r_1=4, r_2=-1$

بنابراین $T(n)$ به صورت روبرو قابل محاسبه است: $T(n)=c_1(4)^n+c_2(-1)^n \Rightarrow T(n)\in\theta(4^n)$

---

**۶۸ـ گزینه «۴»** اگر از روش جایگذاری استفاده کنیم داریم:

$$T(n)=T(n-1)+n$$
$$\Rightarrow T(n)=T(n-2)+(n-1)+n \Rightarrow T(n)=T(n-3)+(n-2)+(n-1)+n$$

بنابراین می‌توانیم بنویسیم: $T(n)=1+2+3+\ldots+n=\frac{n(n+1)}{2}\in\theta(n^2)$

---

**۶۹ـ گزینه «۱»** با استفاده از ضابطهٔ ریاضی تابع داریم:

$$f(m,n)=\begin{cases}1 & \text{اگر } m=1 \text{ یا } n=0 \text{ یا } m=n\\ f(m-1,n)+f(m-1,n-1) & \text{در غیر این صورت}\end{cases}$$

$f(6,3)=f(5,3)+f(5,2)=10+10=20$
$f(5,3)=f(4,3)+f(4,2)=4+6=10$
$f(4,3)=f(3,3)+f(3,2)=1+3=4$
$f(3,3)=1$
$f(4,1)=f(3,1)+f(3,0)=3+1=4$
$f(3,1)=f(2,1)+f(2,0)=2+1=3$
$f(2,1)=f(1,1)+f(1,0)=1+1=2$
$f(1,1)=1$

$f(5,2)=f(4,2)+f(4,1)=6+4=10$
$f(4,2)=f(3,2)+f(3,1)=3+3=6$
$f(3,2)=f(2,2)+f(2,1)=1+2=3$
$f(2,2)=1$
$f(3,0)=1$
$f(2,0)=1$
$f(1,0)=1$

---

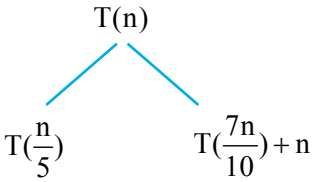
**۷۰ـ گزینه «۱»** با استفاده از ضابطهٔ ریاضی تابع داریم $(\text{recursive}=r)$:

$$r(n)=\begin{cases}1 & \text{اگر } n=1\\ r(n-1)+r(n-1)=2r(n-1) & \text{در غیر این صورت}\end{cases}$$

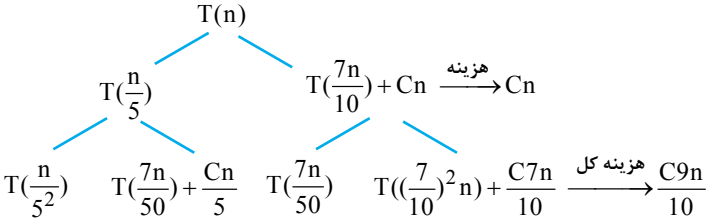
$r(5)=2r(4)=2\times 8=16$
$r(4)=2r(3)=2\times 4=8$
$r(3)=2r(2)=2\times 2=4$
$r(2)=2r(1)=2\times 1=2$
$r(1)=1$

---

**۷۱ـ گزینه «۱»** درخت بازگشت مربوط به رابطه بازگشتی داده شده به صورت زیر می‌باشد:




حال اگر گره‌های $T(\frac{7n}{10})$ و $T(\frac{n}{5})$ را گسترش دهیم خواهیم داشت:

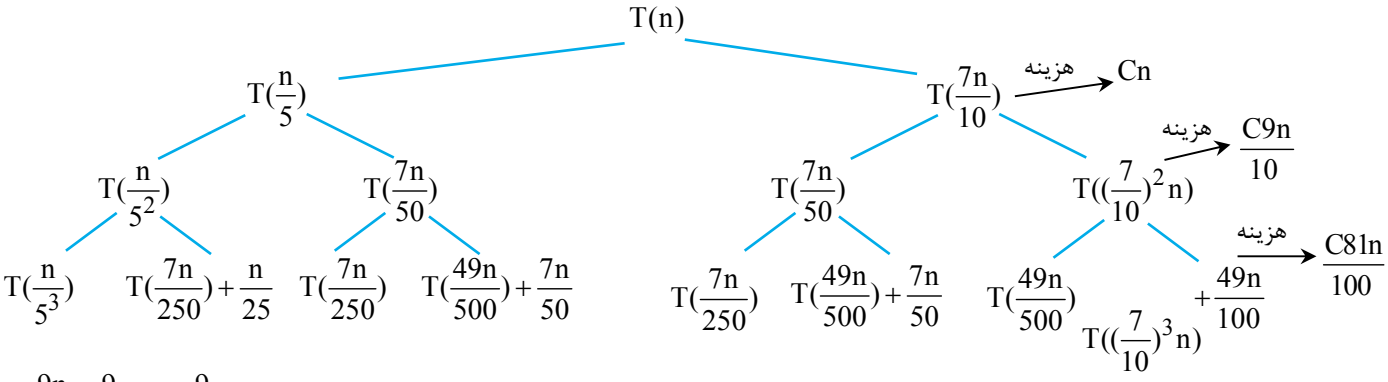



www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



با ادامه سطح بعدی درخت به صورت زیر خواهد بود:



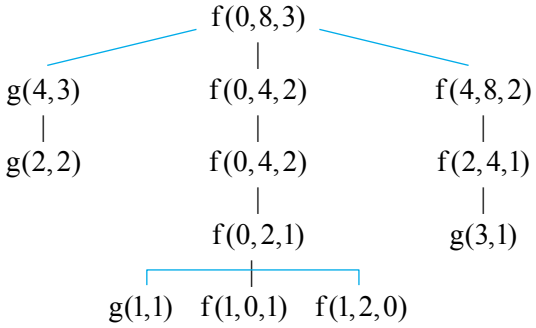

با ادامه همین روند هزینه سطوح درخت به صورت روبرو خواهد بود. $n,\frac{9n}{10},(\frac{9}{10})^2 n,(\frac{9}{10})^3 n,\ldots$

با توجه به این که شاخه مربوط به $T((\frac{7}{10})^i n)$ دیرتر از بقیه شاخه‌های درخت متوقف می‌شود، بنابراین برای محاسبه یک حد بالا از این شاخه استفاده می‌نماییم:

$$T(n)\le \sum_{i=0}^{\log_{\frac{10}{7}} n} (\frac{9}{10})^i n$$

---

**۷۲ـ گزینه «۲»**




با توجه به این درخت بازگشتی فقط گزینه ۲ می‌تواند درست باشد.

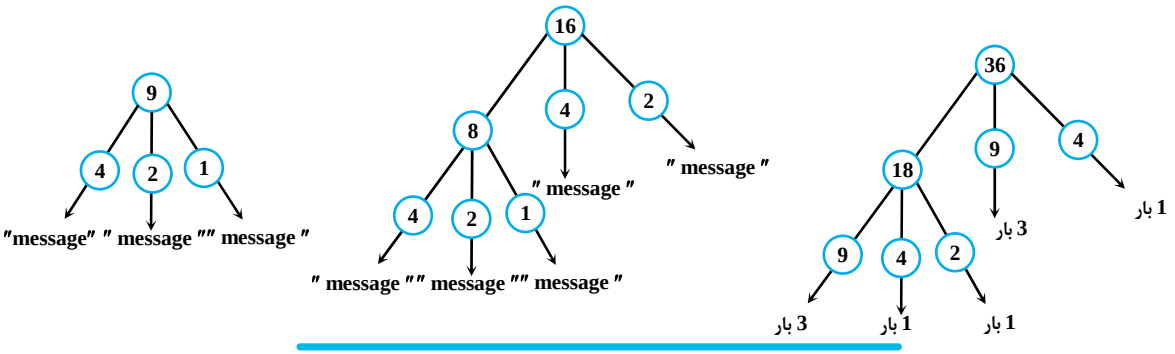
---

**۷۳ـ گزینه «۲»** همانگونه که در متن درس ذکر شد، علامت‌های $\lceil\ \rceil$ و $\lfloor\ \rfloor$ در به دست آوردن پیچیدگی توابع بازگشتی نقشی ندارند، بنابراین $T(n)=2T(\sqrt{n})+\log n$. حال از روش تغییر متغیر استفاده می‌کنیم و قرار می‌دهیم $n=2^k$، پس $T(2^k)=2T(2^{\frac{k}{2}})+\log 2^k$. اگر قرار دهیم $T(2^k)=t(k)$، خواهیم داشت:

$$t(k)=2t(\frac{k}{2})+k \Rightarrow t(k)=\theta(k\log k) \xrightarrow{k=\log n} T(n)=T(\log n \log\log n)$$

---

**۷۴ـ گزینه «۲»** اگر به درخت‌های زیر توجه کنیم جواب واضح است:




---

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۷۵ـ گزینه «۳»** در یک شمارنده کم‌ارزشترین بیت همواره تغییر می‌کند یعنی، با هر شمارش مقدار کم‌ارزش‌ترین بیت عوض می‌شود. مقدار دومین کم‌ارزشترین بیت پس از هر دو بار شمارش عوض می‌شود و مقدار سومین کم ارزش‌ترین بیت پس از هر چهار بار و .....

پس تعداد کل برابر است با: $n+\frac{n}{2}+\frac{n}{4}+.....$

---

**۷۶ـ گزینه «۳»** با انجام تغییر متغیر $n=2^m$ خواهیم داشت: $T(2^m)=4T(2^{\frac{m}{2}})+1$

اگر قرار دهیم $T(2^m)=t(m)$ داریم: $t(m)=4t(\frac{m}{2})+1$

$$t(m)\in\theta(m^2)\Rightarrow t(n)\in\theta((\log_2^n)^2)$$

که با جایگذاری $T(2)$ گزینه ۳ جواب خواهد بود.

---

**۷۷ـ گزینه «۱»** تابع بازگشتی مربوط به مرتبه زمانی الگوریتم داده شده را می‌توان به صورت زیر درنظر گرفت: $T(n)=T(\frac{n}{2})+\theta(1)$

بنابراین: $T(n)\in\theta(\log_2^n)$

---

**۷۸ـ گزینه «۲»** با کمی دقت مشخص می‌شود که تابع rsum یک تابع بازگشتی با دو پارامتر می‌باشد، که پارامتر اوّل یک آرایه بوده و پارامتر دوّم یک عدد صحیح است و در این تابع تا زمانی‌که $n<=0$ نباشد، دستور rsum : = rsum (a,n-1) +a[n]; اجرا می‌شود، یعنی از مقدار n دریافتی تا $n=1$ این دستور اجرا می‌شود و سپس برای $n=0$ دستور rsum : = 0 اجرا خواهد شد. بنابراین گام این تابع برابر با گام if که $(n+1)$ بار شرط چک شده به اضافهٔ تعداد گام دستورات درون if است، که چون هر بار یکی از دو دستور اجرا می‌شود، آن هم برابر $n+1$ است. بنابراین گام این تابع برابر $2n+2$ می‌باشد.

---

**۷۹ـ گزینه «۱»** در حالت کلی تعداد حرکات لازم برای انتقال n دیسک برابر است با $2^n-1$. (بدترین حالت، زمانی است که همه در میله شماره ۱ باشند)

---

**۸۰ـ گزینه «۱»** در گزینهٔ (۱) با استفاده از جدول ذکر شده در این فصل داریم: $T(n)\in O(2^{n/2})$ و جواب چند جمله‌ای نخواهد بود.

---

**۸۱ـ گزینه «۲»** اگر از قضیه اصلی استفاده کنیم خواهیم داشت: $T(n)=\theta(n^{\log_{99}^{100}})$

$$\text{Log}(n!)\cong n\log n$$

$$n\log n\in O(n^{\log_{99}^{100}-\varepsilon})$$

---

**۸۲ـ گزینه «۱»** اگر از روش جایگذاری استفاده کنیم، خواهیم داشت:

$$T(n)=T(n-1)+\frac{3}{n}$$

$$\Rightarrow T(n)=T(n-2)+\frac{3}{n-1}+\frac{3}{n}$$

$$\Rightarrow T(n)=T(n-3)+\frac{3}{n-2}+\frac{3}{n-1}+\frac{3}{n}$$

$$\vdots$$

$$T(n)=\frac{3}{2}+\frac{3}{3}+\frac{3}{4}+...+\frac{3}{n-2}+\frac{3}{n-1}+\frac{3}{n}=3\sum_{i=2}^{n}\frac{1}{i}\approx 3\ln n$$

---

**۸۳ـ گزینه «۱»** با استفاده از ضابطهٔ ریاضی fun داریم:

$$fun(x,y)=\begin{cases}2 & \text{اگر } x<=0\\ 2\times fun(x-3,y) & \text{اگر } x>0 \text{ و } x \text{ زوج باشد}\\ 2\times fun(x-3,y)\times y & \text{اگر } x>0 \text{ و } x \text{ فرد باشد}\end{cases}$$

$$fun(7,5)=2\times fun(4,5)\times 5=2\times 40\times 5=400$$

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



$$fun(4,5)=2\times fun(1,5)=2\times 20=40$$
$$fun(1,5)=2\times fun(-2,5)\times 5=2\times 2\times 5=20$$
$$fun(-2,5)=2$$

---

**۸۴ـ گزینه «۱»** می‌توان bbba را به صورت روبرو پرانتزگذاری نمود:

$$((\underbrace{bb}_{b})((\underbrace{bb}_{b})a))=(b(\underbrace{ba}_{c}))=(bc)=a$$

البته گزینه ۳ نیز می‌تواند درست باشد.

$$(\underbrace{(\underbrace{(ab)}_{b}\underbrace{(bb)}_{b})}_{b}(b\underbrace{(ca)}_{a}))=(b\underbrace{(ba)}_{c})=a$$

---

**۸۵ـ گزینه «۴»** با توجه به تابع داده شده درخت بازگشت مربوطه دارای عمق $\log_{\frac{3}{2}}^{n}$ خواهد بود و هزینه هر سطح نیز $(\log n)^2$ می‌باشد، بنابراین در کل هزینه به $O(\log_n^3)$ محدود خواهد بود.

---

**۸۶ـ گزینه «۴»** تابع داده شده یک رابطه بازگشتی به صورت مقابل را ایجاد می‌نماید: $g(n)=5g(n-1)-6g(n-2)$

معادله مشخصه این رابطه به صورت روبرو خواهد بود: $r^2-5r+6=0$

ریشه‌های معادله مشخصه عبارتند از: $r_2=3\ ,\ r_2=2$

با توجه به شرایط مرزی $g(1)=1$ و $g(0)=0$ خواهیم داشت: $g(n)=3^n-2^n$

---

**۸۷ـ گزینه «۱»** این الگوریتم به صورت بازگشتی بزرگ‌ترین مقسوم‌علیه مشترک دو عدد x و y را به دست می‌آورد یا به عبارت دیگر در هر تکرار مقدار y را بجای x و باقیمانده x بر y را به جای y ارسال می‌کند تا جایی که y برابر صفر شود که بزرگ‌ترین مقسوم‌علیه مشترک برابر x خواهد بود.

---

**۸۸ـ گزینه «۲»** تابع بازگشتی f دو بار با پارامترهای $n-1$ خودش را فراخوانی می‌کند بنابراین: $T(n)=T(n-1)+T(n-1)\Rightarrow T(n)\in O(2^n)$

---

**۸۹ـ گزینه «۲»** با استفاده از ضابطهٔ ریاضی تابع، داریم:

$$fun(m)=\begin{cases}\sqrt{20} & \text{اگر } m=1\\ \sqrt{20+fun(m-1)} & \text{در غیر این صورت}\end{cases}$$

$$fun(20)=\sqrt{20+fun(19)}=\sqrt{20+\sqrt{20+\sqrt{20+\cdots}}}\simeq 5$$
$$fun(19)=\sqrt{20+fun(18)}=\sqrt{20+\sqrt{20+\sqrt{20+\cdots}}}$$
$$\vdots \qquad\qquad \vdots$$
$$fun(3)=\sqrt{20+fun(2)}=\sqrt{20+\sqrt{20+\sqrt{20}}}$$
$$fun(2)=\sqrt{20+fun(1)}=\sqrt{20+\sqrt{20}}$$
$$fun(1)=\sqrt{20}$$

---

**۹۰ـ گزینه «۴»** با استفاده از ضابطهٔ ریاضی تابع fun داریم:

$$fun(k)=\begin{cases}k+1 & \text{اگر } k<=2\\ fun(k-1)+fun(k-2) & \text{در غیر این صورت}\end{cases}$$

$$fun(5)=fun(4)+fun(3)=8+5=13$$
$$fun(4)=fun(3)+fun(2)=5+3=8$$
$$fun(3)=fun(2)+fun(1)=3+2=5$$
$$fun(2)=2+1=3$$
$$fun(1)=1+1=2$$

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۹۱ـ گزینه «۴»** برای حل این رابطه بازگشتی از یک تغییر متغیر استفاده می‌کنیم:

$$n = 3^m \Rightarrow \log_3^n = m$$

$$T(3^m) = 4T(\frac{3^{\frac{m}{2}}}{3}) + m^2$$

$$\left.\begin{array}{l} T(3^m) = 4T(3^{\frac{m}{2}-1}) + m^2 \\ \Rightarrow \quad T(3^m) = S(m) \\ T(3^{\frac{m}{2}-1}) = S(\frac{m}{2}-1) \end{array}\right\} \Rightarrow S(m) = 4S(\frac{m-2}{2}) + m^2 \Rightarrow S(m) \in \theta(m^2 \log m)$$

$$\Rightarrow T(n) \in \theta(\log^2 n \log\log n)$$

**۹۲ـ گزینه «۲»** توابع fun1, fun2 دو تابع بازگشتی غیرمستقیم هستند. به طوری که در fun1 تابع fun2 و در fun2 تابع fun1 صدا می‌شود و چون در تابع fun1 ابتدا fun2 صدا شده و بعد a چاپ می‌شود، مقادیر خروجی از آخر به اول چاپ خواهد شد و چون در تابع fun2 ابتدا b چاپ شده و بعد fun1 صدا می‌شود، پس مقادیر خروجی از اول به آخر چاپ می‌شود. با توجه به موارد ذکر شده fun1(8) در خروجی 62048 و fun2(8) در خروجی 8426 را چاپ خواهد کرد.

**۹۳ـ گزینه «۱»** با توجه به تعداد فراخوانی‌های تابع f در خودش معادله بازگشتی تابع f به صورت زیر است:

$$T(a) = 2T(a-1)$$

که مرتبه اجرایی این تابع بازگشتی $O(2^a)$ می‌باشد.

**۹۴ـ گزینه «۲»** در این تابع تا زمانی که مقدار $a < b$ نباشد، به جای a، a-b را ارسال می‌کند؛ در واقع از پارامتر a مقادیر b کم می‌شود تا زمانی که مقدار a کوچک‌تر از b شود، آنگاه مقدار a را بر می‌گرداند که در این صورت a برابر باقیماندهٔ تقسیم a بر b خواهد شد.

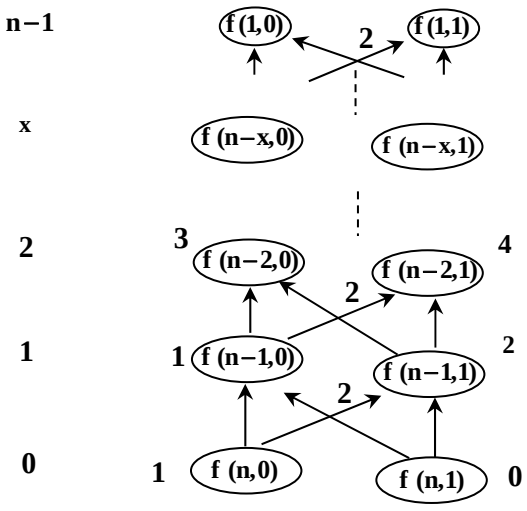
**۹۵ـ گزینه «۲»** ابتدا ضابطهٔ ریاضی تابع را به دست می‌آوریم، داریم:

$$recursive(n) = \begin{cases} 1 & \text{اگر } n = 1 \\ recursive(n-1) + (n-3) & \text{در غیر این صورت} \end{cases}$$

با استفاده از ضابطهٔ ریاضی داریم:

$$recursive(5) = recursive(4) + (5-3) = 1 + 2 = 3$$
$$recursive(4) = recursive(3) + (4-3) = 0 + 1 = 1$$
$$recursive(3) = recursive(2) + (3-3) = 0 + 0 = 0$$
$$recursive(2) = recursive(1) + (2-3) = 1 + (-1) = 0$$
$$recursive(1) = 1$$

**۹۶ـ گزینه «۳»** نحوه محاسبه $F(1,1)$ از رابطه بازگشتی در شکل زیر مشخص است:




www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



منظور سؤال از محاسبه $F(1,1)$ به صورت کارا، محاسبه آن به صورت معکوس است و همان طور که در شکل فوق مشاهده می‌کنید با توجه به داشتن $F(n,0)$ و $F(n,1)$، ابتدا $F(n-1,0)$ و $F(n-1,1)$ محاسبه می‌شود، سپس $F(n-2,0)$ و $F(n-2,1)$ و به همین ترتیب، تا زمانی که $F(1,1)$ به دست آید. به عبارت دیگر:

$$F(n,0)+F(n,1)\to F(n-1,0)$$
$$2F(n,0)+F(n,1)\to F(n-1,1)$$
$$F(n-1,0)+F(n-1,1)\to F(n-2,0)$$
$$2F(n-1,0)+F(n-1,1)\to F(n-2,1)$$
$$\vdots$$
$$F(n-x,0)+F(n-x,1)\to F(n-x-1,0)$$
$$2F(n-x,0)+F(n-x,1)\to F(n-x-1,1)$$
$$\vdots$$
$$2F(n-(n-2),0)+F(n-(n-2),1)\to F(n-(n-1),1)=F(1,1)$$

لذا، تعداد دقیق عمل جمع برابر $2(n-1)-1=2n-3$ است. بنابراین گزینه (۳) صحیح است.

---

**۹۷ـ گزینه «۱»** با توجه به صورت مسئله، این خرس قطبی در $n$ روز $\frac{n}{2}$ از قطعه گوشت‌ها را می‌خورد و $\frac{n}{2}$ دیگر را نصف کرده و می‌خورد. در $\frac{n}{2}$ روز بعدی $\frac{n}{4}$ را می‌خورد و $\frac{n}{4}$ دیگر را نصف کرده و می‌خورد و با این وضعیت روزهای باقیمانده از عمر خرس عبارت است از:

$$n+\frac{n}{2}+\frac{n}{4}+\frac{n}{8}+\frac{n}{16}+\cdots=n(1+\frac{1}{2}+\frac{1}{4}+\frac{1}{8}+\frac{1}{16}+\cdots)\simeq 2n\in\theta(n)$$

---

**۹۸ـ گزینه «۳»** اگر از صورت کلی قضیهٔ اصلی (Master theorem) استفاده کنیم، آنگاه به این دلیل که تابع $n^2\log n$ و تابع $n^{\log_2^4}=n^2$ به اندازهٔ توانی از $n$ اختلاف ندارند جواب به صورت مقابل بدست می‌آید:

$$f(n)\in\theta(n^2\log n\times\log n)=\theta(n^2(\log n)^2)$$

---

**۹۹ـ گزینه «۱»** پیچیدگی زمانی یک چندجمله‌ای از مرتبه جمله با بزرگ‌ترین توان می‌باشد (البته گزینه دوم نیز می‌تواند درست باشد).

---

**۱۰۰ـ گزینه «۱»** با استفاده از روش جایگذاری می‌توان نوشت:

$$T(n)=T(n-1)+\frac{2}{n}$$
$$\Rightarrow T(n)=T(n-2)+\frac{2}{n-1}+\frac{2}{n}$$
$$\Rightarrow T(n)=T(n-3)+\frac{2}{n-2}+\frac{2}{n-1}+\frac{2}{n}$$
$$\vdots$$
$$T(n)=T(1)+\frac{2}{2}+\frac{3}{2}+\ldots+\frac{2}{n-2}+\frac{2}{n-1}+\frac{2}{n}$$

بنابراین:

$$T(n)=2(\frac{1}{n}+\frac{1}{n-1}+\frac{1}{n-2}+\ldots+\frac{1}{2})\approx 2\ln n$$

---

**۱۰۱ـ گزینه «۲»** با استفاده از ضابطه ریاضی تابع داریم: $fun(n,m)=\begin{cases}n & m=2 \text{ اگر}\\ n*fun(n,m-1) & \text{در غیر این صورت}\end{cases}$ و با محاسبه $fun(n,4)$ داریم:

$$fun(n,4)=n*fun(n,3)=n^3$$
$$fun(n,3)=n*fun(n,2)=n^2$$
$$fun(n,2)=n$$

که نتیجه می‌دهد: $fun(n,m)=n^{m-1}$

---

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۱۰۲ـ گزینه «۱»** با استفاده از ضابطه ریاضی تابع، داریم:

$$fun(a)=\begin{cases} a*a-5 & a<=1 \quad \text{اگر} \\ 3*fun(a-2)+4 & \text{در غیر این صورت} \end{cases}$$

$$fun(8)=3*fun(6)+4=3*(-83)+4=-249+4=-245$$
$$fun(6)=3*fun(4)+4=3*(-29)+4=-87+4=-83$$
$$fun(4)=3*fun(2)+4=3*(-11)+4=-29$$
$$fun(2)=3*fun(0)+4=3*(-5)+4=-11$$
$$fun(0)=0*0-5=-5$$

---

**۱۰۳ـ گزینه «۲»** در این تابع اگر $x<0$ باشد، دوباره تابع با $-x$ فراخوانی می‌شود که در این صورت تابع با مقدار مثبت x فراخوانی خواهد شد و اگر $x<10$ باشد خود x برگشت داده می‌شود. در غیر این صورت باقیمانده x بر 10 برگشت داده می‌شود یا به عبارت دیگر رقم یکان عدد را بر می‌گرداند.

---

**۱۰۴ـ گزینه «۴»** به ازاء $C=200$ و $n_0=10$ داریم: $(\forall n>10)$ $2^n+100n^3+n\log_2^n\le 200\ 2^n$

و این نشان می‌دهد که: $2^n+100n^3+n\text{Log}_2^n\in O(2^n)$

---

**۱۰۵ـ گزینه «۲»** با توجه به معادلهٔ بازگشتی ذکر شده داریم:

$$T^2(n)=\frac{1}{2}(T^2(n-1)+T^2(n-2))+n \Rightarrow T^2(n)-\frac{1}{2}T^2(n-1)-\frac{1}{2}T^2(n-2)=n$$
$$\Rightarrow 2T^2(n)-T^2(n-1)-T^2(n-2)=2n$$

که این معادلهٔ بازگشتی یک معادلهٔ غیر همگن بوده و برای حل آن ابتدا معادلهٔ همگن $2T^2(n)-T^2(n-1)-T^2(n-2)=0$ را حل می‌کنیم.

با تغییر متغیر $T^2(n)=M(n)$ داریم: $2M(n)-M(n-1)-M(n-2)=0$

که معادلهٔ شاخص آن عبارت است از: $2r^2-r-1=0$

و با توجه به طرف مقابل معادلهٔ غیر همگن می‌توانیم نتیجه بگیریم: $T^2(n)=\theta(n^2)$

---

**۱۰۶ـ هیچ‌کدام از گزینه‌ها صحیح نیست.** با توجه به تابع داده شده تعداد ستاره طبق رابطه بازگشتی زیر به دست می‌آید:

$$T(n)=2T(\frac{n}{2})+1\ ,\ T(0)=1$$

بنابراین خواهیم داشت: $T(2)=2T(1)+1\ ,\ T(1)=2T(0)+1=3 \Rightarrow T(2)=7$

در نتیجه رابطه باید به صورت $T(n)=2^{\left\lfloor \log_2^x \right\rfloor+2}-1$ باشد.

---

**۱۰۷ـ گزینه «۴»** در این تابع بازگشتی تا زمانی که $a\ge b$ باشد از پارامتر a، b واحد کم شده و دوباره تابع در خودش فراخوانی می‌شود، تا زمانی که $a<b$ شود که در این صورت مقدار a به عنوان مقدار نهایی تابع برگشت داده می‌شود و با توجه به مطالب بالا می‌توان نتیجه گرفت که این تابع برابر باقی‌ماندهٔ a بر b است، البته به راحتی می‌توان این مطلب را با یک مثال نشان داد، داریم:

$$T(7,4)=T(7-4,4)=T(3,4)=3$$

در گزینه یک: $7\ \text{div}\ 4=1$ است و در گزینهٔ ۲: $7\bmod(4-1)=1$ و در گزینه (۳): $(7-1)\bmod 4=2$ که هیچ‌کدام از گزینه‌ها جواب صحیح را نتیجه نمی‌دهد و تنها گزینه (۴) صحیح است، یعنی: $7\bmod 4=3$

---

**۱۰۸ـ گزینه «۳»** باید داشته باشیم $F(2)=3!=6$ بنابراین: $F(2)=2F(1)+2!=2F(1)+2 \Rightarrow F(1)=2$

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



# فصل دوم

# «آنالیز سرشکن»

## تست‌های طبقه‌بندی شده کنکوری فصل دوم

۱ـ بر روی پشته S اعمال زیر تعریف شده‌اند: (مهندسی کامپیوتر ـ سراسری۷۸)

**Push(S,x)** : درج x در بالای پشته با هزینه **O(1)**

**Pop(S,x)** : حذف عنصر بالای پشته با هزینه **O(1)**

**Multipop(S,K)** : حذف K عنصر بالای پشته (با فرض آن که K حداکثر برابر تعداد عناصر موجود S است) با هزینه **O(K)**

اگر N عمل از اعمال فوق به ترتیب دلخواه، بر روی پشته S که در ابتدا تهی است انجام شود، مجموع هزینه این N عمل در بدترین حالت چقدر است؟

۱) $O(N \log N)$ ۲) $O(N^2)$ ۳) $O(NK)$ ۴) $O(N)$

۲ـ یک صف به صورت زیر توسط دو پشته $S_1$ و $S_2$ پیاده‌سازی شده است. اگر تابع پتانسیل به صورت $\phi(Q_k) = |S_1|$ تعریف شود که $|S_1|$ عناصر پشته $S_1$ و $Q_k$ صف پس از اجرای K امین عملگر می‌باشد. هزینه عملگر **del Front( )** برابر است با: (مهندسی IT ـ سراسری ۸۷)

```
Algorithm del Front ( )
    if not S2.empty ( ) then
        return  S2.pop ( )
    end if
    while not S1. empty ( ) do
        S2.push (S1.pop( ))
    end while
    Return S2.pop ( )
end del Front
```

```
Algorithm add Rear (x)
    S1. Push (x)
end add Rear
```

۱) 1

۲) 2

۳) 3

۴) هیچ‌کدام

۳ـ دنباله‌ای از $2^n$ عمل بر روی داده ساختاری انجام می‌شود. هزینه‌ی عمل i ام برابر i است. اگر i توانی از ۲ باشد، وگرنه برابر 1 است. میانگین هزینه یک عمل دلخواه (یعنی مجموع هزینه‌ها تقسیم بر تعدادشان) به کدام گزینه نزدیکتر است؟ (مهندسی کامپیوتر ـ سراسری ۸۹)

۱) 2 ۲) n ۳) 3 ۴) 2n

## پاسخنامه تست‌های طبقه‌بندی شده کنکوری فصل دوم

۱ـ گزینه «۴» $pop(s,x), push(s,x)$ دارای هزینهٔ $O(1)$ است، پس n عمل این دو دارای هزینهٔ $O(n)$ می‌باشد. $multipop(s,k)$ دارای هزینهٔ $O(k)$ است و با توجه به خالی بودن پشته در ابتدای اجرا، این عمل زمانی میسر است که k عمل push انجام شده باشد که در بدترین حالت $k = n-1$ خواهد شد. هزینه سرشکن شده این عملیات $O(N)$ خواهد بود.

---

۲ـ گزینه «۱» الگوریتم داده شده در صورتی که پشته S2 تهی نباشد عنصر بالای پشته را حذف می‌کند و به عنوان خروجی بر می‌گرداند. سپس تمامی عناصر پشته S1 را در پشته S2 قرار می‌دهد که در نهایت باعث کاهش پتانسیل به اندازه یک واحد خواهد شد.

---

۳ـ گزینه «۳» هزینه $2^n$ عمل متوالی را می‌توان به صورت زیر محاسبه نمود:

$$\sum_{i=0}^{n} 2^i + \underbrace{2^n - n}_{\text{هزینه سایر اعمال}} = 2^{n+1} - 1 + 2^n - n \approx 3 \times 2^n$$

بنابراین هزینه میانگین هر عمل $\frac{3 \times 2^n}{2^n} = 3$ می‌باشد.

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



# فصل سوم

## «مقدمه‌ای بر پیچیدگی محاسباتی (Computational Complexity)»

### تست‌های طبقه‌بندی شده کنکوری فصل سوم

**۱ـ اگر یک مسئله NP-complete مانند L وجود داشته باشد که $L \in P$ باشد در آن صورت:** (علوم کامپیوتر ـ سراسری ۸۲)

۱) $P = NP$ ۲) $P \neq NP$ ۳) $L \in NP - hard$ ۴) $L \notin NP - hard$

**۲ـ کدام الگوریتم دارای پیچیدگی نمایی نمی‌باشد؟** (علوم کامپیوتر ـ سراسری ۸۲)

۱) پیدا کردن دور هامیلتونی در یک گراف
۲) پیدا کردن طول زیر رشته مشترک بین دو رشته
۳) مسئله کوله‌پشتی 1,0
۴) الگوریتم بازگشتی فیبوناچی

**۳ـ $G = (V, E)$ یک گراف بدون جهت و همبند و وزن‌دار است و a و b دو رأس مجزا در آن. فرض کنید $P_1$ مسئله پیداکردن کوتاهترین مسیر بین a و b و $P_2$ مسئله پیداکردن بلندترین مسیر ساده بین a و b باشد، کدامیک از گزینه‌های زیر در مورد $P_1$ و $P_2$ درست است؟** (مهندسی کامپیوتر ـ سراسری ۸۳)

۱) $P_1$ و $P_2$ را می‌توان در زمان چند جمله‌ای حل کرد.
۲) $P_1$ و $P_2$ را نمی‌توان در زمان چند جمله‌ای حل کرد.
۳) $P_1$ را می‌توان در زمان چند جمله‌ای حل کرد ولی $P_2$ را نمی‌توان.
۴) $P_2$ را می‌توان در زمان چند جمله‌ای حل کرد ولی $P_1$ را نمی‌توان.

**۴ـ کدام گزینه نادرست است؟** (علوم کامپیوتر ـ سراسری ۸۳)

۱) تمام مسائل P به وسیله یک الگوریتم غیرقطعی در زمان چندجمله‌ای حل می‌شوند.
۲) تمام مسائل NP به وسیله یک الگوریتم غیرقطعی در زمان چندجمله‌ای حل می‌شوند.
۳) تمام مسائل NP-hard به وسیله یک الگوریتم غیرقطعی در زمان چندجمله‌ای حل می‌شوند.
۴) تمام مسائل NP-complete به وسیله یک الگوریتم غیرقطعی در زمان چندجمله‌ای حل می‌شوند.

**۵ـ فرض کنید $P \neq NP$ است. کدام یک از مسائل زیر در زمان چندجمله‌ای حل می‌شود؟** (علوم کامپیوتر ـ سراسری ۸۶)

**A**- یافتن طولانی‌ترین مسیر ساده در یک گراف غیرجهت‌دار
**B**- یافتن کوتاه‌ترین مسیر ساده در یک گراف غیر جهت‌دار
**C**- یافتن تمام درختان فراگیر (Spanning) در یک گراف غیر جهت‌دار

۱) B ۲) A و B ۳) B و C ۴) A و B و C

**۶ـ کدام گزاره نادرست است؟** (علوم کامپیوتر ـ سراسری ۸۷)

۱) $P = NP \cap CO - NP$
۲) $P \subseteq NP$
۳) $P \subseteq CO - NP$
۴) $if\ NP \neq CO - NP \Rightarrow P \neq NP$

**۷ـ پیچیدگی زمان الگوریتم محاسبه عبارات true برای یک عبارت منطقی (satisfiability) با n متغیر برابر است با:** (علوم کامپیوتر ـ سراسری ۸۷)

۱) $O(2^n)$ ۲) $O(n2^n)$ ۳) $O(n^k)$ ۴) $O(82^n)$

**۸ـ کدام گزینه نادرست است؟** (علوم کامپیوتر ـ سراسری ۸۸)

۱) $P \subset NP - Complete$
۲) $A \leq_p B \Rightarrow A \in NP$
۳) $NP - Complete \subset NP - hard$
۴) برای $A \leq_p B$ داریم: اگر B در زمان چندجمله‌ای حل شود A نیز حل می‌شود.

**۹ـ مسئله زیر را در نظر بگیرید. کدام گزینه نادرست است؟** (علوم کامپیوتر ـ سراسری ۸۸)

$$\sum_{1 \leq i \leq n} x_i w_i \leq M$$

$$\max \sum_{1 \leq i \leq n} x_i p_i$$

$$s.t \qquad x_i = \begin{cases} 0 \\ 1 \end{cases}$$

۱) این مسئله یک مسئله NP نمی‌باشد.
۲) این مسئله یک مسئله NP-hard است.
۳) این مسئله راه‌حل چندجمله‌ای برحسب هیچ پارامتری ندارد.
۴) این مسئله یک راه‌حل چندجمله‌ای برحسب n و M دارد.

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۱۰ـ پیچیدگی الگوریتم غیرقطعی برای مسئله k-clique در یک گراف $G=(V,E)$ برابر است با:** (علوم کامپیوتر ـ سراسری ۸۹)

۱) $O(k^2)$ ۲) $O(|V|^2 k)$ ۳) $O(k+|V|^2)$ ۴) $O(|V|+|E|)$

**۱۱ـ کدام یک از مسائل زیر NP می‌باشد؟** (علوم کامپیوتر ـ سراسری ۸۹)

۱) Minimum node cover ۲) Max clique ۳) $3-sat$ ۴) همه موارد

**۱۲ـ الگوریتم‌هایی که مانند الگوریتم مسئله فروشنده دوره‌گرد، دارای الگوریتمی کارا نمی‌باشند و عدم امکان یک الگوریتم کارا نیز برای آن‌ها اثبات نگردیده است، چه نامیده می‌شوند؟** (مهندسی کامپیوتر ـ آزاد ۸۹)

۱) $NP-$ ناقص ۲) $NP-$ توسعه‌پذیر

۳) دارای چند جمله‌ای زمانی $P(n)$ معین نمی‌باشند. ۴) $NP-$ کامل

## پاسخنامه تست‌های طبقه‌بندی شده کنکوری فصل سوم

**۱ـ گزینه «۱»** اگر یک مسئله NP-complete مانند L وجود داشته باشد که $L \in P$ باشد در آن صورت می‌توان گفت مجموعه P و NP با یکدیگر برابر می‌باشند. (در این صورت یکی از بزرگترین اتفاق‌های علمی در علوم کامپیوتر به وقوع می‌پیوندد!)

**۲ـ گزینه «۲»** گزینه ۱ همان مسئله معروف فروشنده دوره‌گرد در حالتی خاص است. گزینه ۳ هم از مسائل نمایی است. الگوریتم فیبوناچی می‌تواند به صورت خطی پیاده شود اما الگوریتم بازگشتی آن نمایی است. تنها گزینه ۲ می‌تواند با برنامه‌سازی پویا پیچیدگی کمتر از نمایی داشته باشد.

**۳ـ گزینه «۳»** مسئله کوتاه‌ترین مسیر بین یک جفت رأس در زمان چندجمله‌ای توسط الگوریتمی مانند بلمن فورد قابل محاسبه است. اما مسئله یافتن بلندترین مسیر بین یک جفت رأس یک مسئله $NP-hard$ می‌باشد (البته دقت کنید که تاکنون الگوریتمی با زمان چندجمله‌ای برای آن ایجاد نشده اما عدم امکان وجود آن نیز اثبات نشده است).

**۴ـ گزینه «۳»** مسائل P با یک الگوریتم قطعی در زمان چندجمله‌ای حل می‌شوند (چه رسد به غیرقطعی) و مسائل NP با الگوریتم‌های غیرقطعی در زمان چندجمله‌ای قابل حل هستند، اما مسائل NP-hard اینگونه نیستند. مسائل NP-Complete نوع خاصی از NP هستند و ادعای بیان شده برای آنها صادق است.

**۵ـ گزینه «۱»** A، C هر دو دارای پیچیدگی نمایی و حتی بدتر از آن هستند.

**۶ـ گزینه «۱»** صورت صحیح گزینه اول به شکل مقابل است: $P \subseteq NP \bigcap CO-NP$

**۷ـ گزینه «۲»** کل حالت‌ها $2^n = \underbrace{2\times 2\times 2\times \ldots \times 2}_{n}$ بوده و برای هر حالت به $O(n)$ برای ارزیابی عبارات نیاز داریم.

**۸ـ گزینه «۲»** گزینه‌های ۱، ۳و۴ گزاره‌های درستی را بیان می‌کنند. اما گزینه ۲ صحیح نمی‌باشد، زیرا از کاهش‌پذیر بودن A به B نمی‌توان نتیجه گرفت که A در NP می‌باشد. البته درست یا نادرست بودن گزینه قابل تعیین نیست.

**۹ـ گزینه «۲»** مسئله موردنظر همان مسئله کوله‌پشتی صفر و یک است.

**۱۰ـ گزینه «۱»** پیچیدگی الگوریتم غیرقطعی برای مسئله k-clique در یک گراف برابر $O(k^2)$ می‌باشد. (کلیک زیر گراف کامل در یک گراف می‌باشد).

**۱۱ـ گزینه «۴»** صورت تصمیم‌گیری تمام مسائل داده شده NP می‌باشد.

**۱۲ـ گزینه «۴»** این نوع مسائل NPC می‌باشند.

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir

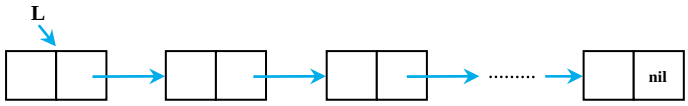


# فصل چهارم

## «داده‌ساختارهای مقدماتی»

### تست‌های طبقه‌بندی شده کنکوری فصل چهارم

**۱ـ کدام یک از توابع زیر آدرس آخرین گره یک لیست پیوندی یک طرفه L را بر می‌گرداند؟** (مهندسی کامپیوتر ـ سراسری۷۵)




۱)
```
function Lastnode (L);
    begin
      if L ≠ nil and Link (L) ≠ nil then
        Lastnode := Lastnode (Link(L))
      else
        Lastnode := L
    end;
```

۲)
```
function Lastnode (L);
    begin
      if Link(L) ≠ nil then
        Lastnode := Lastnode (Link(L))
      else
        Lastnode := L
    end;
```

۳)
```
function Lastnode (L);
    begin
      if L ≠ nil then
        if Link (Link(L)) ≠ nil then
          Lastnode := Lastnode (Link(L))
        else
          Lastnode := L
      else
        Lastnode := nil
    end;
```

۴)
```
function Lastnode (L);
    begin
      if L ≠ nil and Link (L) ≠ nil then
        Lastnode := Lastnode (Link(Link(L)))
      else
        Lastnode := L
    end;
```

**۲ـ تابع زیر چه عملی انجام می‌دهد؟ توضیح اینکه LIST نشان‌دهنده لیستی از اعداد بوده و منظور از تابع Head تابعی است کـه مقـدار اولـین عنصر در لیست را بر می‌گرداند و تابع Tail لیستی حاوی همه عناصر لیست ورودی به استثنای اولین عنصر را بر می‌گرداند. مکان‌های لیست را از مکان 1 در نظر بگیرید:** (مهندسی کامپیوتر ـ سراسری۷۶)

```
function what (L:LIST):integer;
   begin
        if L=nil then
           what : = (0)
        else
nil then              ≠              if Tail (L)
              what :=(Head (L) + what (Tail (Tail(L))))
           else
              what :=Head (L)
     end;
```

۱) تعداد عناصر لیست را بر می‌گرداند.

۲) مجموع عناصر در مکان‌های فرد لیست ورودی را برمی‌گرداند.

۳) تعداد عناصر در مکان‌های فرد لیست ورودی را برمی‌گرداند.

۴) مجموع عناصر لیست ورودی را بر می‌گرداند.

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



۳ـ یک **Stack** و یک صف با عملیات زیر مفروض است: (مهندسی کامپیوتر ـ سراسری ۷۶)

الف ـ عملیات **Stack** :

**Push(x)**: مقدار **x** را وارد **Stack** می‌کند.

**Pop(x)**: مقدار بالای **Stack** را در **x** می‌گذارد.

ب ـ عملیات صف:

**Enqueue(x)**: مقدار **x** را در انتهای صف می‌گذارد.

**Dequeue(x)**: مقدار سر صف را در **x** می‌گذارد.

فرض کنید در ابتدا **Stack** حاوی **3** عدد صحیح متفاوت به ترتیب دلخواه است (مثلاً **3** و **1** و **2**). اگر در پایان انجام عملیات زیر، **Stack** حاوی همان **3** عدد به ترتیب صعودی از پایین به بالا (مثلاً **3** و **2** و **1**) باشد، چند ترتیب اولیه در **Stack** برای **3** عدد مزبور ممکن است؟

**Pop(x); Enqueue(x); Pop(x); Enqueue(x); Dequeue(x); Push(x); Pop(x); Enqueue(x);**

**Pop(x); Enqueue(x); Dequeue(x); Push(x); Dequeue(x); Push(x); Dequeue(x); Push(x);**

۱) 3 ۲) 2 ۳) 4 ۴) 1

۴ـ رشتهٔ $S = a_1a_2a_3\cdots a_n$ در آرایهٔ **A[1..n]** ذخیره شده است. کدام یک از توابع زیر تعیین می‌کند که رشتهٔ **S** یک **Palindrome** می‌باشد؟ رشتهٔ **S** یک **Palindrome** است، اگر از چپ و راست به یک صورت خوانده شود. مثلاً **radar** یک رشته **Palindrome** است. (مهندسی کامپیوتر ـ سراسری ۷۷)

۱)
```
Function  Palindrome (A, start, end) : Boolean;
 begin
      if  start ≥ end then Palindrome:= true
     else if  A[start] = A[end] and Palindrome (A, start + 1, end − 1) then Palindrome := true
    else Palindrome :=false
end ;
```

۲)
```
Function  Palindrome (A, start, end) : Boolean;
  begin
      if  start = end then Palindrome:= true
     else if  A[start] = A[end] and Palindrome (A, start + 1, end − 1) then
                    Palindrome := true
            else
                    Palindrome := false
end;
```

۳)
```
Function  Palindrome (A, start, end) : Boolean;
   begin
      if  start ≥ end Palindrome:= true
     else if  Palindrome (A, start + 1, end − 1) then  Palindrome := true
             else
       Palindrome := false
end;
```

۴)
```
Function  Palindrome (A, start, end) : Boolean;
  begin
      if  start = end then Palindrome:= true
     else if  start = end and Palindrome (A, start + 1, end − 1) then
           Palindrome := true
      else
            Palindrome := false
end;
```

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



۵ـ عبارت پیشوندی (prefix) مقابل داده شده است: $++a/b-cd/-ab-+c*d5/a-bc$

معادل پسوندی (postfix) آن کدام است؟ (* علامت ضرب تلقی می‌شود.) (مهندسی کامپیوتر ـ سراسری ۷۷)

۱) $abcd-/+ab-cd5*+abc-/-/+$
۲) $ab+cd-/ab-cd+/5*+ab/c--+$
۳) $ab+cd-ab-+/cd5*+/abc-/-$
۴) $abcd/-+abc-d5*abc-+/-/+$

۶ـ آرایه $M(10\times20\times30)$ را که هر عنصر آن 4 بایت حافظه نیاز دارد به صورت ستونی (Column Major) از بایت 20/000 (مبنای 10) در حافظه ذخیره می‌نماییم. (یعنی به ترتیب 10 ماتریس $20\times30$ که هر ماتریس خود به صورت ستونی ذخیره شده است.) در این صورت، آدرس خانه $M(21,11,9)$ از چه بایتی شروع می‌شود؟ (مهندسی کامپیوتر ـ سراسری ۷۸)

۱) 20480 ۲) 40480 ۳) 40840 ۴) 20840

۷ـ عناصر صف‌های $Q_1$ و $Q_2$ از چپ به راست به صورت زیر است (عنصر سمت چپ ابتدای صف است): (مهندسی کامپیوتر ـ سراسری ۷۹)

$Q_1 = 10,25,17,41,19,26,75 \quad Q_2 = 1,5,7,4,9,6$

اگر x و y عناصر صف باشند، پس از اجرای قطعه برنامه زیر:

```
makenull(Q3)
i=0
while (not empty(Q1) and not empty(Q2))
    i = i+1
    x = DeleteQ(Q1)
    y = DeleteQ(Q2)
    if (y = i) then AddQ(Q3,x)
endwhile
```

محتوی صف $Q_3$ برابر است با:

۱) $Q_3 = 1,4,6$ ۲) $Q_3 = 10,25,17$ ۳) $Q_3 = 10,41,26$ ۴) $Q_3 = 1,5,7$

۸ـ آنچه در زیر آمده است یک شبه کد اشتباه برای الگوریتمی است که باید متوازن بودن یا نبودن رشته‌ای از پرانتزها را تعیین کند:

```
declare a character stack
while (more input is available)
   {read a character
      if (the character is a '(' )
      then push it on the stack
      else if (the character is a ')' and the stack is not empty)
           then pop a character of the stack
           else print "unbalanced" and exit
      }
print "balanced"
```

کدام یک از رشته‌های نامتوازن زیر توسط الگوریتم فوق، متوازن در نظر گرفته می‌شوند؟ (مهندسی کامپیوتر ـ سراسری ۸۰)

۱) ((()) ۲) ()) (() ۳) ((())) () ۴) (() ()))

۹ـ یک لیست خطی یک‌طرفه با دو اشاره‌گر F و R که به ترتیب به عنصر اول و آخر لیست اشاره می‌کنند پیاده‌سازی شده است. هزینه کدام یک از عملیات زیر وابسته به تعداد عناصر لیست است؟ (مهندسی کامپیوتر ـ سراسری ۸۰)

۱) حذف اولین عنصر ۲) حذف آخرین عنصر ۳) درج یک عنصر در انتهای لیست ۴) درج یک عنصر در ابتدای لیست

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



۱۰ـ اگر سه اشاره‌گر $P_3, P_2, P_1$ روی لیست L حرکت کنند و سه تابع First و Next و end مقدار منطقی متناسب را بازگردانند، تعداد دفعات اجرای تابع First بر حسب n (تعداد عناصر لیست) چیست؟ (مهندسی کامپیوتر ـ سراسری ۸۱)

```
P1 := First(L);
while P1 < > end(L) do
begin
        P2 = P1;
        while P2 < > end(L) do
        begin
                P2 := Next(P2,L);
                P3 := First(L);
                While P3 < > P2  do      P3 : = Next(P3,L)
        end;
        P1 := Next(P1,L);
end;
```

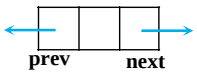
۱) $\frac{(n-1)^2}{2}+1$

۲) $(n-1)^2+1$

۳) $\frac{n}{2}(n-1)+1$

۴) $\frac{n}{2}(n+1)$

۱۱ـ درج یک عنصر در یک لیست پیوندی دو طرفه چه تعداد از اشاره‌گرهای next و prev را تغییر می‌دهد؟ (علوم کامپیوتر ـ سراسری ۸۱)




۱) دو عدد next و دو عدد prev

۲) یک عدد next و یک عدد prev

۳) یک عدد next و دو عدد prev

۴) دو عدد next و یک عدد prev

۱۲ـ اگر تخصیص حافظه برای آرایه‌ها سطری باشد، در آن صورت برای آرایهٔ سه بعدی $A[1..n_1, 1..n_2, 1..n_3]$ مؤلفه A[i, j, k] در چه محلی از حافظه قرار می‌گیرد به طوری که α آدرس اولین خانهٔ حافظهٔ آرایه باشد؟ (علوم کامپیوتر ـ سراسری ۸۱)

۱) $\alpha+(i-1)n_1n_2+(j-1)n_2+(k-1)$

۲) $\alpha+(i-1)n_2n_3+(j-1)n_3+(k-1)$

۳) $\alpha+(k-1)n_1n_2+(j-1)n_2+(i-1)$

۴) $\alpha+(k-1)n_2n_3+(j-1)n_3+(i-1)$

۱۳ـ عبارت $(a/(b-c+d))*(e-a)*c$ داده شده است. در این صورت نشانه‌گذاری postfix این عبارت به صورت ..... است. (علوم کامپیوتر ـ سراسری ۸۱)

۱) $ab/c-de*+ac*-$

۲) $-+-/abc*de*ae$

۳) $abc-d+/ea-*c*$

۴) هیچ‌کدام

۱۴ـ می‌خواهیم ساختار داده‌ای مشابه پشته برای حداکثر n عدد طراحی کنیم تا اعمال Push، Pop، Find Max و Find Min را بتواند به صورت کارا انجام دهد. برای این کار از دو آرایه A[1..n] و B[1..n] استفاده می‌کنیم که با انجام اعمال فوق مقادیرشان عوض می‌شود. A[i] عنصر iام پشته است و B[i] اندیس کوچک‌ترین عنصر بین A[1] تا A[i]. با استفاده از این آرایه‌ها، اعمال فوق را با چه هزینه‌ای می‌توان انجام داد؟ (مهندسی کامپیوتر ـ سراسری ۸۳)

۱) همهٔ اعمال $O(1)$

۲) همهٔ اعمال $O(\lg n)$

۳) همهٔ اعمال به جز Find Max $O(1)$ است.

۴) همهٔ اعمال به جز Pop و Find Max $O(1)$ است.

۱۵ـ تعریف آرایه Long L[10][20][30] را در زبان C در نظر بگیرید، اگر هر خانه از نوع long چهار بایت حافظه را اشغال کند و این آرایه از آدرس واقعی 10000 در مبنای ده شروع شود و تخصیص حافظه به صورت سطری (row major) باشد، برای دسترسی به عنصر L[5][10][20] این آرایه، آدرس اولین بایت مورد دسترسی توسط کدام گزینه مشخص شده است؟ (مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۳)

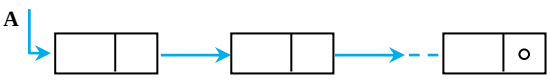
۱) 13280

۲) 23280

۳) 3320

۴) 13320

۱۶ـ فرض کنید بخواهیم ترتیب اشاره‌گرها را در لیست تک پیوندی A وارون نماییم.




به عبارت دیگر هر اشاره‌گر به جای اشاره به عنصر بعدی، به عنصر قبلی اشاره نماید. برای انجام این کار، کدام یک از گزاره‌های ذیل درست است. (فرض کنید الگوریتم غیر بازگشتی باشد). (مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۳)

۱) ۳ اشاره‌گر به جز نام لیست مورد نیاز است.

۲) ۲ اشاره‌گر به جز نام لیست مورد نیاز است.

۳) اشاره‌گر دیگری غیر از A، مورد نیاز نیست.

۴) کافی است از یک استک استفاده کنیم، لذا اشاره‌گر بالای استک و نام لیست، مورد نیاز است.

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۱۷ـ زمان اجرای الگوریتم زیر چقدر است؟P: یک آرایه n عنصری از اعداد که Stack Pointer را در یک Stack نشان می‌دهد و خروجی (S) یک آرایه n عنصری از اعداد است.** (مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۴)

```
Test (P)
    D is an empty stack
    for k=0 to n-1 do
        done=False
        while not (D.is Empty( )or done) do
            if P(K)>P(D.top( ))then
                D.pop ( )
            else
                Done=True
        end while
        if D.isEmpty( ) then h= -1
        else h← D.top( )
        end if
        S[k]=k-h
        D.push(k)
        end for
    return S
```

۱) $O(n^2)$
۲) $O(n)$
۳) $O(\log n)$
۴) $O(n \log n)$

**۱۸ـ عبارت محاسباتی میانوندی $Q : A+(B*C-(D/E\hat{}F)*G)*H$ را در نظر بگیرید، عبارت پسوندی (Postfix) معادل آن عبارت است از:** (مهندسی کامپیوتر ـ آزاد ۸۴)

۱) $ABC*DEF\hat{}/G*-H*+$
۲) $ABC*D\hat{}E*F/G-H*+$
۳) $ABC*DE*F\hat{}/G-H*+$
۴) هیچ‌کدام

**۱۹ـ یک پشته خالی و اعداد 1 تا 6 در ورودی داده شده است. اعمال زیر بر روی پشته قابل انجام هستند:** (مهندسی کامپیوتر ـ آزاد ۸۴)

**کوچک‌ترین عدد ورودی را برداشته و وارد پشته می‌کند. (push)**

**عنصر بالای پشته را در خروجی گذاشته و سپس آن را حذف می‌کند. (pop)**

**کدام یک از گزینه‌های زیر را با هیچ ترتیبی نمی‌توان از اعمال فوق به دست آورد؟ (اعداد را از چپ به راست بخوانید)**

۱) 123564
۲) 324651
۳) 215346
۴) 432165

**۲۰ـ عبارت محاسباتی میانوندی $Q : A+(B*C-(D/E\hat{}F)*G)*H$ را در نظر بگیرید، عبارت پسوندی (Postfix) معادل آن عبارت است با:** (مهندسی فناوری اطلاعات (IT) ـ آزاد ۸۴)

۱) $ABC*DEF\hat{}/G*-H*+$
۲) $ABC*D\hat{}E*F/G-H*+$
۳) $ABC*DE*F\hat{}/G-H*+$
۴) هیچ‌کدام

**۲۱ـ کدام راه برای نمایش ذخیره ماتریس‌های بالا مثلثی مناسب‌تر است؟** (مهندسی فناوری اطلاعات (IT) ـ آزاد ۸۴)

۱) به صورت یک ماتریس خلوت با استفاده از آرایه $n*3$ زیرا نیمی از عضوهای آن صفر است.
۲) به صورت یک ماتریس عادی زیرا نیمی از عضوهای آن صفر نیست و ضریب 1.2 برای خلوت بودن کافی نیست.
۳) به صورت یک آرایه یک بعدی که عضوهای غیر صفر را سطر به سطر در خود جای می‌دهد به علاوه پیدا کردن فرمولی برای محل هر عضو.
۴) به صورت لیست پیوندی به طوری که هر عضو بر سطر و ستون و مقدار، دارای پیوندهای سطری و ستونی نیز باشد.

**۲۲ـ اگر آرایه A به صورت int A[20][10][5] تعریف شده باشد (یعنی محدوده اندیس‌ها به صورت A[0..19][0..9][0..4] باشد). با فرض این که هر عدد int چهار بایت لازم دارد و آدرس شروع آرایه صفر باشد، آدرس عنصر A[3][4][2] در حالت ستون محور (column major) عبارت است از:** (مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۵)

۱) 1934
۲) 1932
۳) 482
۴) 483

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



۲۳ـ اگر آرایهٔ $A[\ell_1...u_1, \ell_2...u_2]$ را به صورت ستونی در یک آرایهٔ یک بعدی ذخیره نماییم فرمول ذخیره‌سازی چیست؟ (به فرض $\alpha$ آدرس شروع آرایه باشد). (مهندسی فناوری اطلاعات (IT) ـ آزاد ۸۵)

۱) $Loc(A[i,j]) = \alpha + [(i-\ell_1)(u_1-\ell_1+1)+j-\ell_2]*e$

۲) $Loc(A[i,j]) = \alpha + [(j-\ell_2)(u_1-\ell_1+1)+i-\ell_1]*e$

۳) $Loc(A[i,j]) = \alpha + [(i-\ell_1)(u_2-\ell_2+1)+j-\ell_2]*e$

۴) $Loc(A[i,j]) = \alpha + [(j-\ell_2)(u_2-\ell_2+1)+i-\ell_1]*e$

۲۴ـ فرض کنید $x=(x_1,x_2,...,x_n)$ و $y=(y_1,y_2,...,y_m)$ دو لیست پیوندی خطی ساده باشند. مرتبه زمانی الگوریتم اتصال دو لیست در لیست $z$ چیست؟ $z=(x_1,x_2,...,x_n,y_1,y_2,...,y_m)$ (مهندسی فناوری اطلاعات (IT) ـ آزاد ۸۵)

۱) $O(m)$ ۲) $O(n+m)$ ۳) $O(nm)$ ۴) $O(n)$

۲۵ـ عبارت پسوندی (Postfix) معادل عبارت میانوندی $((A-(B+C))*D)\$(E+F)$ کدام است؟ (مهندسی فناوری اطلاعات (IT) ـ آزاد ۸۵)

۱) $ABC+D-*EF+\$$ ۲) $ABC--D*EF+\$$ ۳) $ABC+-D*EF+\$$ ۴) $ABC-D+*EF+\$$

۲۶ـ دو پشته را در یک آرایه $V[1...m]$ ارائه کرده‌ایم که در خلاف جهت یکدیگر رشد می‌کنند. مرتبه زمان روال‌های حذف و اضافه و شرط پر بودن هر دو پشته چیست؟ (مهندسی فناوری اطلاعات (IT) ـ آزاد ۸۵)

۱) $top+1=top_2, O(1)$ ۲) $top_1=top_2, O(1)$ ۳) $top_1+1=top_2, O(m)$ ۴) $top_1=top_2, O(m)$

۲۷ـ مقدار ارزیابی عبارت پسوندی (Postfix) زیر چیست؟ ($ عملگر توان است) (مهندسی کامپیوتر ـ آزاد ۸۵)

**6 2 3 + - 3 8 2 / + * 2 $ 3 +**

۱) 50 ۲) 48 ۳) 60 ۴) 52

۲۸ـ اگر یک ماتریس خلوت (Sparse Matrix) را به صورت آرایه یک بعدی ذخیره نماییم که هر عنصر آن سه تایی مرتب **(i , j , value)** می‌باشد و **i** شماره سطر و **j** شماره ستون و **value** مقدار عنصر غیر صفر می‌باشد. مرتبه زمانی به دست آوردن ترانهاده ماتریس به ترتیب ستونی کدام است؟ (مهندسی کامپیوتر ـ آزاد ۸۵)

(ماتریس خلوت به صورت **A[0 .. t , 1 .. 3]** و به ترتیب سطری در حافظه تعریف شده و ماتریس اصلی **B[1 .. m , 1 .. n]** می‌باشد.)

۱) $O(t)$ ۲) $O(nt)$ ۳) $O(mn)$ ۴) $O(n+t)$

۲۹ـ در اضافه کردن یک گره به لیست پیوندی دو طرفه، چند اشاره‌گر باید تغییر یابد، و تعداد حالات مجاز برای تغییر آن‌ها چیست؟ (مهندسی کامپیوتر ـ آزاد ۸۵)

۱) 4 اشاره‌گر و 8 حالت مجاز ۲) 2 اشاره‌گر و 6 حالت مجاز ۳) 4 اشاره‌گر و 6 حالت مجاز ۴) 2 اشاره‌گر و 8 حالت مجاز

۳۰ـ دو صف را در یک آرایه یک بعدی **M[1..n]** ذخیره می‌کنیم به گونه‌ای که ابتدا و انتهای آرایه جلوی هر صف باشد. کدام شرط پر بودن آرایه نیست؟ (با فرض این که با هر حذفی از صف، صف مربوطه به سمت جلو شیفت داده می‌شود). (مهندسی کامپیوتر ـ آزاد ۸۶)

۱) $rear_1 + 1 = rear_2$ ۲) $rear_1 = n$ ۳) $rear_2 = 1$ ۴) $rear_1 = rear_2$

۳۱ـ ماتریس پایین مثلثی ماتریس مربعی است $n \times n$ که عناصر بالای قطر اصلی آن برابر صفر می‌باشند. اگر عناصر غیرصفر را به صورت قطری با شروع از قطر اصلی در یک آرایه یک بعدی ذخیره کنیم آدرس ذخیره‌سازی **loc(A[i , j])** چیست؟ (مهندسی کامپیوتر ـ آزاد۸۶)

($\alpha$ آدرس شروع آرایه در حافظه است و **e** اندازه هر عنصر آرایه می‌باشد.)

۱) $\alpha + [\frac{i(i-1)}{2} + j - 1] * e$ ۲) $\alpha + [[n+(n-1)+\cdots+(n-(i-j))] - n + (i-1)] * e$

۳) $\alpha + [\frac{j(j-1)}{2} + i - 1] * e$ ۴) $\alpha + [\frac{1}{2} + j - 1] * e$

۳۲ـ برای معکوس کردن یک لیست پیوندی ساده (زنجیر) حداقل به چند اشاره‌گر نیاز داریم؟ (مهندسی کامپیوتر ـ آزاد ۸۶)

۱) 2 ۲) 4 ۳) 3 ۴) 5

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۳۳ـ** اگر رشته اعداد **2,7,9,5,3,4** را به ترتیب از سمت چپ وارد یک **STACK** نماییم. کدام یک از خروجی‌های زیر از این **STACK** امکان‌پذیر خواهد بود؟ (خروجی‌ها را از سمت چپ به راست) (مهندسی فناوری اطلاعات (IT) ـ آزاد ۸۶)

۱) 9,2,7,5,3,4 ۲) 5,7,9,4,3,2 ۳) 7,5,9,2,4,3 ۴) 2,5,7,9,3,4

**۳۴ـ** آرایه زیر را در نظر بگیرید:

**A : array[1..10, 1..20, 1..7] of real;** با فرض این که هر عنصر **real** نیاز به **4** بایت حافظه دارد و آرایه **A** به صورت سطری در حافظه با آدرس شروع **1024** ذخیره شود. مشخص کنید آدرس شروع عنصر **A[8,15,5]** کدام است؟ (مهندسی فناوری اطلاعات (IT) ـ آزاد ۸۶)

۱) 1978 ۲) 4840 ۳) 1658 ۴) 3816

**۳۵ـ** آرایه دو بعدی **A[1..n , 1..m]** به صورت سطری در حافظه با آدرس شروع **100** ذخیره شده است، در این صورت آدرس عنصر **A[i,j]** آرایه در حافظه کدام است؟ (فرض کنید اندازه هر عنصر آرایه یک بایت است) (مهندسی فناوری اطلاعات (IT) ـ آزاد ۸۶)

۱) $i+j*m-(m-99)$ ۲) $n*i-n+j+99$ ۳) $i*m+j*n+100$ ۴) $j+i*m-m+101$

**۳۶ـ** الگوریتم زیر کدام عملیات را انجام می‌دهد؟ (مهندسی کامپیوتر ـ آزاد ۸۷)

```
void fun (Dlink * P , Dlink *t)
{
    P ⟶ llink =t;
    P⟶ rlink = t ⟶ rlink
    t⟶ rlink ⟶ llink = P;
    t⟶ rlink = P;
}
```

۱) گره t را به سمت راست گره P اضافه می‌کند.
۲) گره t را از لیست حذف نموده و P را جایگزین آن می‌نماید.
۳) گره P را به سمت راست گره t اضافه می‌کند.
۴) گره P را از لیست حذف نموده و t را جایگزین آن می‌نماید.

**۳۷ـ** تعداد عناصر غیر صفر یک آرایه اسپارس سه قطری **n×n** چقدر می‌باشد؟ (مهندسی فناوری اطلاعات (IT) ـ آزاد ۸۸)

۱) $3n-3$ ۲) $3n+2$ ۳) $3n-2$ ۴) $3n+3$

**۳۸ـ** خروجی حاصل از اجرای **proce()** به ترتیب از راست به چپ چیست؟ (مهندسی کامپیوتر ـ آزاد ۸۸)

```
void proce()
{ char *s[3]={"red","green","blue"};
  cout << s[1];
  cout << "\n" << (s[1]+1);
}
```

۱) blue , green
۲) green , red
۳) green و خروجی دوم یک آدرس را نمایش می‌دهد.
۴) reen , green

**۳۹ـ** تابع زیر داده **ITEM** را به صف اضافه می‌کند. چنانچه صف را توسط یک آرایه چرخشی پیاده‌سازی نماییم در جای خالی چه دستوری باید نوشته شود؟ (مهندسی کامپیوتر ـ آزاد ۸۹)

```
QINSERT(QUEUE,N,FRONT,REAR,ITEM)
If FRONT = 1 and REAR = N, or if FRONT = REAR + 1, then :
   write : OVERFLOW, and Return.
If FRONT = NULL, then :
    Set FRONT := 1 and REAR := 1
ELSE if REAR = N then :
    Set [ ? ]
Else :
     Set REAR := REAR + 1
Set     QUEUE[REAR] := ITEM.
Return.
```

۱) REAR := N − 1 ۲) REAR := 1 ۳) REAR := N + 1 ۴) REAR = FRONT = 2

**۴۰ـ** عبارت محاسباتی **P** با نماد پسوندی نوشته شده است: **P : 12,7,3,−,/,2,1,5,+,*,+** پس از تبدیل **P** به عبارت میانوندی آن را ارزیابی می‌نماییم. نتیجه برابر با کدام یک از گزینه‌های زیر می‌باشد؟ (مهندسی فناوری اطلاعات (IT) ـ آزاد ۸۹)

۱) 17 ۲) 13 ۳) 15 ۴) 20

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۴۱ـ تعداد دنباله‌های مجاز خروجی با k ورودی از یک پشته (stack) برابر با کدام گزینه است؟** (مهندسی کامپیوتر ـ آزاد ۹۰)

۱) $\frac{1}{k+1}\binom{2k}{k}$ ۲) $\frac{1}{k}\binom{2k}{k}$ ۳) $2k+1$ ۴) $\frac{1}{k-1}\binom{2k}{k}$

**۴۲ـ آرایه int S[2][3]={2,4,6,8,10,12} را در نظر بگیرید، با فرض آن که زبان مربوطه آرایه‌ها را به صورت سطری ذخیره نموده و برای هر متغیر صحیح دو بایت اختصاص می‌دهد، اگر آدرس شروع آرایه 0X 0012ff80 باشد آدرس عضو S [1] [1] کدام گزینه است؟**

(مهندسی کامپیوتر ـ آزاد ۹۰)

۱) 0X 0012ff88 ۲) 0X 0012ff86 ۳) 0X 0012ff0c ۴) 0X 0012ff82

## پاسخنامه تست‌های طبقه‌بندی شده کنکوری فصل چهارم

**۱ـ گزینه «۱»** آخرین گره در یک لیست پیوندی یک طرفهٔ ساده، گره‌ای است که فیلد Link آن برابر nil باشد، پس باید لیست پیوندی از ابتدا تا گره‌ای که فیلد Link آن برابر با nil است پیمایش شود، البته به شرطی که $L \neq nil$ باشد.

**۲ـ گزینه «۲»** در تابع what که به صورت بازگشتی تعریف شده، اگر لیست ورودی تهی باشد صفر، و در غیر این صورت اگر فقط دارای یک گره باشد مقدار آن گره و چنانچه بیش از یک گره داشته باشد، مقدار اولین عنصر به اضافهٔ مقدار برگشتی از خود فراخوانی تابع، با لیست ورودی که گرهٔ اوّل و دوّم از آن حذف شده است را برمی‌گرداند. در نتیجه مجموع مقادیر موجود در مکان‌های فرد لیست اولیه، توسط تابع what محاسبه می‌شود.

**۳ـ گزینه «۴»** با توجه به اینکه هر عنصری که از پشته pop می‌شود در صف قرار می‌گیرد و هر عنصری که از صف حذف می‌گردد در پشته push می‌شود، پس هیچ عنصری در این عملیات حذف نشده و یک ترتیب خطی از دستورات را داریم، بنابراین برای هر ترتیب نهایی فقط یک ترتیب اولیه وجود دارد.

**۴ـ گزینه «۱»** برای آنکه مشخص شود که رشتهٔ $A[1..n]$ یک palindrome است باید عنصر اول و آخر یعنی $A[end], A[start]\ (start=1, end=n)$ با هم مقایسه شوند. اگر مساوی بود $A[end-1], A[start+1]$ مقایسه شود و همین روند ادامه یابد. اگر n فرد باشد تا زمانی که start=end شود و اگر n زوج بود تا جایی که $start > end$ باشد ادامه می‌دهیم.

**۵ـ گزینه «۱»** ابتدا کل عبارت را با استفاده از روش پرانتزگذاری، پرانتزگذاری کرده و سپس عملگر را به بعد از عملوندهایش منتقل می‌کنیم، داریم:

$$(+(+a(/b(-cd)))(/(-ab)(-(+c(*d5))(/a(-bc)))))$$
$$\Rightarrow (+(+a(/b(cd-)))(/(ab-)(-(+c(d5*))(/a(bc-)))))$$
$$\Rightarrow (+(+a(bcd-/))(/(ab-)(-(cd5*+)(abc-/))))$$
$$\Rightarrow (+(abcd-/+)(/(ab-)(cd5*+abc-/-)))$$
$$\Rightarrow (+(abcd-/+)(ab-cd5*+abc-/-/)) \Rightarrow abcd-/+ab-cd5*+abc-/-/+$$

**۶ـ گزینه «۲»** با استفاده از فرمول‌های ذکر شده برای آرایهٔ سه بعدی در این فصل داریم:

آدرس به روش ستونی : $20/000+[(9-1)\times 20\times 30+(11-1)\times 30+(21-1)]\times 4 = 20/000+[4800+300+20]\times 4$

$$= 20/000+5120\times 4 = 20/000+20480 = 40480$$

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۷ـ گزینه «۳»** در این قطعه برنامه ابتدا یک صف خالی به نام $Q_3$ ایجاد می‌شود. سپس در حلقهٔ while تا زمانی که صف $Q_1$ و صف $Q_2$ هیچ‌کدام خالی نباشد یکی به شمارندهٔ i اضافه شده و دو عنصر یکی از صف $Q_1$ و دیگری از صف $Q_2$ حذف می‌شود و به ترتیب در x و y قرار می‌گیرد. اگر مقدار y با شمارندهٔ i برابر باشد، مقدار x در صف $Q_3$ قرار خواهد گرفت.

در این صورت خواهیم داشت:

| i | x | y | شرط حلقه | شرط $y=i$ | $Q_3$ |
|---|---|---|---|---|---|
| 0 | – | – | ✓ | | – |
| 1 | 10 | 1 | ✓ | ✓ | 10 |
| 2 | 25 | 5 | ✓ | × | 10 |
| 3 | 17 | 7 | ✓ | × | 10 |
| 4 | 41 | 4 | ✓ | ✓ | 10,41 |
| 5 | 19 | 9 | ✓ | × | 10,41 |
| 6 | 26 | 6 | × | ✓ | 10,41,26 |

**۸ـ گزینه «۱»** در این الگوریتم، تا زمانی که کاراکتر ورودی وجود داشته باشد، اگر کاراکتر ورودی '(' باشد آن را در پشته push کرده و چنانچه به ')' برسد به شرطی که پشته خالی نباشد یک '(' را از پشته pop می‌کند و اگر پشته تهی باشد پیغام unbalanced صادر می‌شود. اشکال این الگوریتم در این جا است که اگر کاراکتر ورودی وجود نداشته باشد، یعنی حلقهٔ while خاتمه یافته باشد باید بررسی کند که آیا پشته پُر یا خالی است. اگر پشته خالی باشد، رشتهٔ پرانتزهای ورودی متوازن است و در غیر این صورت نامتوازن می‌باشد.

**۹ـ گزینه «۲»** حذف اولین عنصر، به راحتی انجام می‌شود و در هر حالتی فقط باید یک اشاره‌گر جا به جا شود. برای درج یک عنصر در ابتدا نیز در هر حالتی باید دو اشاره‌گر را تغییر دهیم. درج یک عنصر در انتهای لیست نیز به دلیل آنکه اشاره‌گر R به آخرین عنصر اشاره دارد آسان است و نیاز به جابه‌جایی دو اشاره‌گر دارد. امّا حذف آخرین عنصر نیاز به آدرس عنصر ماقبل آخر دارد و چون این آدرس در دسترس نیست پس باید لیست را از ابتدا پیمایش کنیم تا به آدرس ماقبل آخر برسیم. هزینهٔ این عمل وابسته به تعداد عناصر لیست است.

**۱۰ـ گزینه «۳»** تعداد دفعات اجرای تابع First برابر است با:

(خط اول برنامه) 1+ تعداد دفعات اجرای حلقهٔ while دوم

بنابراین، در دور اول حلقه while اول، تابع First، n-1 بار فراخوانی می‌شود. در دور دوم n-2 بار و به همین ترتیب در هر دور تعداد فراخوانی‌ها یکی کاهش می‌یابد. لذا، تعداد کل دفعات اجرای تابع First به صورت زیر به دست می‌آید:

$$1+(n-1)+(n-2)+\cdots+1=1+\sum_{i=1}^{n-1} i=1+\frac{n(n-1)}{2}$$

**۱۱ـ گزینه «۱»** برای درج یک عنصر جدید بعد از گرهٔ x در یک لیست پیوندی دو طرفه، ابتدا اشاره‌گر next و prev گره جدید مقداردهی می‌شود و سپس اشاره‌گر prev از گرهٔ بعد از گرهٔ x، مقداردهی می‌شود و در آخر اشاره‌گر next گرهٔ x را مقداردهی می‌کنیم.

**۱۲ـ گزینه «۲»** در آرایهٔ سه بعدی $A[1...n_1,1..n_2,1...n_3]$ عنصر $A[i,j,k]$ به روش سطری برابر: $(i-1)n_2n_3+(j-1)n_3+k$

بوده و با استفاده از فرمول $L\times$ (چندمین -1) + H داریم: $(L=1),(\alpha=H)$

$$\alpha+(i-1)n_2n_3+(j-1)n_3+(k-1)$$

**۱۳ـ گزینه «۳»** با استفاده از روش پرانتزگذاری داریم:

$$(((a/((b-c)+d))*(e-a))*c)\Rightarrow(((a/(bc-)+d))*(ea-))*c)$$
$$\Rightarrow(((a/(bc-d+))*(ea-))*c)\Rightarrow(((abc-d+/)*(ea-))*c)$$
$$\Rightarrow((abc-d+/ea-*)*c)\Rightarrow abc-d+/ea-*c*$$

**۱۴ـ گزینه «۳»** به طور کلی در پشته اعمال push و pop با هزینهٔ $O(1)$ انجام می‌شود و چون در B[i]، اندیس کوچک‌ترین عنصر بین A[1] تا A[i] قرار دارد، Findmin هم با هزینهٔ $O(1)$ انجام می‌شود ولی Findmax نیاز به پیمایش کل پشته دارد که در بدترین حالت هزینهٔ آن $O(n)$ است.

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۱۵ـ گزینه «۲»** در روش سطری از بعد اوّل شروع می‌کنیم و چون در زبان C اندیس آرایه از صفر شروع می‌شود، داریم:

$$10000+[5\times20\times30+10\times30+20]\times4=10000+(3000+300+20)\times4=10000+3320\times4=23280$$

**۱۶ـ گزینه «۲»** برنامه زیر بهترین روش معکوس کردن یک لیست پیوندی است که در آن از دو اشاره‌گر به جز نام لیست استفاده شده است.

```
m=Null;
while(A){
t=m; m=A;
A=A→Link;
m→Link=t;}
A=m;
```

**۱۷ـ گزینه «۲»** با توجه به متن سؤال، پیاده‌سازی پشته با استفاده از آرایه است. اگر خروجی تابع top () در پشته را اندیس خانه آخرین عضو پشته در نظر بگیریم. همچنین اندیس شروع آرایه P و اندیس شروع در پشته D را یک فرض کنیم؛ مشاهده می‌شود که D.top () همواره برابر K است. در نتیجه شرط $P(K)>P(D.top(1))$ همیشه غلط است. در نتیجه مقدار متغیر done برابر true می‌شود و حلقه while در هر دور حداکثر یک بار اجرا خواهد شد. بنابراین، تمام دستورات حلقه for دارای مرتبه پیچیدگی $O(1)$ است و کل زمان اجرای الگوریتم، پیچیدگی زمانی $\theta(n)$ خواهد داشت.

**تذکر:** مرتبه پیچیدگی زمان الگوریتم مورد نظر سؤال برابر $\theta(n)$ است، لذا گزینه‌های ۱، ۲ و ۴ نیز می‌توانند صحیح باشند. و علت آن در نظر نگرفتن تفاوت بین نمادهای $\theta$ و $O$ در سؤالات کنکور است.

**۱۸ـ گزینه «۱»** با استفاده از روش پرانتزگذاری، جواب حاصل می‌شود.

$$A+(((B*C)-((D/(E\,\hat{}\,F))*G))*H)$$

$$\Rightarrow ABC*DEF\,\hat{}\,/G*-H*+$$

**۱۹ـ گزینه «۳»** برای به دست آوردن گزینه ۳ ابتدا باید عدد 1 و بعد 2 را در پشته قرار داده و سپس 2 و بعد 1 را خارج کنیم و بعد عدد3 ,4 ,5 را وارد شیشه کرده و سپس 5 را خارج کنیم. حال مشخص می‌شود که بعد از 5 نمی‌توان عدد 3 را در خروجی قرار داد و این گزینه با هیچ ترتیبی نمی‌تواند از اعمال موردنظر به دست آید.

**۲۰ـ گزینه «۱»** با استفاده از روش پرانتزگذاری داریم:

$$(A+(((B*C)-((D/(E\,\hat{}\,F))*G))*H))$$

$$\Rightarrow(A+(((BC*)-((D/(EF\hat{}))*G))*H))\Rightarrow(A+(((BC*)-((DEF\,\hat{}\,/)*G))*H))$$

$$\Rightarrow(A+(((BC*)-(DEF\,\hat{}\,/G*))*H))\Rightarrow(A+((BC*DEF\,\hat{}\,/G*-)*H))$$

$$\Rightarrow(A+(BC*DEF\,\hat{}\,/G*-H*))\Rightarrow ABC*DEF\,\hat{}\,/G*-H*+$$

**۲۱ـ گزینه «۳»** در ماتریس بالامثلثی تمامی عناصر پایین قطر اصلی صفر می‌باشد، پس می‌توان عناصر غیر صفر را سطر به سطر در یک آرایهٔ یک بعدی ذخیره کرد و با استفاده از فرمول زیر مشخص کرد که $A[i,j]\neq0$ با کدام $B[k]$ برابر است: (n تعداد سطرها و ستون‌های ماتریس بالا مثلثی است.)

$$K=n(i-1)-\frac{i(i-1)}{2}+j$$

**۲۲ـ گزینه «۲»** برای محاسبه آدرس A[3][4][2] به صورت زیر عمل می‌کنیم:

$$0+[(2-0)\times10\times20+(4-0)\times20+(3-0)]\times4=(400+80+3)\times4=483\times4=1932$$

**۲۳ـ گزینه «۲»** ابتدا مشخص می‌کنیم $A[i,j]$ چندمین عنصر به روش ستونی در آرایهٔ دو بعدی A می‌باشد، داریم:

$$(j-\ell_2)(u_1-\ell_1+1)+(i-\ell_1+1)$$

سپس با استفاده از فرمول، آدرس را مشخص می‌کنیم:

$$Loc(A[i,j])=\alpha+[((j-\ell_2)(u_1-\ell_1+1)+i-\ell_1+1)-1]\times e=\alpha+[(j-\ell_2)(u_1-\ell_1+1)+i-\ell_1]\times e$$

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۲۴ـ گزینه «۴»** با توجه به لیست پیوندی Z که ابتدا لیست x و بعد لیست y آمده است، کافی است فقط لیست x پیمایش شده و لیست y به انتهای آن اضافه شود، پس فقط n گره را باید پیمایش کنیم، بنابراین مرتبه زمانی آن $O(n)$ خواهد بود.

**۲۵ـ گزینه «۳»** با استفاده از روش پرانتزگذاری داریم:

$$((A-(B+C))*D)\$(E+F) \Rightarrow ((A-(BC+))*D)\$(EF+)$$
$$\Rightarrow ((ABC+-)*D)\$(EF+) \Rightarrow (ABC+-D*)\$(EF+) \Rightarrow ABC+-D*EF+\$$$

**۲۶ـ گزینه «۱»** در هر دو پشته مرتبهٔ زمانی حذف و اضافه $O(1)$ می‌باشد و هر گاه $Top_2$ یکی جلوتر از $Top_1$ باشد، هر دو پشته پر است $(Top_1+1=Top_2)$.

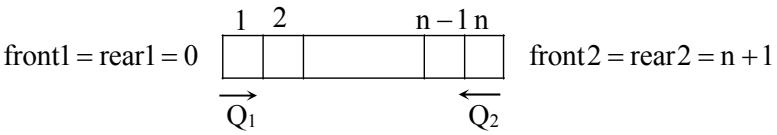
**۲۷ـ گزینه «۴»** با استفاده از روش پرانتزگذاری داریم: $(\$=\uparrow)$

$$6,2,3,+,-,3,8,2,/,+,*,2,\uparrow,3,+$$
$$((((6,(2,3,+),-)(3,(8,2,/),+),*),2,\uparrow),3,+)$$
$$\Rightarrow ((((6,5,-),(3,(8,2,/),+),*),2,\uparrow),3,+)$$
$$\Rightarrow (((1,(3,(8,2,/),+),*),2,\uparrow),3,+) \Rightarrow (((1,(3,4,+),*),2,\uparrow),3,+)$$
$$\Rightarrow (((1,7,*),2,\uparrow),3,+) \Rightarrow ((7,2,\uparrow),3,+) \Rightarrow (49,3,+) \Rightarrow 52$$

**۲۸ـ گزینه «۴»** با توجه به روش ذکر شده در قسمت ترانهاده ماتریس خلوت، مرتبه زمانی این عمل $O(cols + items)$ یعنی تعداد ستون‌ها + تعداد عناصر غیرصفر است.

**۲۹ـ گزینه «۱»** برای اضافه کردن یک گره به لیست پیوندی دوگانه، باید 4 اشاره‌گر تغییر یابد و تعداد حالات مجاز 8 حالت است؛ زیرا اگر x را گره‌ای در نظر بگیریم که می‌خواهیم گره جدید p را به بعد از آن اضافه کنیم باید فیلد $x \rightarrow Rlink$ حتماً بعد از فیلد $x \rightarrow Rlink \rightarrow Llink$ مقداردهی شود.

**۳۰ـ گزینه «۴»**




آرایه فوق را در نظر می‌گیریم، با این فرض که هیچ‌گاه $front2 = n+1, front1 = 0$ تغییر نمی‌کند و در هر حذفی از صف‌ها، صف مربوطه شیفت داده می‌شود، بنابراین اگر $rear2 = 1$ یا $rear1 = n$ یا $rear1 + 1 = rear2$ هر دو صف پر بوده و در واقع در آرایه فضای خالی وجود ندارد.

**۳۱ـ گزینه «۲»** ماتریس پایین مثلثی زیر را در نظر بگیرید:

اگر عناصر غیرصفر این ماتریس را به ترتیب قطری با شروع از قطر اصلی در یک آرایه یک بعدی ذخیره کنیم، به عنوان مثال، آدرس $A[3,2]$ برابر با $\alpha+(6-1)e$، یعنی $\alpha+5e$ خواهد شد. با بررسی گزینه‌ها داریم:

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 5 & 2 & 0 & 0 \\ 8 & 6 & 3 & 0 \\ 10 & 6 & 7 & 4 \end{bmatrix}$$

۱) $\alpha+[\frac{3\times 2}{2}+2-1]*e=\alpha+4e$

۲) $\alpha+[[4+(4-1)]-4+(3-1)]*e=\alpha+(6-1)*e=\alpha+5e$

۳) $\alpha+[\frac{2\times 1}{2}+3-1]*e=\alpha+3e$

۴) $\alpha+[\frac{3}{2}+2-1]*e=\alpha+2.5e$

**۳۲ـ گزینه «۳»** با توجه به الگوریتم معکوس کردن یک لیست پیوندی ساده برای این کار به سه اشاره‌گر نیاز می‌باشد.

**۳۳ـ گزینه «۳»** می‌توانیم ابتدا 2، بعد 7 را در پشته وارد کنیم و سپس 7 را در خروجی قرار دهیم و 9 و بعد 5 را در پشته وارد کنیم و 5 و بعد 9 و بعد 2را در خروجی بنویسیم و در آخر 3 و بعد 4 را در پشته قرار داده و در خروجی 4 بعد 3 را بنویسیم.

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۳۴ـ هیچ‌کدام از گزینه‌ها صحیح نیست.** ابتدا بر اساس روش سطری، چندمین بودن $A[8,15,5]$ را در آرایهٔ سه بعدی $A$ مشخص می‌کنیم، داریم:

$$(8-1)\times 20\times 7+(15-1)\times 7+5=7\times 140+14\times 7+5=1083$$

سپس با استفاده از فرمول، آدرس شروع را محاسبه می‌کنیم:

$$1024+(1083-1)\times 4=1024+1082\times 4=5352$$

---

**۳۵ـ گزینه «۲»** ابتدا بر اساس روش سطری، چندمین بودن $A[i,j]$ را در آرایهٔ دو بعدی $A$ مشخص می‌کنیم، داریم: $(i-1)\times n+j=n*i-n+j$

سپس با استفاده از فرمول، آدرس شروع را محاسبه می‌کنیم: $100+(n*i-n+j-1)\times 1=n*i-n+j+99$

---

**۳۶ـ گزینه «۳»** در تابع fun ابتدا t در فیلد لینک سمت چپ p قرار می‌گیرد و بعد فیلدلینک سمت راست p برابر با فیلدلینک سمت راست t می‌شود و این عملیات نشان می‌دهد که گره p به سمت راست گره t اضافه شده است.

---

**۳۷ـ گزینه «۳»** در ماتریس سه قطری $n\times n$، در قطر اصلی $n$ عنصر غیر صفر وجود دارد و در قطرهای بالا و پایین قطر اصلی هر کدام $n-1$ عنصر غیرصفر وجود دارد که در مجموع $n+(n-1)+(n-1)=3n-2$ عنصر غیرصفر وجود دارد.

---

**۳۸ـ گزینه «۴»** در این تابع یک آرایه‌ای از اشاره‌گرهای کاراکتری تعریف شده و در زمان تعریف با مقادیر سه رشته $blue, green, red$ مقداردهی شده است، پس $S[1]$ آدرس شروع رشته green است که با چاپ آن رشته green در خروجی چاپ می‌شود و $S[1]+1$ برابر $\&S[1][1]$ است که آدرس حرف دوم رشته green می‌باشد و چاپ آن باعث می‌شود رشته reen در خروجی چاپ شود.

---

**۳۹ـ گزینه «۲»** در این برنامه اگر $rear=n$ و $front=1$ باشد یا $front=rear+1$ پیغام overflow صادر می‌شود و اگر $front=Null$، یعنی صف خالی باشد $front=rear=1$ می‌شود و اگر $front\neq Null$ و $Rear=N$ باشد، چون شرط اول نیز برقرار نبوده که به این شرط رسیده، پس $front\neq 1$ است و می‌توانیم از عنصر اول برای درج استفاده کنیم پس $Rear=1$ صحیح است.

---

**۴۰ـ گزینه «۳»** با استفاده از روش پرانتزگذاری داریم: $((12,(7,3,-),/)\ ,\ (2,(1,5,+),*),+)=((12,4,/)\ ,\ (2,6,*),+)=(3,12,+)=15$

---

**۴۱ـ گزینه «۱»** تعداد دنباله‌های مجاز با $k$ ورودی در پشته برابر با عدد $k$ام از دنباله اعداد کاتالان یعنی $C_k=\frac{1}{k+1}\binom{2k}{k}$ است.

---

**۴۲ـ گزینه «۱»** می‌دانیم در زبان C اندیس از صفر شروع می‌شود، بنابراین آرایهٔ s به صورت $S[0..1][0..2]$ بوده و در این آرایه آدرس $S[1][1]$ مورد سؤال می‌باشد. با استفاده از روش انتقال اندیس‌ها آدرس را به دست می‌آوریم، داریم:

$$S\overset{+1}{[0..1]}\overset{+1+1}{[0..2]}\approx S[1..2][1..3]$$

$$S\overset{+1}{[1]}\overset{+1}{[1]}\approx S[2][2]$$

چندمین به روش سطری $=(2-1)\times 3+2=5$

آدرس $=0X0012ff80+(5-1)\times 2=0X0012ff80+8=0X0012ff88$

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



# فصل پنجم

## «داده‌ساختارهای مبتنی بر گراف»

### تست‌های طبقه‌بندی شده کنکوری فصل پنجم

#### «گراف و درخت»

۱ـ تابع زیر را برای درخت دودویی T در نظر بگیرید: (مهندسی کامپیوتر ـ سراسری۷۵)

```
function Number (T : Tree): Integer;
    begin
        Number : = 0 ;
        if T ≠ nil then
            if  Rchild (T) = nil and Lchild (T) = nil then
                Number := 1
            else
                Number := Number (Rchild (T)) + Number (Lchild(T)) + 1;
    end;
```

این تابع چه عملی را انجام می‌دهد؟

۱) تعداد برگ‌های T را محاسبه می‌کند.
۲) تعداد گره‌های T را محاسبه می‌کند.
۳) تعداد گره‌های دو فرزندی T را محاسبه می‌کند.
۴) تعداد گره‌های غیر برگ را محاسبه می‌کند.

۲ـ پیش ترتیب (Preorder) یک درخت دودویی کامل با برچسب‌های متفاوت داده شده است. همچنین برچسب‌های برگ‌های آن درخت مشخص شده‌اند. می‌خواهیم درخت را بسازیم (درخت دودویی کامل یک درخت دودویی است که در آن گره‌ها یا بدون فرزند باشند یا دارای دو فرزند باشند). (مهندسی کامپیوتر ـ سراسری۷۵)

۱) تنها در صورتی که درخت متوازن (Balanced) هم باشد آن را می‌توان ساخت.
۲) درخت را می‌توان ساخت ولی حاصل واحد نیست.
۳) اطلاعات فوق کافی نیست لذا درخت را نمی‌توان ایجاد کرد.
۴) درخت را می‌توان ساخت و حاصل درختی واحد است.

۳ـ اگر T یک درخت دودویی باشد روال زیر آن را به کدام روش پیمایش می‌کند؟ (مهندسی کامپیوتر ـ سراسری۷۶)

```
Procedure Traversal (T);
begin
  push (S, T);
  while (Stack S is not empty) do
   begin
     T := pop (S);
     while (T ≠ nil) do
       begin
          push (S , rightchild(T));
           visit(T);
           T := leftchild(T).
       end;
    end;
end;
```

۱) Preorder
۲) Inorder
۳) Postorder
۴) Levelorder

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۴ـ دو پیمایش Preorder و Postorder از یک درخت دودویی با n گره در دسترس است. کدام گزینه صحیح است؟** (مهندسی کامپیوتر ـ سراسری ۷۶)

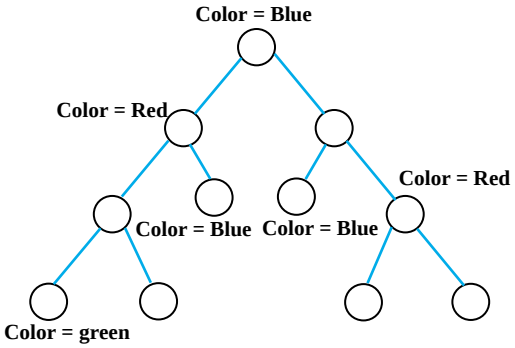
۱) برگ‌ها و گره‌های تک فرزندی و ریشه را می‌توان تعیین کرد و تعداد درخت‌های دودویی به تعداد کل گره‌های پیمایش شده بستگی دارد.

۲) نمی‌توان درختی از روی این دو پیمایش ساخت و فقط ریشه را می‌توان تعیین کرد.

۳) برگ‌ها و گره‌های تک فرزندی و ریشه را می‌توان تعیین کرد و تعداد درخت‌های دودویی که می‌توان ساخت به تعداد گره‌های تک فرزندی بستگی دارد.

۴) فقط ریشه درخت را می‌توان تعیین کرد و نمی‌توان برگ‌ها و گره‌های تک فرزندی را تعیین کرد.

**۵ـ در درخت زیر، برای تعدادی از گره‌ها مشخصهٔ Color تعریف شده است که رنگ آن گره‌ها را نشان می‌دهد. برای مشخص‌کردن رنگ یک گرهٔ n، برنامه زیر را اجرا می‌کنیم:** (مهندسی کامپیوتر ـ سراسری ۷۷)




```
Function  Find Color (n) : Color;
begin
    if  n  has  Color  attribute  then
            Find Color: = Color (n);
     else
            Find Color: = Find Color (Parent (n))
end;
```

**اگر Find Color (n) برای هر یک از 11 گره‌های درخت فوق اجرا شود، در مجموع چه تعداد رنگ قرمز (Red) باز می‌گردد؟**

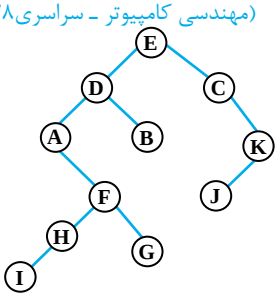
۱) ۲ ۲) ۶ ۳) ۴ ۴) ۵

**۶ـ روال زیر کدام ویژگی از درخت دودویی T را محاسبه می‌کند؟** (مهندسی کامپیوتر ـ سراسری ۷۸)

```
Function  test (T)
    If  T=Null  Then  return  0
    return 1+max (test (T . leftchild), test (T.rightchild))
```

۱) تعداد عناصر ۲) تعداد زیر درخت‌ها ۳) تعداد برگ‌ها ۴) ارتفاع

**۷ـ کدام گزینه پیمایش زیر به روش RLN (زیر درخت راست، زیر درخت چپ و سپس ریشه) را نشان می‌دهد؟** (مهندسی کامپیوتر ـ سراسری ۷۸)




۱) K J C B G I H F A D E

۲) K J C B D G F H I A E

۳) J K C B G I H F A D E

۴) J K C G I H F B A D E

**۸ـ n عنصر با کلیدهای مختلف را می‌توان با استفاده از یک درخت دودویی جستجوی T که در ابتدا تهی است به صورت زیر مرتب کرد:**

**۱ـ عناصر را به ترتیب در T درج کن.**

**۲ـ T را به روش «بین ترتیب» (inorder) پیمایش کن و عناصر را به همین ترتیب در خروجی بنویس. مرتبه زمان اجرای این الگوریتم به ترتیب در بهترین حالت، بدترین حالت و حالت متوسط (از راست به چپ) کدام است؟** (مهندسی کامپیوتر ـ سراسری ۷۸)

۱) $O(n^2)$ و $O(n^2)$، $O(n\log n)$

۲) $O(n\log n)$ و $O(n\log n^2)$، $O(n)$

۳) $O(n\log n)$ و $O(n^2)$، $O(n\log n)$

۴) $O(n\log n)$ و $O(n^2)$، $O(n)$

**۹ـ تعداد گره‌های یک درخت دودویی که یک عبارت ریاضی را نمایش می‌دهد 14 می‌باشد. عملگرهای این عبارت، دودویی یا یکتایی (unary) می‌باشند. کدام یک از گزینه‌های زیر صحیح است؟** (مهندسی کامپیوتر ـ سراسری ۷۹)

۱) این عبارت حتماً تعداد فردی عملگر دودویی دارد.

۲) این عبارت حتماً تعداد زوجی عملگر یکتایی دارد.

۳) این عبارت نمی‌تواند عملگر دودویی داشته باشد.

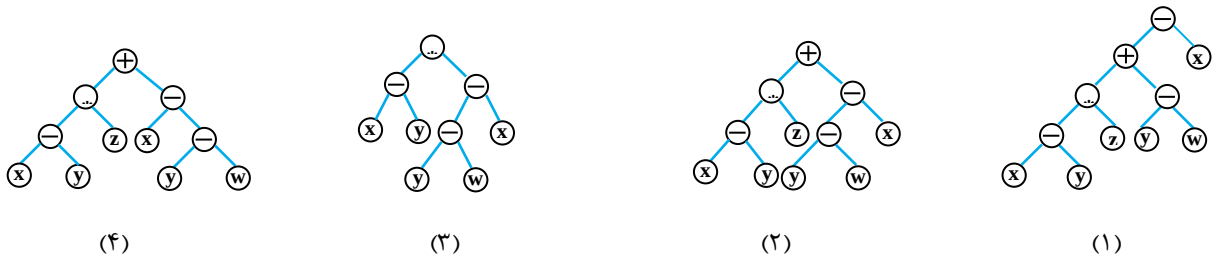
۴) حداقل یک عملگر یکتایی در این عبارت وجود دارد.

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



۱۰ـ درخت دودویی معادل عبارت $(x-y)*z+(y-w)-x$ برابر است با: (علوم کامپیوتر ـ سراسری ۸۱)



(۱) (۲) (۳) (۴)

۱۱ـ اگر T یک درخت دودویی غیرتهی با n گره باشد و $n_0$ تعداد گره‌های نهایی (برگ)، $n_1$ تعداد گره‌هایی با درجه 1 و $n_2$ تعداد گره‌های با درجه 2 و k عمق درخت باشد، کدام یک از روابط ذیل همیشه صحیح است؟ (مهندسی کامپیوتر ـ آزاد ۸۱)

الف) $k=\lfloor \log_2^n \rfloor +1$ ب) $n_0=n_2+1$ ج) $n=2^k-1$

۱) الف و ج ۲) ب ۳) الف و ب ۴) الف، ب و ج

۱۲ـ در یک درخت T با n عنصر، همه‌ی عناصر غیربرگ دارای دقیقاً ۲ فرزند هستند. E(T) و I(T) را به ترتیب مجموع عمق برگ‌ها و مجموع عمق عناصر غیربرگ تعریف می‌کنیم. اگر $T(n)=E(T)-I(T)$ باشد، داریم: (مهندسی کامپیوتر ـ سراسری ۸۲)

۱) $T(n)=T(n-2)+2$ ۲) $T(n)=T(n-1)+1$ ۳) $T(n)=T(n-1)+n-1$ ۴) $T(n)=T(n-2)+n-2$

۱۳ـ در یک درخت باینری دلخواه پیچیدگی زمان سه پیمایش postorder ، preorder و Inorder به ترتیب برابر است با: (علوم کامپیوتر ـ سراسری ۸۲)

۱) $O(n),O(\log n),O(n)$ ۲) $O(n^2),O(n),O(n^2)$ ۳) $O(n),O(n),O(n)$ ۴) $O(n),O(\log n),O(n^2)$

۱۴ـ روش‌های پیمایش پیش ترتیب، میان ترتیب و پس ترتیب را در درخت‌های دودویی در نظر بگیرید. کدام گزاره صحیح است؟ (علوم کامپیوتر ـ سراسری ۸۲)

۱) ترتیب ملاقات برگ‌ها در روش‌های پیش ترتیب و میان ترتیب یکسان است ولی در پیش ترتیب و پس ترتیب متفاوت می‌باشند.
۲) تنها در حالت‌های خاصی ممکن است برگ‌های درخت در هر سه روش به یک ترتیب ملاقات شوند.
۳) برگ‌های درخت با ترتیب‌های متفاوتی ملاقات می‌شوند.
۴) برگ‌های درخت در هر سه روش همیشه به یک ترتیب ملاقات می‌شوند.

۱۵ـ اگر دنباله‌های Inorder و Preorder از یک درخت دودویی موجود باشد آنگاه: (مهندسی کامپیوتر ـ آزاد ۸۲)

۱) می‌توان درخت دودویی ساخت اما یکتا نخواهد بود.
۲) امکان ساخت درخت دودویی وجود ندارد.
۳) می‌توان درخت دودویی ساخت و یکتا خواهد بود.
۴) می‌توان درخت دودویی ساخت اما دو درخت متفاوت تولید خواهد شد.

۱۶ـ سطح ـ ترتیب (level-order) یک درخت، عناصر درخت را به ترتیب سطح (عمق) آن عناصر و در هر سطح از چپ به راست بررسی می‌کند. فرض کنید که دنباله‌های سطح ـ ترتیب و میان ترتیب (inorder) یک درخت دودویی جستجو داده شده‌اند. کدام یک از گزینه‌های زیر درست است؟ (مهندسی کامپیوتر ـ سراسری ۸۳)

۱) تنها با دنباله سطح ـ ترتیب می‌توان درخت را ساخت، ولی درخت یکتا نیست.
۲) تنها با دنباله سطح ـ ترتیب می‌توان درخت را به صورت یکتا ساخت.
۳) با سطح ـ ترتیب و میان ترتیب می‌توان درخت را ساخت ولی درخت یکتا نیست.
۴) با سطح ـ ترتیب و میان ترتیب می‌توان درخت را ساخت و درخت یکتا است.

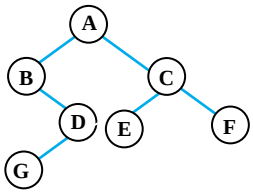
www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۱۷ـ الگوریتم زیر را در نظر بگیرید:** (علوم کامپیوتر ـ سراسری ۸۳)

```
void traverse (ptr p){
    stack s; ptr q;
    push (p,s);
    while (!empty (s)){
        q =pop (s);
        if (q! = NULL){push (q → left,s);
                       print (q → label);
                       push (q → right,s);
                       }
                   }
               }
```




**اگر این الگوریتم برای پیمایش درخت بالا استفاده شود، کدام یک از ترتیب‌های زیر به دست می‌آید؟**

۱) A C F E B D G ۲) A B D G C E F ۳) G D B E F C A ۴) A B C D E F G

**۱۸ـ رویه زیر را در نظر بگیرید:** (مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۳)

```
Procedure Test(T)
begin
   While (T≠0)do
    begin
       Test(Left child (T));
       Visit   T;
       T:=Rightchild (T);
       end
end
```

۱) یک درخت دودویی را به صورت postorder پیمایش می‌کند.

۲) یک درخت دودویی را به صورت preorder پیمایش می‌کند.

۳) یک درخت دودویی را به صورت inorder پیمایش می‌کند.

۴) هیچ‌کدام

**۱۹ـ یک درخت دودویی کامل با ارتفاع h چند گره دارد؟** (مهندسی کامپیوتر ـ سراسری۸۴)

۱) $2^h$ گره ۲) $2^{h+1}$ گره ۳) بین $2^h$ و $2^{h+1}$ گره ۴) بین $2^{h-1}$ و $2^h$ گره

**۲۰ـ الگوریتم پیمایش درخت در زمان O(d) اجرا می‌شود. d چه می‌باشد؟** (علوم کامپیوتر ـ سراسری ۸۴)

۱) تعداد برگ‌های درخت ۲) عمق درخت ۳) درجه درخت ۴) تعداد گره‌های درخت

**۲۱ـ شرط اصلی برای اینکه بتوان از روی دو پیمایش داده شده یک درخت باینری، درخت اصلی را ساخت چیست؟** (علوم کامپیوتر ـ سراسری ۸۴)

۱) تمام داده‌ها در درخت متمایز باشند.

۲) فقط ریشه از بقیه متمایز باشد.

۳) داده‌های زیر درخت چپ ریشه از داده‌های زیر درخت راست متمایز باشند ولی در خود زیر درخت‌های چپ و راست می‌توانند داده تکراری وجود داشته باشد.

۴) داده‌های موجود در برگ‌های درخت متمایز باشند.

**۲۲ـ پس ترتیب (Postorder) یک درخت دودویی کامل با برچسب‌های متفاوت داده شده است. همچنین برچسب‌های برگ‌های آن درخت مشخص شده‌اند. می‌خواهیم درخت را بسازیم:**

**(درخت دودویی کامل، درختی دودویی است که در آن گره‌ها یا بدون فرزند باشند و یا دارای دو فرزند باشند)** (مهندسی کامپیوتر ـ آزاد ۸۴)

۱) تنها در صورتی که درخت متوازن (Balanced) هم باشد آن را می‌توان ساخت.

۲) درخت را می‌توان ساخت ولی حاصل واحد نیست.

۳) اطلاعات فوق کافی نیست لذا درخت را نمی‌توان ایجاد کرد.

۴) درخت را می‌توان ساخت و حاصل درختی واحد است.

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۲۳ـ کدام گزینه درخت دوتایی (Binary tree) را به درستی تعریف می‌نماید؟** (مهندسی کامپیوتر و مهندسی فناوری اطلاعات (IT) ـ آزاد ۸۴)

۱) از تعداد محدودی گره تشکیل شده، گره مشخصی به نام ریشه دارد و سایر گره‌ها در مجموعه‌های جدا از هم قرار دارند.

۲) مجموعه‌ای است شامل تعداد محدودی گره که یا تهی است و یا گره مشخصی به نام ریشه دارد و سایر گره‌ها در مجموعه‌ای جدا به نام زیر درخت راست و زیر درخت چپ قرار دارند.

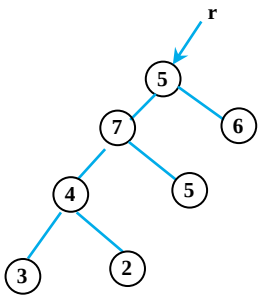
۳) از تعداد محدودی گره تشکیل شده، گره مشخصی به نام ریشه دارد و سایر گره‌ها در مجموعه‌های جدا به نام زیر درخت راست و زیر درخت چپ قرار دارند.

۴) مجموعه‌ای است شامل تعداد محدودی گره، گره مشخصی به نام ریشه دارد و سایر گره‌ها زیر درخت‌های آن را تشکیل می‌دهند.

**۲۴ـ فرض کنید در یک درخت، درجه تمام عناصر داخلی (Internal) K باشد، فرض کنید n ، تعداد عناصر خارجی (External) باشد، در این صورت، کدام‌یک از گزاره‌های زیر ، برقرار است؟** (مهندسی کامپیوتر ـ سراسری ۸۵)

۱) $n \bmod k = 1$　　۲) $(n+1) \bmod k = 1$　　۳) $n \bmod (k-1) = 1$　　۴) $(n-1) \bmod (k-1) = 1$

**۲۵ـ الگوریتم زیر بر روی یک گره r از درخت دودویی اجرا می‌شود.** (مهندسی کامپیوتر ـ سراسری ۸۵)

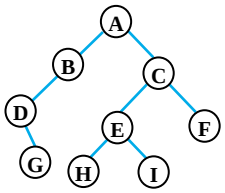



```
Traverse (r, inc)
    if  r is not a leaf
        Then {Traverse (left(r), inc + 1)
              Traverse(right(r), inc + 1)
              If key(left(r)) > key(right(r))
                Then key(r) ← key(r) + inc
              }
```

**اگر Traverse (r,0) بر روی درخت با ریشه r و کلیدهای شکل فوق اجرا شود بیشترین مقدار key عناصر درخت در انتها چه خواهد بود؟**

۱) 10　　۲) 9　　۳) 8　　۴) 7

**۲۶ـ پیمایش پسوندی (postorder) درخت دودویی زیر چیست؟ (از چپ به راست)** (مهندسی کامپیوتر ـ آزاد ۸۵)




۱) GDBACEHIF

۲) GDBHIEFCA

۳) GDBCEHIFA

۴) DGBAHEICF

**۲۷ـ اگر عبارت SBDHXEJKTFG نتیجه پیمایش پیشوندی یک درخت دودویی کامل باشد کدام یک از گزینه‌های زیر صحیح است ؟**

(مهندسی کامپیوتر ـ سراسری۸۶)

۱) HXDBJKEFGTS نتیجه نمایش پسوندی و HDXBEJKSTFG نتیجه نمایش میانوندی همان درخت هستند.

۲) HXDJKEBFGTS نتیجه نمایش پسوندی و HDXBEJKSTFG نتیجه نمایش میانوندی همان درخت هستند.

۳) HXDBJKEFGTS نتیجه نمایش پسوندی و HDXBJEKSFTG نتیجه نمایش میانوندی همان درخت هستند.

۴) HXDJKEBFGTS نتیجه پیمایش پسوندی و HDXBJEKSFTG نتیجه نمایش میانوندی همان درخت هستند.

**۲۸ـ کدام‌یک از گزاره‌های زیر در مورد یک درخت 5تایی کامل با 95 گره صحیح است؟ با فرض این که ریشه گرهٔ اول باشد و در هر عمق گره‌ها به ترتیب از چپ به راست در نظر گرفته شوند.** (مهندسی کامپیوتر ـ سراسری۸۶)

۱) این درخت 76 برگ دارد و گرهٔ پانزدهم آن پدر گرهٔ هفتاد و دوم آن است.

۲) این درخت 76 برگ دارد و گرهٔ چهاردهم آن پدر گرهٔ هفتاد و دوم آن است.

۳) این درخت 77 برگ دارد و گرهٔ چهاردهم آن پدر گرهٔ هفتاد و دوم آن است.

۴) این درخت 77 برگ دارد و گرهٔ پانزدهم آن پدر گرهٔ هفتاد و دوم آن است.

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



۲۹ـ در یک درخت جست و جوی دودویی $T$ با $n$ گره ، فرض کنید هر گره $x$ دارای کلید $key(x)$ و نیز $size(x)$ است که تعداد عناصر زیر درخت به ریشهٔ $x$ (شامل خود $x$) را نشان می‌دهد. فرض کنید $left[x], right[x], p[x]$ به ترتیب فرزند چپ، راست و پدر $x$ باشد . می‌خواهیم با دریافت $x$ مرتبه‌ی $key(x)$ را به دست آوریم. مرتبه یک عدد، تعداد عددهای کوچک‌تر یا مساوی آن عدد است. کدام یک از الگوریتم‌های زیر درست است؟ (مهندسی کامپیوتر ـ سراسری۸۶)

۱)
```
r := ϕ
y := x
while y ≠ root[T] do
        if y = right[p[y]]
        then r := r + size[left[p[y]]] ×1
         y := p[y]
end
return r
```

۲)
```
r := size[left[x]] + 1
y := x
while y ≠ root[T] do
        if y = right[p[y]]
        then r := r + size[left[p[y]]] + 1
         y := p[y]
end
return r
```

۳)
```
r := size[left[x]]
y := x
while y ≠ root[T] do
        if y = right[p[y]]
        then r := r + size[left[y]]
         y := p[y]
end
return r
```

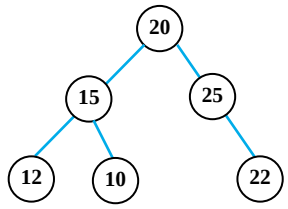
۴)
```
r := ϕ
y := x
while y ≠ root[T] do
        if y = right[p[y]]
        then r := r + size[left[p[y]]]
         y := p[y]
end
return r
```

۳۰ـ خروجی الگوریتم زیر برای یک درخت باینری معادل کدام یک از پیمایش‌های زیر می‌باشد؟ (علوم کامپیوتر ـ سراسری ۸۶)

```
void RCO(ptr p){
    if (p != Null)
        print(p → Data);
        RCO(p → right);
        RCO(p → left);
}
```

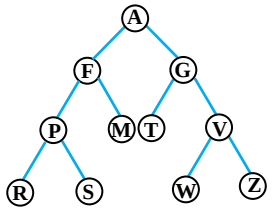
۱) Inorder
۲) reverse preorder
۳) reverse postorder
۴) معادل هیچ پیمایشی نمی‌باشد.

۳۱ـ پیمایش پسوندی درخت زیر چیست؟ (از چپ به راست) (مهندسی کامپیوتر ـ آزاد ۸۶)




۱) 20 15 12 10 25 22
۲) 12 10 15 22 25 20
۳) 12 15 10 20 25 22
۴) 12 10 22 15 25 20

۳۲ـ کدام یک از گزینه‌های زیر پیمایش درخت زیر را به روش LRN (زیر درخت چپ، زیر درخت راست، ریشه) نشان می‌دهد؟ (مهندسی فناوری اطلاعات (IT) ـ آزاد ۸۶)




۱) R S P F M T W Z V G A
۲) T W Z V G R S P M F A
۳) F P R S M G T V W Z A
۴) R S P M F T W Z V G A

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۳۳ـ** یک درخت دودویی کامل دارای ارتفاع 9 می‌باشد. این درخت حداقل دارای چه تعداد گره (NODE) می‌باشد؟

(مهندسی فناوری اطلاعات (IT) ـ آزاد ۸۶)

۱) 17 ۲) 19 ۳) 15 ۴) 21

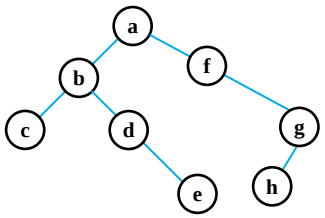
**۳۴ـ** بیشترین تعداد گره‌های (NODE) در یک درخت دودویی با ارتفاع 7 برابر است با:

(مهندسی فناوری اطلاعات (IT) ـ آزاد ۸۶)

۱) 15 ۲) 127 ۳) 255 ۴) 129

**۳۵ـ** نمایش درخت زیر به کدام صورت درست می‌باشد؟

(علوم کامپیوتر ـ سراسری ۸۷)




۱) (a,b(c,d),e,f(g(h)))

۲) (a(b(c,d),e),f(g(h)))

۳) (a(b(c,d(e),f(g(h)))

۴) (a(b(c,d(,e)),f(,g(h,)))

**۳۶ـ** یک درخت دودویی به نام T با n گره داریم. اگر اولین گره در پیمایش Postorder درخت با آخرین گره از پیمایش Preorder درخت یکسان باشد، کدام گزینه زیر را می‌توان نتیجه گرفت؟

(مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۷)

۱) زیر درخت چپ T خالی است. ۲) درخت T حداکثر 3 گره دارد. ۳) زیر درخت راست T خالی است. ۴) ارتفاع درخت T برابر n است.

**۳۷ـ** یک درخت عبارت E فقط با عملگرهای دودویی و گونه کاملاً پرانتزی آن را در نظر بگیرید. اگر درخت فقط یک برگ با برچسب a باشد، گونه کاملاً پرانتزی آن (a) و گونه $(E_1 r E_2)$ است، که در این حالت r عملگر ریشه و $E_1$ و $E_2$ گونه‌های کاملاً پرانتزی زیر درخت‌های ناتهی $E_1$ و $E_2$ است. مثلاً (((a)/((d)/(e)))-((b)*(c))) گونه کاملاً پرانتزی عبارت (a-b*c)/(d/e) است. گونه «میان‌وندی ساده شده»یک عبارت میانوندی کاملاً پرانتزی همان رشته است که از آن برچسب برگ‌ها و نیز همه "(" ها حذف شده باشند. مثلاً ( ( ( / ( ( + ( ) / ( ( ( - ( ) گونه میانوندی ساده شده E است.

فقط با داشتن گونه میانوندی ساده شده یک عبارت ریاضی E با عملگرهای دودویی (خود E را نداریم) کدام‌یک از گزینه‌های زیر صحیح است؟

(بهترین جواب را انتخاب کنید)

(مهندسی کامپیوتر ـ سراسری ۸۷)

۱) درخت عبارت E را می‌توان با $O(n^2)$ و به صورت تک ساخت.

۲) درخت عبارت E را می‌توان با $O(n)$ و به صورت تک ساخت.

۳) درخت عبارت E را می‌توان با $O(n^2)$ ساخت، ولی جواب لزوماً تک نیست.

۴) درخت عبارت E را می‌توان با $O(n)$ ساخت، ولی جواب لزوماً تک نیست.

**۳۸ـ** در یک درخت n-ary هر گره حداکثر n فرزند می‌تواند داشته باشد. در درخت n-ary با k گره و ارتفاع h کدام یک از روابط زیر حد بالایی برای تعداد برگ‌های درخت می‌باشد؟

(مهندسی کامپیوتر ـ سراسری ۸۷)

۱) $h^n$ ۲) $n^h$ ۳) $\log_h^k$ ۴) $\frac{k}{\log_n^k}$

**۳۹ـ** یک درخت دوتایی (binary tree) با پنج سطح دارای چند گره می‌باشد؟

(مهندسی فناوری اطلاعات (IT) ـ آزاد ۸۷)

۱) 15 ۲) 32 ۳) 31 ۴) 16

**۴۰ـ** با ۴ گره چند درخت دودویی متفاوت (از لحاظ فرم درخت و بدون توجه به داده‌های آن) می‌توان ساخت؟

(مهندسی کامپیوتر ـ آزاد ۸۷)

۱) 5 ۲) 42 ۳) 18 ۴) 14

**۴۱ـ** در یک درخت سه‌تایی کامل که مجموعاً 40 رأس می‌باشد، چه تعداد از این رئوس برگ می‌باشند؟

(مهندسی کامپیوتر ـ آزاد ۸۷)

۱) 32 ۲) 27 ۳) 16 ۴) 36

**۴۲ـ** درخت دودویی T در شکل زیر به صورت آرایه نمایش داده شده است. پیمایش Postorder و Inorder آن کدام است؟

(مهندسی کامپیوتر ـ آزاد ۸۷)

| A | | B | | | D | E | | | F | G | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 2 | 3 | | | 6 | 7 | | | 12 | 13 | |

۱) Postorder = FGDEBA
Inorder = AFDGBE

۲) Postorder = FGDEBA
Inorder = ADFGEB

۳) Postorder = FGDBEA
Inorder = AFDGBE

۴) Postorder = FGDBEA
Inorder = AFDBEG

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



۴۳ـ فرض کنید T یک درخت دودوئی کامل با n گره و به ارتفاع $\lg n$ است. می‌خواهیم یک مسیر از یک رأس u به یک رأس دیگر به نام v پیدا کنیم. گره‌های u و v داده شده‌اند و می‌دانیم که هر گره از این درخت به گره‌های فرزند و گره‌ی پدر دسترسی دارد. (مهندسی کامپیوتر ـ سراسری ۸۸)

۱) این کار را می‌توان در $O(\lg^2 n)$ انجام داد.
۲) این کار را نمی‌توان در کم‌تر از $O(n)$ انجام داد.
۳) سریع‌ترین روش استفاده از الگوریتم دایکسترا است.
۴) این کار متناسب با ارتفاع درخت و با $O(\lg n)$ امکان‌پذیر است.

۴۴ـ تعداد درخت‌های دودویی که پیمایش‌های پیش ترتیب (preorder) و پس ترتیب (postorder) آن‌ها برابر با رشته‌های زیر باشد،چقدر است؟ (مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۸)

**Preorder:a b d e f g c h i j k l**
**Postorder: e d g f b i h k l j c a**

۱) 1 درخت ۲) 4 درخت ۳) 8 درخت ۴) بی‌نهایت درخت

۴۵ـ برای هر درخت دودویی غیرتهی، چنانچه $m_0$ تعداد گره‌های پایانی و $m_2$ تعداد گره‌هایی باشد که درجه دو دارند، آنگاه کدام یک از روابط زیر صادق است؟ (مهندسی فناوری اطلاعات (IT) ـ آزاد ۸۸)

۱) $m_0 = m_2 - 1$
۲) $m_0 = m_2 - 2$
۳) میان $m_0$ و $m_2$ رابطه‌ای وجود ندارد.
۴) $m_0 = m_2 + 1$

۴۶ـ کدام‌یک از عبارات زیر نادرست می‌باشد؟ (مهندسی کامپیوتر ـ آزاد ۸۸)

۱) فقط درخت مورب راست است که پیمایش‌های میان ترتیب و پیش ترتیب آن یکسان است.
۲) با دارا بودن پیمایش‌های میان ترتیب و پیش ترتیب یک درخت دودویی می‌توان آن درخت را به صورت یکتا رسم نمود.
۳) با دارا بودن پیمایش‌های پیش‌ترتیب و پس‌ترتیب می‌توان درخت را به صورت یکتا رسم نمود.
۴) پیمایش‌های عمقی چون بازگشتی می‌باشند از پشته استفاده می‌نمایند.

۴۷ـ می‌دانیم که در یک درخت دودویی، سطح (یا عمق) یک گره برابر طول مسیر از آن گره تا ریشه است. ارتفاع درخت هم بزرگ‌ترین سطح گره‌ها در آن درخت است. «پهنای» یک درخت دودویی T را برابر بیش‌ترین تعداد گره‌های هم سطح در T تعریف می‌کنیم. آیا درخت دودویی با n گره و ارتفاع و پهنای زیر وجود دارد؟

I. ارتفاع $\Theta(n)$ و پهنای 1
II. ارتفاع $\Theta(\log n)$ و پهنای $\Theta(n)$
III. ارتفاع $\Theta(n)$ و پهنای $\Theta(n)$
IV. ارتفاع $\Theta(\log n)$ و پهنای $\Theta(\sqrt{n})$

جواب چند تا از موارد فوق درست است؟ (مهندسی کامپیوتر ـ سراسری ۸۹)

۱) 2 ۲) 1 ۳) 3 ۴) 4

۴۸ـ پیمایش preorder درخت دودویی T (از چپ به راست) به صورت acbdhigef می‌باشد. همچنین می‌دانیم رأس‌های b، d، e و f و i در درخت T برگ هستند. پیمایش postorder درخت T کدام است؟ (مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۹)

۱) b c d a i h e g f ۲) b c i h e f g d a ۳) b d c e f g i h a ۴) b d c i e f g h a

۴۹ـ درخت‌های دودویی که Preorder و Postorder آن‌ها در زیر ذکر شده است، چه تعداد است؟ (مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۹)

**Pre : a b d e f g c h i j**
**Post : d g f e b i j h c a**

۱) 8
۲) 4
۳) 1
۴) هیچ درختی را نمی‌توان پیدا کرد.

۵۰ـ تعداد گره‌های غیر برگ یک درخت دودویی پر به عمق d برابر با کدامیک از گزینه‌های زیر است؟ (مهندسی فناوری اطلاعات (IT) ـ آزاد ۸۹)

۱) $2^{d-1}$ ۲) $2^d - 1 - 2^{d-1}$ ۳) $2^d - 1$ ۴) $2^d$

۵۱ـ کدامیک از گزینه‌های زیر نادرست است؟ (مهندسی فناوری اطلاعات (IT) ـ آزاد ۸۹)

۱) در درخت می‌توانیم دارای حلقه باشیم.
۲) یکی از مزایای مهم تبدیل درخت عمومی به دودویی، کاهش اشاره‌گرهای Null می‌باشد.
۳) درخت یک ساختمان داده غیرخطی است.
۴) درخت مرتب درختی است که ترتیب زیر درخت‌ها در آن مهم باشد.

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



۵۲ـ در یک درخت دودویی خاص T گره‌های داخلی هر کدام یک عنصر دارند اما برگ b واقع در عمق d، به جای یک گره، یک آرایه‌ی نامرتب $A_b$ به طول حداکثر $2^d$ است (عمق ریشه صفر فرض می‌شود).
در بین همه‌ی درخت‌های با n عنصر، درختی با بیش‌ترین ارتفاع را در نظر بگیرید. اگر ارتفاع این درخت h باشد، چه رابطه‌ای برقرار است؟ (بهترین گزینه را انتخاب کنید.) (مهندسی کامپیوتر ـ سراسری ۹۰)

۱) $n \le 2^{h+1} - 1$ ۲) $n \le h + 2^h$ ۳) $n \le 2^{h+1}$ ۴) $n \le h + 2^{h+1} - 1$

۵۳ـ با n گره، چند درخت دودویی متمایز (از لحاظ توپولوژی) به ارتفاع n می‌توان ساخت؟ ارتفاع یک درخت دودویی با یک گره را یک در نظر بگیرید. (مهندسی فناوری اطلاعات (IT) ـ سراسری ۹۰)

۱) $2^{n-1}$ ۲) $2^n - 1$ ۳) $2 \times n - 1$ ۴) $2 \times (n-1)$

۵۴ـ نمایش Preorder یک درخت دودویی به صورت (از چپ به راست) XYZABFOD است. این درخت گره تک فرزندی ندارد. همچنین گره‌های DOVBZ مجموعه برگ‌های درخت را تشکیل می‌دهند. نمایش Post order درخت کدام گزینه است؟ (مهندسی فناوری اطلاعات (IT) ـ سراسری ۹۰)

۱) ZBVAYODFX ۲) BVZAYOFDX ۳) ZAYOFVDBX ۴) ZAYVODFBX

۵۵ـ در یک درخت دودویی غیر تهی، تعداد برگ‌های درخت برابر L می‌باشد، تعداد گره‌های با درجهٔ 2 برابر کدام است؟ (علوم کامپیوتر ـ سراسری ۹۰)

۱) $L - 1$ ۲) $L + 1$ ۳) $2^L$ ۴) $\lfloor \log_2(L+1) \rfloor$

**«درخت جست‌وجوی دودویی»**

۵۶ـ می‌خواهیم با واردکردن مقادیر 1 و 2 و 3 به هر ترتیب دلخواه در یک درخت تهی دودویی جستجو (Null Binary Search Tree) یک درخت دودویی جستجو با 3 گره بسازیم. چند درخت دودویی جستجوی متفاوت ممکن است ساخته شود؟ (مهندسی کامپیوتر ـ سراسری ۷۶)

۱) 4 درخت ۲) 3 درخت ۳) 6 درخت ۴) 5 درخت

۵۷ـ یک درخت دودویی جستجوی T با 5 گره با برچسب‌های $a1 < a2 < a3 < a4 < a5$ را در نظر بگیرید. به گره $a_i$ عدد $x_i$ را نسبت می‌دهیم. فرض کنید $x_1 = 2$ و $x_i = 1$ برای $i = 2..5$ می‌خواهیم T را طوری بسازیم که عبارت $C_T = \sum_{i=1}^{5} x_i(depth(a_i)+1)$ مینیمم شود. $depth(a_i)$ عمق گره $a_i$ در T است. کمترین مقدار $C_T$ چند است؟ (مهندسی کامپیوتر ـ سراسری ۷۹)

۱) 13 ۲) 14 ۳) 15 ۴) 16

۵۸ـ فرض کنید اعداد 1 تا 1000 در یک درخت دودویی جستجو ذخیره شده‌اند، و ما می‌خواهیم عدد 363 را پیدا کنیم. کدام یک از ترتیب‌های زیر (از چپ به راست) نمی‌تواند بیانگر ترتیب دسترسی به عناصر درخت در این جستجو باشد؟ (مهندسی کامپیوتر ـ سراسری ۸۰)

۱) 925,202,911,240,912,245,363 ۲) 924,220,911,244,898,258,362,363
۳) 2,252,401,398,330,344,397,363 ۴) 2,399,387,219,266,382,381,278,363

۵۹ـ در یک درخت دودویی جست و جوی T، اگر x برگ و y پدر x باشد، کدام گزینه در مورد مقادیر $a = key[x]$ و $b = key[y]$ درست است؟ (مهندسی کامپیوتر ـ سراسری ۸۰)

۱) b بزرگ‌ترین کلید در T است که کوچک‌تر از a باشد.
۲) a کوچک‌ترین کلید در T است که بزرگ‌تر از b باشد.
۳) b کوچک‌ترین کلید در T است که بزرگ‌تر از a باشد.
۴) هیچ‌کدام از گزینه‌های فوق همواره درست نیست.

۶۰ـ گره‌های 7 و 5 و 2 و 9 و 6 و 4 و 1 و 3 (به ترتیب از چپ به راست) را در یک درخت جستجوی دودویی خالی به نام T درج می‌کنیم. پیمایش پس ترتیب (Postorder) T کدام است؟ (از چپ به راست) (مهندسی کامپیوتر ـ سراسری ۸۱)

۱) 3 و 4 و 6 و 9 و 7 و 5 و 1 و 2 ۲) 9 و 7 و 6 و 5 و 4 و 3 و 1 و 2 ۳) 9 و 7 و 5 و 6 و 4 و 3 و 1 و 2 ۴) 7 و 9 و 5 و 6 و 4 و 3 و 2 و 1

۶۱ـ با n عنصر متفاوت، چند درخت دودویی جست‌وجوی متفاوت به ارتفاع $n-1$ وجود دارد؟ (مهندسی کامپیوتر ـ سراسری ۸۱)

۱) 1 ۲) 2 ۳) $n!$ ۴) $2^{(n-1)}$

۶۲ـ اگر اعداد 3,5,8,2,1,9,12,4,6,7,10 از چپ به راست در یک درخت دودویی جستجوی تهی درج شوند، ارتفاع درخت حاصل چند است؟ (علوم کامپیوتر ـ سراسری ۸۱)

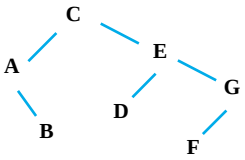
۱) 3 ۲) 4 ۳) 6 ۴) 5

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



۶۳ـ کدام یک از دنباله‌های زیر که هر یک نشان‌دهندهٔ ترتیب درج عناصر از چپ به راست در یک درخت دودویی جست‌وجوی تهی است، درخت زیر را تولید نمی‌کند؟ (مهندسی کامپیوتر ـ سراسری ۸۳)




۱) C A E G D B F
۲) C A B E G D F
۳) C E G A F D B
۴) C E B G A D F

۶۴ـ در یک درخت جستجوی باینری کدام عمل در **O(1)** انجام می‌شود؟ (علوم کامپیوتر ـ سراسری ۸۳)

۱) بررسی تهی بودن درخت ۲) پیدا کردن کمترین مقدار ۳) حذف یک عنصر ۴) درج کردن یک عنصر جدید

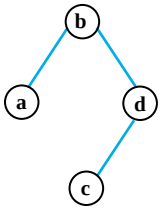
۶۵ـ یک درخت جستجوی باینری با کلمه **SEARCH** (از چپ به راست) بسازید. تعداد متوسط مقایسه برای پیدا کردن یک کاراکتر در این درخت برابر است با: (علوم کامپیوتر ـ سراسری ۸۳)

۱) 2.83 ۲) 2.4 ۳) 2.6 ۴) 6

۶۶ـ یک درخت جستجوی باینری با کلمه **"SEARCH"** (از چپ به راست) بسازید. تعداد متوسط مقایسه برای پیدا کردن یک کاراکتر در این درخت برابر است با: (علوم کامپیوتر ـ سراسری ۸۳)

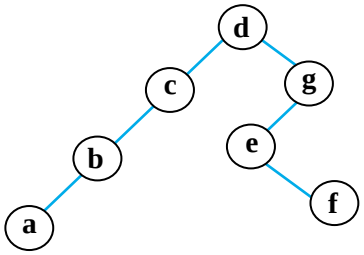
۱) 2.83 ۲) 2.4 ۳) 2.6 ۴) 6

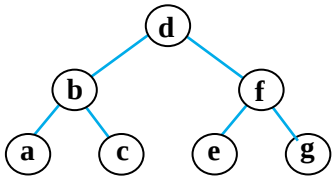
۶۷ـ چند حالت عناصر با کلیدهای $a < b < c < d$ را می‌توان وارد یک درخت دودویی جست و جوی تهی کرد تا درختی به شکل زیر ایجاد شود؟ (مهندسی کامپیوتر ـ سراسری۸۴)




۱) ۴
۲) ۳
۳) ۲
۴) ۱

۶۸ـ با چند عدد دوران می‌توان درخت دودویی جست‌وجوی سمت چپ را به درخت راست تبدیل کرد؟ (مهندسی کامپیوتر ـ سراسری۸۴)




۱) ۲
۲) ۳
۳) ۴
۴) نمی‌شود.

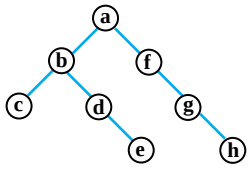
۶۹ـ در یک درخت جستجوی باینری با **n** گره، تعداد مقایسه برای جستجوی یک عنصر حداکثر برابر است با: (علوم کامپیوتر ـ سراسری ۸۴)

۱) $O(\frac{n}{2})$ ۲) $O(n)$ ۳) $O(\sqrt{n})$ ۴) $O(\lg n)$

۷۰ـ برای یک درخت جستجوی باینری که در گره‌های آن تعدادی عدد ذخیره شده است، با داشتن پیمایش **Pre-order** آن کدام جمله صحیح است؟ (علوم کامپیوتر ـ سراسری ۸۴)

۱) فقط با یک پیمایش نمی‌توان درخت را ساخت.
۲) می‌توان درخت را در $O(n^2)$ ساخت.
۳) می‌توان درخت را در $O(n \log n)$ ساخت.
۴) می‌توان درخت را در $O(n)$ ساخت.

۷۱ـ در درخت جستجوی باینری زیر با حذف ریشه کدام یک از عناصر زیر جایگزین می‌شود؟ (علوم کامپیوتر ـ سراسری ۸۴)




۱) f یا b
۲) h یا c
۳) f یا e
۴) e یا h

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



۷۲ـ در یک درخت جستجوی باینری با N گره، تعداد مقایسه برای جستجوی یک عنصر حداکثر برابر است با: (علوم کامپیوتر ـ سراسری ۸۴)

۱) $O(\frac{n}{2})$ ۲) $O(n)$ ۳) $O(\sqrt{n})$ ۴) $O(\lg n)$

۷۳ـ بهترین جواب را علامت بزنید. ساختن یک درخت دودویی جستجو برای n عنصر داده شده حتماً از مرتبه‌ی زیر است؟

(مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۴)

۱) $\Theta(n^2)$ ۲) $\Omega(n^2)$ ۳) $\Theta(n \lg n)$ ۴) $\Omega(n \lg n)$

۷۴ـ کدام یک از گزینه‌های زیر نادرست است؟ (مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۴)

۱) با داشتن پیمایش Inorder یک درخت جستجوی دودویی (BST) همواره می‌توان درخت را منحصر به فرد رسم کرد.

۲) با داشتن پیمایش Preorder یک درخت جستجوی دودویی (BST) همواره می‌توان درخت را منحصر به فرد رسم کرد.

۳) با داشتن پیمایش‌های Inorder و Preorder یک درخت دودویی همواره می‌توان درخت را منحصر به فرد رسم کرد.

۴) با داشتن پیمایش Preoder یک درخت دودویی کامل (Complete) همواره می‌توان درخت را منحصر به فرد رسم کرد.

۷۵ـ تابع زیر چک می‌کند که آیا یک درخت دودویی با عناصر صحیح (int) و متمایز و با ریشه root یک درخت جستجوی دودویی است یا خیر؟

```
Bool IsBST (tree*t)
{
   Return IsBST (t,MININT,MAXINT)
}
Int IsBST (tree*t,int min, int max)
{
 If(t==NULL)return TRUE;
 If(t->data<min||t->data> max)return FALSE;
 Return IsBST(t->Left Child,…A…,…B…)& &IsBST(t->Right Child,…C...,…D…);
}
```

کدام گزینه در جای خالی A, B, C و D باید قرار داده شود؟ (مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۴)

۱) $A = t->data, B = Max, C = min, D = t->date + 1$

۲) $A = max, B = t->data, C = t->data + 1, D = min$

۳) $A = t->data, B = min, C = max, D = t->data + 1$

۴) $A = min, B = t->data, C = t->data + 1, D = max$

۷۶ـ می‌خواهیم با وارد کردن مقادیر 1 و 2 و 3 به هر ترتیب دلخواه در یک درخت تهی دودویی جستجو یک درخت دودویی جستجو با 3 گره بسازیم چند درخت دودویی جستجو می‌توانیم بسازیم؟ (مهندسی کامپیوتر ـ آزاد ۸۴)

۱) ۳ درخت ۲) ۴ درخت ۳) ۵ درخت ۴) ۶ درخت

۷۷ـ اگر عناصر یک درخت جستجوی باینری را به صورت inorder پیمایش کنیم و در داخل یک استک قرار دهیم و سپس عناصر استک را خارج کنیم و یک درخت جستجوی باینری بسازیم، درخت حاصل چه درختی خواهد بود؟ (علوم کامپیوتر ـ سراسری ۸۵)

۱) درخت باینری تغییر نمی‌کند.

۲) درخت باینری مورب به چپ

۳) درخت باینری مورب به راست

۴) زیر درخت‌های چپ و راست درخت باینری تعویض می‌شود.

۷۸ـ با داده‌های زیر یک درخت جستجوی باینری می‌سازیم: (علوم کامپیوتر ـ سراسری ۸۵)

**1300,1400,1025,1100,1000,1050,1200,1500**

اگر عدد 1025 را از درخت حذف کنیم در آن صورت پیمایش preorder درخت پس از حذف برابر است با:

۱) 1300,1050,1000,1100,1400,1200,1500

۲) 1300,1000,1100,1050,1200,1500,1400

۳) 1300,1000,1050,1100,1200,1400,1500

۴) 1300,1050,1000,1100,1200,1400,1500

۷۹ـ یک درخت جستجوی دودویی 150 گره دارد. کدام یک از موارد زیر صحیح است؟ (مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۵)

۱) این درخت حداقل 6 سطح و حداکثر 8 سطح دارد.

۲) این درخت حداقل 6 سطح و حداکثر 7 سطح دارد.

۳) این درخت حداقل 7 سطح و حداکثر 8 سطح دارد.

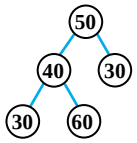
۴) این درخت حداقل 7 سطح و حداکثر 150 سطح دارد.

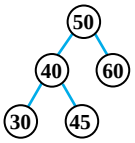
www.modaresanesharif.ac.ir

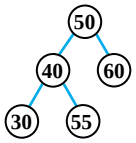
www.modaresanesharif.ac.ir

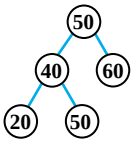


**۸۰ـ کدام یک از درخت‌های زیر یک درخت جستجوی دودویی است؟** (مهندسی فناوری اطلاعات (IT) ـ آزاد ۸۵)

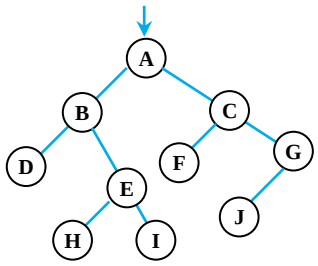
(۱) 


(۲) 


(۳) 


(۴) 


**۸۱ـ اگر T یک درخت جستجوی باینری به صورت زیر باشد که در هر گره آن یک عدد صحیح ذخیره شده است، چهارمین کوچک‌ترین عنصر آن در کدام گره قرار دارد؟** (علوم کامپیوتر ـ سراسری ۸۶)




۱) D
۲) I
۳) H
۴) E

**۸۲ـ کدام گزینه پیمایش Preorder یک درخت جستجوی دودویی BST با پیمایش Postorder به صورت زیر است؟**

$$\text{Postorder}: 5,6,15,10,23,24,22,26,20$$

(مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۶)

۱) $20,23,15,5,6,10,26,24,22$
۲) $6,5,10,15,20,22,24,23,26$
۳) $20,10,6,5,15,26,24,23,22$
۴) $20,10,6,5,15,26,22,24,23$

**۸۳ـ تابع زیر چک می‌کند که آیا یک درخت دودویی با عناصر صحیح (int) و متمایز و با ریشه root یک درخت جستجوی دودویی است یا خیر؟**

```
Bool IsBST(tree*t)
{
   Return IsBST(t,MININT,MAXINT);
}
Int IsBST (tree*t,int min, int max)
{
  If(t= =NULL)return TRUE;
 If(t->data<min || t->data>max)return FALSE;
 Return IsBST(t->Left Child,…A…, …B...)&&IsBST (t->Right Child,…C…,…D…);}
```

**کدام گزینه در جای خالی D,C,B,A باید قرار داده شود؟** (مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۶)

۱) $A=\max\ \ B=t->data\ \ C=t->data+1\ \ D=\min$
۲) $A=\min\ \ B=t->data\ \ C=t->data+1\ \ D=\max$
۳) $A=t->data\ \ B=\max\ \ C=\min\ \ D=t->data+1$
۴) $A=t->data\ \ B=\min\ \ C=\max\ \ D=t->data+1$

**۸۴ـ می‌خواهیم با استفاده از اعداد 14,12,20,15 به هر ترتیب دلخواه در یک درخت تهی دودوئی جست‌وجو، یک درخت دودوئی جست‌وجو با 4 گره بسازیم. در این‌صورت تعداد درخت‌های دودوئی جست‌وجوی ممکن برابر است با:** (مهندسی کامپیوتر ـ آزاد ۸۶)

۱) 16
۲) 12
۳) 14
۴) 8

**۸۵ـ گره‌های 5,9,2,6,7,4,1 را به ترتیب از چپ به راست در یک درخت دودویی جستجوی خالی به نام T درج می‌کنیم. در این صورت پیمایش پس ترتیب (Post Order) درخت T کدام است؟ (داده‌ها را از چپ به راست بخوانید)** (مهندسی فناوری اطلاعات (IT) ـ آزاد ۸۶)

۱) $4,2,1,7,6,5,9$
۲) $5,2,1,4,9,6,7$
۳) $1,4,2,7,6,9,5$
۴) $2,4,7,6,1,9,5$

**۸۶ـ چند درخت جستجوی باینری با عمق 4 می‌توان با این داده‌ها ساخت؟ (ریشه در سطح شماره ۱) $\{1,4,5,10,16\}$** (علوم کامپیوتر ـ سراسری ۸۷)

۱) 22
۲) 25
۳) $5!$
۴) $C_5=\frac{1}{6}\binom{10}{5}$

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۸۷ـ** فرض کنید که پیمایش پسوندی (Postfix) یک درخت دودویی جستجوی T با n عنصر در آرایه‌ای به نام Postfix و به طول n ذخیره شده است، که Postfix[i] کلید i امین گره در این پیمایش باشد. فرض کنید که برای $1 < i < n$ داریم $postfix[i-1] = a$ و $postfix[i] = x$ و $postfix[i+1] = b$ شرط لازم و کافی برای آنکه x برگ درخت T باشد کدام‌یک از موارد زیر است؟ (مهندسی کامپیوتر ـ سراسری ۸۷)

۱) $a < x$ ۲) $x > b$ ۳) $a < b$ ۴) هر سه مورد صحیح است.

**۸۸ـ** تعداد درخت‌های دودویی جست‌وجویی که می‌توان با 36 کلید داده شده مجزا از هم ساخت به طوری که اختلاف عمق برگ‌های آن درخت حداکثر 1 باشد، چند تاست؟ (مهندسی کامپیوتر ـ سراسری ۸۷)

۱) $\binom{16}{5}$ ۲) $\binom{36}{5}$ ۳) $\binom{32}{5}$ ۴) $\binom{36}{32}5!$

**۸۹ـ** چنانچه بخواهیم با استفاده از اعداد 18,12,8,23,8,18 به هر ترتیب دلخواه یک درخت جستجو بسازیم، آنگاه تعداد درختان جستجویی که می‌توانیم ایجاد نماییم برابر است با: (مهندسی فناوری اطلاعات (IT) ـ آزاد ۸۷)

۱) 16 ۲) 10 ۳) 8 ۴) 14

**۹۰ـ** فرض کنید یک درخت تصمیم‌گیری دودویی برای ادغام دو لیست مرتب (Sorted) زیر داریم: List 1:$x_1$ , $x_2$

List2:$y_1$ , $y_2$ , $y_3$ , $y_4$ , $y_5$

در درخت، گره‌های میانی نشان‌دهنده مقایسه و گره‌های برگ نشان‌دهنده ترتیب لیست مرتب نهایی است. حد پایین تعداد گره‌های برگ درخت چند است؟ (مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۸)

۱) 32 ۲) 21 ۳) 15 ۴) 10

**۹۱ـ** تابع زیر بررسی می‌کند آیا یک درخت دودویی با کلیدهای صحیح (int) و متمایز یک درخت جستجوی دودویی است یا خیر؟ محل‌های خالی A تا D کدام گزینه است؟ (مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۸)

```
bool ISBST (tree*t)
{
    return ISBST (t,MININT, MAXINT);
}
Int ISBST (tree*t,int min,int max)
{
  If(t==null)return TRUE
  If(t - >key <min|| t - >key> max) return FALSE
return ISBST (t- >leftchild , A , B)&& ISBST(t - >rightchild, C, D)}
```

۱) $A = min\ ,\ B = t->key + 1\ ,\ C = t->key - 1\ ,\ D = max$

۲) $A = t->key + 1\ ,\ B = min\ ,\ C = max\ ,\ D = t->key - 1$

۳) $A = t->key - 1\ ,\ B = min\ ,\ C = max\ ,\ D = t->key + 1$

۴) $A = min\ ,\ B = t->key - 1\ ,\ C = t->key + 1\ ,\ D = max$

**۹۲ـ** اگر $b_n$ تعداد درخت‌های دودویی باشد که با n گره ساخته می‌شوند کدام یک نادرست است؟ (مهندسی کامپیوتر ـ سراسری ۸۸)

۱) $b_n \in \Omega(2^n)$ ۲) $b_n = \frac{1}{n+1}\binom{2n}{n}$ ۳) $\binom{2}{n}\prod_{i=1}^{n} b_n$ ۴) $b_n = \sum_{k=1}^{n} b_{k-1}b_{n-k}$

**۹۳ـ** هفتمین کلید در پیمایش preorder یک درخت جستجوی دودویی (BST) که پیمایش postorder آن به شکل زیر است: (از چپ به راست)، کدام گزینه است؟

Postorder : 5, 6, 15, 10, 23, 24, 22, 26, 20

(مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۹)

۱) 26 ۲) 24 ۳) 22 ۴) 15

**۹۴ـ** فرض کنید n عدد در بازهٔ [1,log n] داده شده باشند. ارتفاع درخت جستجوی دودویی برای این اعداد چه وضعیتی دارد؟ (فرض کنید از هر عدد حداقل یک نمونه وجود دارد.) (علوم کامپیوتر ـ سراسری ۹۰)

۱) $\log n \le$ ارتفاع $\le n$

۲) $\log^2 n \le$ ارتفاع $\le n$

۳) $\log n \le$ ارتفاع $\le \log^2 n$

۴) $\log(\log n) \le$ ارتفاع $\le \log n$

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



«هرم»

**۹۵ـ** صف اولویت (Priority Oueue) با استفاده از Heap پیاده‌سازی شده است. کدام‌یک از عبارات زیر صحیح است؟ n تعداد اقلام در صف اولویت می‌باشد. (مهندسی کامپیوتر ـ سراسری۷۵)

۱) عملگر «پیداکردن ماکزیمم و حذف آن» از صف اولویت دارای پیچیدگی زمانی $O(1)$ می‌باشد.

۲) عملگر «اضافه‌کردن» یک قلم جدید به صف اولویت دارای پیچیدگی زمانی $O(n\log_2 n)$ می‌باشد.

۳) عملگر «پیداکردن ماکزیمم» دارای پیچیدگی $O(\log_2 n)$ می‌باشد.

۴) عملگر «اضافه‌کردن» یک قلم جدید به صف اولویت دارای پیچیدگی زمانی $O(\log_2 n)$ می‌باشد.

**۹۶ـ** فرض آنکه آرایه A از اندیس 1 تا $n-1$ به صورت heap است. (ریشه heap بزرگ‌ترین عنصر است). الگوریتم زیر برای درج A(n) به این heap پیشنهاد شده است. کدام یک از گزینه‌های زیر را باید بجای . . . قرار دهیم تا الگوریتم همواره درست کار کند؟ (مهندسی کامپیوتر ـ سراسری۷۹)

```
procedure insert(A,n)
    integer i,j,n;
    j ← n;
    i ← ⌊n/2⌋;
    temp ← A(n);
    while ... do
        A(j) ← A(i)
        j ← i;
        i ← ⌊i/2⌋;
    endwhile
    A(j) ← temp;
end
```

۱) $i>1$ and $A(i)>temp$

۲) $i>0$ and $A(i)<temp$

۳) $i>0$ and $A(i)\neq temp$

۴) $i>1$ and $A(i)<A(j)$

**۹۷ـ** آرایه زیر یک heap است. برای درج عدد 95 در آرایه به گونه‌ای که آرایه نهایی نیز وضعیت heap داشته باشد، چند عمل exchange (تعویض دو کمیت) لازم است؟ (مهندسی کامپیوتر ـ سراسری ۸۰)

| |
|---|
| 100 |
| 90 |
| 82 |
| 85 |
| 74 |
| 75 |
| 73 |
| 68 |
| 70 |

۱) دو

۲) چهار

۳) شش

۴) هشت

**۹۸ـ** یک max−heap با n عنصر به صورت آرایه پیاده‌سازی شده است (عنصر ماکزیمم در ریشه است). مناسب‌ترین گزینه برای پیدا کردن عنصر مینیمم در این ساختمان داده کدام است؟ (مهندسی کامپیوتر ـ سراسری ۸۱)

۱) این کار را همواره می‌توان با $O(\log n)$ مقایسه بین عناصر heap انجام داد.

۲) این کار به حداکثر $\frac{n}{2}$ مقایسه بین عناصر heap نیاز دارد.

۳) این کار ممکن است به $n-1$ مقایسه بین عناصر heap نیاز داشته باشد.

۴) تنها در صورتی که heap عناصر تکراری نداشته باشد، می‌توان این کار را با $O(\log n)$ مقایسه بین عناصر heap انجام داد.

**۹۹ـ** یک max-heap با N عنصر متمایز را در نظر بگیرید که با یک آرایه پیاده‌سازی شده است (بزرگ‌ترین عنصر در درایه‌ی اول قرار دارد). چهارمین بزرگ‌ترین عنصر در کدام یک از درایه‌های زیر می‌تواند قرار گیرد؟ (مهندسی کامپیوتر ـ سراسری۸۲)

۱) 2 یا 3 ۲) 8 تا 15 ۳) 4 ، 5 ، 6 یا 7 ۴) همه موارد فوق

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



۱۰۰ـ یک ماکزیمم ـ هیپ حاوی 64 عنصر با کلیدهای متفاوت 1 تا 64 است. بزرگ‌ترین عددی که می‌تواند در برگ واقع در آخرین سطح این هیپ قرار گیرد، کدام یک از اعداد زیر می‌تواند باشد؟ (در ماکزیمم هیپ هر عنصر از فرزندانش کوچک‌تر نیست) (مهندسی کامپیوتر ـ سراسری ۸۳)

۱) 59 ۲) 58 ۳) 57 ۴) 56

۱۰۱ـ اگر آرایه | 14 | 19 | 23 | 11 | 4 | 22 | 15 | 10 | ، نمایش یک درخت باینری باشد، و آن را تبدیل به یک Min – Heap نماییم در آن صورت محتوای آرایه برابر خواهد بود با: (علوم کامپیوتر ـ سراسری ۸۳)

۱) 4,10,15,19,11,23,22,14 ۲) 4,10,14,15,11,19,22,23

۳) 4,10,11,14,15,19,22, 23 ۴) 4,10,19,14,11,23,22,15

۱۰۲ـ فرض کنید که ماکزیمم ـ هیپ حاوی اعداد متمایز 1 تا 1023 است. حداکثر چند تا از اعداد بیش‌تر از 1000 می‌توانند در پایین‌ترین سطح درخت قرار گیرند؟ (علوم کامپیوتر ـ سراسری ۸۳)

۱) 10 ۲) 12 ۳) 13 ۴) 14

۱۰۳ـ فرض کنید که یک ماکزیمم ـ هیپ دودویی حاوی N عدد متمایز است (بزرگ‌ترین عنصر در ریشه و در درایه‌ی اول قرار دارد). چهارمین بزرگ‌ترین عنصر این عناصر در کدام یک از درایه‌ها نمی‌تواند قرار بگیرد؟ (مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۳)

۱) 2 یا 3 ۲) 8 تا 15 ۳) 16 تا 31 ۴) 4، 5، 6 یا 7

۱۰۴ـ ماکزیمم تعداد مقایسه برای min – heap کردن یک max – heap با n گره برابر است با: (علوم کامپیوتر ـ سراسری ۸۴)

۱) $O(n+\log n)$ ۲) $O(\log n)$ ۳) $O(n\log n)$ ۴) $O(2n)$

۱۰۵ـ به یک Min-Heap خالی به ترتیب گره‌هایی با کلیدهای (از راست به چپ) 75,40,55,45,42,50,45,70 اضافه شده است. سپس سه عمل حذف (Delete) بر روی Min-Heap انجام شده است. Min-Heap حاصل مطابق کدام گزینه خواهد بود؟ (مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۴)

۱) [45,50,55,70,75] ۲) [45,50,55,75,70] ۳) [45,55,50,70,75] ۴) [45,55,50,75,70]

۱۰۶ـ آرایه T که در آن تعدادی از خانه‌ها هنوز مقداردهی نشده‌اند را در نظر می‌گیریم. (مهندسی کامپیوتر ـ سراسری ۸۵)

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|
| T | 75 | | 30 | | 17 | 28 | 20 | 7 | | |

با قرار دادن مقادیر کدام یک از موارد زیر این آرایه به یک هیپ تبدیل خواهد شد؟

۱) $T[2]=30\ ;\ T[4]=9\ ;\ T[9]=10\ ;\ T[10]=16\ ;$ ۲) $T[2]=15\ ;\ T[4]=15\ ;\ T[9]=17\ ;\ T[10]=10\ ;$

۳) $T[2]=35\ ;\ T[4]=30\ ;\ T[9]=32\ ;\ T[10]=16\ ;$ ۴) $T[2]=21\ ;T[4]=14\ ;\ T[9]=0\ ;\ T[10]=16;$

۱۰۷ـ اگر در یک min–heap دلخواه جای زیر درخت‌های چپ و راست تعویض شود در آن صورت کدام گزینه صحیح است؟ (علوم کامپیوتر ـ سراسری ۸۵)

۱) همیشه ساختار min–heap از بین می‌رود.

۲) ساختار min–heap تبدیل به max–heap می‌شود.

۳) در برخی از شرایط ساختار min–heap از بین می‌رود.

۴) در هیچ شرایطی ساختار min–heap از بین نمی‌رود.

۱۰۸ـ اگر در یک min-heap دلخواه جای زیردرخت‌های چپ و راست تعویض شود در آن صورت کدام گزینه صحیح است؟ (علوم کامپیوتر ـ سراسری ۸۵)

۱) همیشه ساختار min ۲) ساختار min ۳) heap از بین می‌رود. ۴) heap تبدیل به max

۱۰۹ـ در یک heap با n گره که در یک آرایه نمایش داده شده باشد گره iام در زیردرخت چپ درخت، متناظر با گره jام در زیردرخت راست درخت می‌باشد به طوری که: (علوم کامپیوتر ـ سراسری ۸۵)

۱) $j=i-2^{\lfloor \log_2^i \rfloor+1}$ if $j>n$ $j=\frac{j}{2}$;

۲) $j=i+2^{\lfloor \log_2^i \rfloor-1}$ if $j>n$ $j=\frac{j}{2}$;

۳) $j=i+2^{\lfloor \log_2^i \rfloor-1}$ if $j>n$ $j=\frac{j+1}{2}$;

۴) $j=i+2^{\lfloor \log_2^i \rfloor+1}$ if $j>n$ $j=\frac{j+1}{2}$;

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۱۱۰ـ کدام یک از عبارت‌های زیر همیشه درست است؟** (مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۵)

۱) کوچک‌ترین گره در یک درخت جستجوی دودویی همیشه برگ است.

۲) درج در یک درخت دودویی جستجو می‌تواند در زمان حداکثر $O(1)$ انجام شود.

۳) پیمایش میان ترتیب یک max heap عناصر موجود در آن را به صورت مرتب طی می‌کند.

۴) پیمایش پیش ترتیب یک درخت دودویی جستجو عناصر موجود در آن را به صورت مرتب طی می‌کند.

**۱۱۱ـ کدام یک از گزینه‌های زیر نمی‌تواند یک بازنمایی از یک MaxHeap با آرایه باشد (ترتیب عناصر از چپ به راست است).** (مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۵)

۱) 20 5 10 6 3 4 1

۲) 20 17 17 3 4 2 1

۳) 20 5 10 3 4 2 1

۴) 20 10 18 3 4 2

**۱۱۲ـ در یک آرایه دلخواه با 12 عنصر چند مقایسه حداکثر نیاز می‌باشد تا این آرایه تبدیل به یک min-heap شود؟** (علوم کامپیوتر ـ سراسری ۸۶)

۱) 6 ۲) 10 ۳) 12 ۴) 18

**۱۱۳ـ به یک Min-heap خالی گره‌هایی با کلیدهای (به ترتیب از راست به چپ) 2,50,45,70,75,22,40,55,45 >> اضافه شده است. Min-heap حاصل کدام گزینه است؟** (مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۶)

۱) 2,22,45,40,75,70,45,55,50

۲) 2,45,22,50,75,45,40,70,55

۳) 2,22,40,45,75,70,45,50,55

۴) 2,22,45,40,75,70,45,50,55

**۱۱۴ـ سومین کوچک‌ترین کلید در یک Min Heap با کلیدهای متمایز در درایه‌هایی با چه اندیس‌هایی می‌تواند باشد؟** (مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۶)

۱) 1,2,3 ۲) 2,3,4,5,6,7 ۳) 4,5,6,7 ۴) 1,2,3,4,5,6,7

**۱۱۵ـ نمی‌توان ساختمان داده‌ای برای n عنصر طراحی کرد که .....** (مهندسی کامپیوتر ـ سراسری ۸۷)

۱) ساخت آن $O(n \lg n)$ و حذف بزرگ‌ترین عنصر آن، درج و حذف کلید یک عنصر دلخواه در آن $O(\lg n)$ باشد.

۲) ساخت آن $O(n)$ و حذف بزرگ‌ترین عنصر آن، درج، حذف و افزایش و کاهش کلید یک عنصر دلخواه در آن $O(\lg n)$ باشد.

۳) ساخت آن $O(n)$ و حذف بزرگ‌ترین عنصر آن، درج، حذف و افزایش کلید یک عنصر دلخواه در آن $O(\lg n)$ باشد.

۴) ساخت آن $O(n)$ و حذف بزرگ‌ترین عنصر آن $O(1)$ و درج و حذف یک عنصر دلخواه در آن $O(\lg n)$ باشد.

**۱۱۶ـ آرایه زیر یک heap می‌باشد. برای درج عدد ۸۵ در آرایه به صورتی که آرایه نهایی نیز وضعیت heap را دارا باشد چند عمل exchange (تعویض دو کمیت) لازم می‌باشد؟** (مهندسی فناوری اطلاعات (IT) ـ آزاد ۸۷)

| 90 | 80 | 72 | 75 | 64 | 65 | 63 | 58 | 60 |
|---|---|---|---|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |

۱) چهار ۲) شش ۳) سه ۴) دو

**۱۱۷ـ اگر دنباله‌ی زیر یک max−heap را بخواهد نشان دهد، کدام اعداد باید با چه مقادیر تعویض شوند؟** (علوم کامپیوتر ـ سراسری ۸۸)

**23,17,14,6,3,10,1,5,7,11**

**(a → b به b تغییر نماید یعنی a)**

۱) 14→15 , 7→8 ۲) 1→8 , 10→11 ۳) 3→10 , 6→9 ۴) 3→15 , 6→12

**۱۱۸ـ چند Min-heap متفاوت می‌توان با 7 گره با کلیدهای 1 تا 7 ساخت؟** (مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۸)

۱) 160 ۲) 80 ۳) 40 ۴) 20

**۱۱۹ـ هیپ زیر داده شده است:**

**A[1⋯18] = 20 15 18 7 9 14 16 3 6 8 4 13 10 12 11 1 2 5**

**عمل Change (i,k) کلید A[i] را به k تغییر می‌دهد و با انجام تعداد جابجایی کاری می‌کند که آرایه مجدداً به صورت هیپ درآید. ما این ۲ عمل را به ترتیب انجام می‌دهیم:**

**Change (11,16) Change(2,4)**

**مجموع تعداد جابه‌جایی (swap)ها چند تاست؟** (مهندسی کامپیوتر ـ سراسری ۸۸)

۱) 3 ۲) 4 ۳) 5 ۴) 6

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



۱۲۰ـ کدام یک از اعمال زیر را نمی‌توان در یک **max-heap** با **n** عنصر در مرتبه‌ی **O(lgn)** انجام داد؟ (مهندسی کامپیوتر ـ سراسری ۸۸)

۱) یافتن یک عنصر با کلید مشخص
۲) حذف یک عنصر داده شده
۳) کاهش مقدار کلید یک عنصر داده شده
۴) افزایش مقدار کلید یک عنصر داده شده

۱۲۱ـ یک لیست **12** عنصری حاوی کلیدهای یک تا دوازده به صورت صعودی مرتب است. اگر این لیست به صورت در جا تبدیل به یک **Max Heap** شود، عنصر پنجم لیست کدام گزینه است؟ (مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۹)

۱) 11
۲) 10
۳) 9
۴) 8

۱۲۲ـ فرض کنید که در **Max – heap** استفاده شده در ساختمان داده‌ی **X**، هر عنصر یک زوج مرتب به صورت **< Key, Value >** است که معیار مقایسه‌ی عناصر مقدار **key** آن‌ها می‌باشد. ساختمان داده **X** یک پیاده‌سازی از کدام گزینه است؟ (مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۹)

```
X::A(x){
   Count + + ;
    Max-heap-insert (H,<count,x>);
}
X:: B (x) {
P=Heap-Extract-max(H);
Return-Value (p);
}
```

۱) Stack
۲) Queue
۳) Max – Heap
۴) Min Heap

۱۲۳ـ یک درخت **Max Heap** تحت کدام‌یک از عملیات زیر همچنان **Max Heap** می‌ماند؟ فرض کنید که عمل خواسته شده روی یک یا چند گره دلخواه درخت اعمال شود. (علوم کامپیوتر ـ سراسری ۸۹)

۱) Left Rotate (چرخش به چپ)
۲) Mirror (تعویض بچه‌های چپ و راست گره)
۳) Right Rotate (چرخش به راست)
۴) Exchange (تعویض مقدار موجود در گره با بزرگ‌ترین فرزند)

۱۲۴ـ چند **Min Heap** با هفت عنصر که حاوی کلیدهای متمایز یک تا هفت (با هر ترتیب ورود دلخواه) است می‌توان ساخت؟ (مهندسی فناوری اطلاعات (IT) ـ سراسری ۹۰)

۱) 20
۲) 40
۳) 60
۴) 80

۱۲۵ـ یک **Max Heap** با **n** عنصر را که در آرایه $A[1..n]$ قرار دارد، در نظر بگیرید. مرتبه زمانی الگوریتم حذف عنصر **i**ام $(1 \le i \le n)$ از این **Max Heap** به گونه‌ای که ساختار **Max Heap** را حفظ کند، چیست؟ (مهندسی فناوری اطلاعات (IT) ـ سراسری ۹۰)

۱) ۱
۲) $n$
۳) $\log n$
۴) $n \log n$

۱۲۶ـ اگر **n** داده را همزمان هم در یک هرم مینیمم و هم در یک هرم ماکزیمم قرار دهیم آنگاه حذف کوچک‌ترین عنصر در چه مرتبهٔ زمانی انجام می‌شود؟ (علوم کامپیوتر ـ سراسری ۹۰)

۱) $O(n)$
۲) $O(\log n)$
۳) $O(\log^2 n)$
۴) $O(\frac{n}{\log n})$

۱۲۷ـ اگر **n** داده را همزمان هم در یک هرم مینیمم و هم در یک هرم ماکزیمم قرار دهیم، آنگاه حذف کوچک‌ترین عنصر در چه مرتبه زمانی انجام می‌شود؟ (علوم کامپیوتر ـ سراسری ۹۰)

۱) $O(n)$
۲) $O(\log n)$
۳) $O(\log^2 n)$
۴) $O(\frac{n}{\log n})$

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



## پاسخنامه تست‌های طبقه‌بندی شده کنکوری فصل پنجم

### «گراف و درخت»

**۱ـ گزینه «۲»** گره برگ، گره‌ای است که فرزند چپ و راست آن nil باشد یا به عبارت دیگر فرزند چپ و راست نداشته باشد. با توجه به تابع Number اگر t برگ باشد، مقدار یک، برگشت داده می‌شود و اگر t غیر برگ باشد، تعداد گره‌های زیر درخت چپ به اضافهٔ تعداد گره‌های زیر درخت راست به اضافه یک، برگشت داده می‌شود که این مقدار برابر تعداد کل گره‌های درخت می‌باشد.

**۲ـ گزینه «۴»** با توجه به تعریف درخت دودویی کامل که در سؤال آورده شده است، چون گرهٔ تک فرزندی نداریم و نیز برچسب‌های برگ مشخص شده است، پس می‌توانیم درخت را به صورت منحصر به فرد رسم نماییم.

**۳ـ گزینه «۱»** در روال Traversal، ابتدا آدرس ریشه در پشته، push شده و دستورات حلقهٔ while تا زمانی که پشته تهی نباشد تکرار می‌شود. درون حلقه آدرس ریشه pop شده و در T قرار می‌گیرد و درون حلقهٔ while دوّم تا زمانی که به Nil رسیده باشد آدرس فرزند راست در پشته push و خود گره visit شده و آدرس فرزند چپ در T قرار گرفته و مجدداً این دستورات تکرار می‌شود. با توجه به این عملیات، ابتدا ریشه ملاقات شده، سپس زیر درخت چپ و در آخر زیر درخت راست پیمایش خواهد شد.

**۴ـ گزینه «۳»** با استفاده از پیمایش‌های preorder و postorder می‌توان ریشه و برگ‌ها و گره‌های دو فرزندی و تک فرزندی را مشخص کرد. امّا در مورد گره‌های تک فرزندی نمی‌توان چپ یا راست بودن فرزند را تعیین نمود. بنابراین با داشتن این دو پیمایش، اگر درخت k گرهٔ تک فرزندی داشته باشد، می‌توان $2^k$ درخت متفاوت به دست آورد.

**۵ـ گزینه «۲»** تابع Findcolor یک تابع بازگشتی است که آدرس یک گره را دریافت می‌کند، اگر گره، مشخصه رنگ داشته باشد مقدار آن را برمی‌گرداند، در غیر این صورت به صورت بازگشتی با آدرس پدر گرهٔ دریافتی فراخوانی می‌شود. بنابراین تابع Findcolor اگر گره، مشخصه رنگ داشته باشد مقدار آن و در غیر این صورت مقدار نزدیک‌ترین جد گره که مشخصهٔ رنگ داشته باشد را می‌دهد. در درخت مذکور، دو گره، دارای خصیصهٔ Red بوده و چهار گره دارای نزدیک‌ترین جد با مشخصهٔ Red می‌باشد.

**۶ـ گزینه «۴»** تابع test، آدرس ریشهٔ درخت دودویی را دریافت کرده و این آدرس را با NULL مقایسه می‌کند، اگر آدرس NULL باشد یا به عبارت دیگر درخت تهی باشد، مقدار صفر برگشت داده می‌شود و در غیر این صورت تابع برای زیر درخت چپ و راست مجدداً فراخوانی شده و این عملیات تا جایی انجام می‌شود که به زیر درخت تهی برسد، سپس بیشترین مقدار بین مقادیر برگشتی برای زیر درخت راست و چپ را با یک جمع می‌کند و برمی‌گرداند، این روند نشان می‌دهد که تابع test ارتفاع درخت را محاسبه می‌کند.

**۷ـ گزینه «۳»** در روش پیمایش RLN، ابتدا باید زیر درخت راست، سپس زیر درخت چپ و در آخر ریشه پیمایش شود. البته همین روند باید برای پیمایش زیر درخت‌های راست و چپ نیز به کار رود.

**۸ـ گزینه «۳»** هر گاه یک درخت جستجوی دودویی با n گره، دارای ارتفاع متوازن باشد، عملیات درج با مرتبهٔ اجرایی O(Log n) انجام می‌شود که این حالت، بهترین حالت و حالت متوسط برای درج عنصری در درخت جستجوی دودویی است. در بدترین حالت ممکن است درخت اریب باشد که در این حالت عمق درخت n بوده و عملیات درج با مرتبه اجرایی O(n) انجام می‌شود. بنابراین برای درج n عنصر در یک درخت جستجوی دودویی در بهترین حالت و حالت متوسط مرتبهٔ اجرایی O(nLogn) و در بدترین حالت $O(n^2)$ می‌باشد، پیمایش درخت به روش inorder مرتبهٔ O(n) دارد.

**۹ـ گزینه «۴»** در یک درخت دودویی، سه نوع گره وجود دارد، گره‌های برگ یا گره‌هایی که هیچ فرزندی ندارند، گره‌های تک فرزندی و گره‌های دو فرزندی. اگر تعداد گره‌های برگ را با $n_0$ و تعداد گره‌های تک فرزندی را با $n_1$ و تعداد گره‌های دو فرزندی را با $n_2$ نشان دهیم، تعداد کل گره‌ها (n) از مجموع $n_0+n_1+n_2$ به دست می‌آید، البته با استفاده از اصل موضوعی ۳، داریم: $n_0=n_2+1$. با استفاده از این دو رابطه می‌توانیم نتیجه بگیریم:

$$n=n_0+n_1+n_2=n_2+1+n_1+n_2\Rightarrow n=2n_2+n_1+1$$

با توجه به تعداد گره‌های ذکر شده در سؤال داریم: $14=2n_2+n_1+1\Rightarrow 13=2n_2+n_1$

این رابطه نشان می‌دهد که حتماً تعداد فردی گرهٔ تک فرزندی (یا با توجه به سؤال، عملگر یکتایی) در درخت وجود دارد.

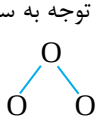
**۱۰ـ گزینه «۱»** برای نمایش یک عبارت ریاضی به صورت یک درخت دودویی، عملگرها در گره‌های پدر و عملوندها در گره‌های برگ قرار می‌گیرند. البته باید بر اساس حق تقدم‌ها از چپ به راست درخت دودویی را به دست آورد.

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۱۱ـ گزینه «۲»** با استفاده از اصل موضوعی ۳، داریم: $n_0 = n_2 + 1$، رابطه (الف) مربوط به ارتفاع یک درخت دودویی کامل بوده و رابطه (ج) مربوط به تعداد گره‌های یک درخت دودویی پر است.

**۱۲ـ گزینه «۱»** با توجه به سؤال، گره‌های درخت T، کلاً یا برگ هستند یا دو فرزندی، بنابراین: $n = n_0 + n_2$ (n تعداد کل گره‌ها). درخت $\underline{3}$ گره‌ای زیر را در نظر بگیرید:  در این درخت یک گرهٔ درجه دو داریم و دو برگ، بنابراین، مجموع عمق برگ‌ها $E(T) = 2 + 2 = 4$ و مجموع عمق عناصر غیر برگ $I(T) = 1$ خواهد بود، در نتیجه $T(3) = 4 - 1 = 3$، با توجه به این مثال می‌توانیم به دست بیاوریم: $T(5) = E(T) - I(T) = 8 - 3 = 5$، و از این دو مقدار نتیجه می‌گیریم: $T(5) = T(3) + 2$. این تساوی نشان می‌دهد که فقط گزینهٔ یک صحیح است.

**۱۳ـ گزینه «۳»** در پیمایش‌های postorder و preorder و Inorder یک درخت دودویی باید تمامی گره‌های درخت یک بار ملاقات شود و اگر درخت دودویی n گره داشته باشد مرتبهٔ زمانی، پیمایش‌های ذکر شده $O(n)$ خواهد بود.

**۱۴ـ گزینه «۴»** در هر سه پیمایش preorder و Inorder و postorder ابتدا زیر درخت چپ و بعد زیر درخت راست ملاقات می‌شود، پس ترتیب ملاقات برگ‌های درخت در هر سه روش یکسان است.

**۱۵ـ گزینه «۳»** با استفاده از دو پیمایش یک درخت دودویی که یکی از این پیمایش‌ها باید inorder باشد می‌توان درخت را به صورت منحصر به فرد ساخت.

**۱۶ـ گزینه «۲»** در درخت جستجوی دودویی تمام مقادیر زیر درخت راست از ریشه بزرگ‌تر و تمام مقادیر زیر درخت چپ از ریشه کوچک‌تر است و همین موارد برای زیر درخت راست و چپ صادق می‌باشد، پس به راحتی می‌توان با استفاده از دنبالهٔ سطح ـ ترتیب درخت جستجوی دودویی، آن را به صورت یکتا ساخت.

**۱۷ـ گزینه «۱»** در این الگوریتم آدرس اولین گره یعنی ریشه در پشته push می‌شود و تا زمانی که پشته خالی نباشد، آدرس ابتدا از پشته pop شده و اگر این آدرس مخالف NULL باشد، ابتدا آدرس زیر درخت چپ در پشته push می‌شود، سپس مقدار کلید گره چاپ شده و دوباره آدرس زیر درخت راست در پشته push می‌شود و با تکرار مجدد حلقه ابتدا این عملیات برای زیر درخت راست و سپس برای زیر درخت چپ انجام می‌شود. بنابراین الگوریتم ذکر شده درخت مورد نظر را به صورت VRL پیمایش می‌کند.

**۱۸ـ گزینه «۳»** در رویه Test ابتدا زیر درخت چپ پیمایش شده، سپس ریشه ویزیت می‌شود و در آخر زیر درخت راست پیمایش خواهد شد.

**۱۹ـ گزینه «۴»** با توجه به اصل موضوعی ۲، بیشترین تعداد گره‌ها در یک درخت دودویی به عمق h برابر $2^h - 1$ است و می‌دانیم درخت دودویی کامل به ارتفاع h درخت دودویی است که تا عمق $h-1$ کاملاً پر و در سطح h ام گره‌ها از چپ به راست چیده شده باشد، بنابراین حداقل گره‌ها در یک درخت دودویی کامل به عمق h برابر $2^{h-1}$، و حداکثر برابر $2^h - 1$ خواهد بود. لازم به ذکر است اگر سطح گره ریشه را صفر در نظر بگیریم، گزینه «۳» صحیح خواهد بود.

**۲۰ـ گزینه «۴»** در پیمایش یک درخت باید تمامی گره‌های درخت یک بار ملاقات شوند.

**۲۱ـ گزینه «۱»** اگر تمامی داده‌های یک درخت، متمایز باشد، از روی دو پیمایش درخت که یکی از پیمایش‌ها باید inorder باشد، می‌توان درخت دودویی را ساخت.

**۲۲ـ گزینه «۴»** چون گره تک فرزندی نداریم و نیز برچسب‌های برگ مشخص شده است، پس می‌توانیم درخت را به صورت منحصر به فرد رسم نماییم.

**۲۳ـ گزینه «۲»** درخت دودویی می‌تواند تهی باشد و اگر تهی نباشد حاوی تعداد محدودی گره است که یک گره خاص به نام ریشه دارد و سایر گره‌ها در دو زیر درخت چپ و راست ریشه قرار دارند. زیر درخت چپ و راست هر کدام درخت دودویی می‌باشند.

**۲۴ـ گزینه «۳»** با توجه به فرض مسئله در درخت، درجهٔ تمامی عناصر غیربرگ k است، بنابراین در عناصر غیر برگ nil وجود ندارد و nk ، nil در برگ‌ها می‌باشد. با در نظر گرفتن اصل موضوعی ۱، اگر تعداد عناصر غیر برگ را h در نظر بگیریم به طور کلی $(n+h)(k-1)+1$، nil خواهیم داشت و داریم:

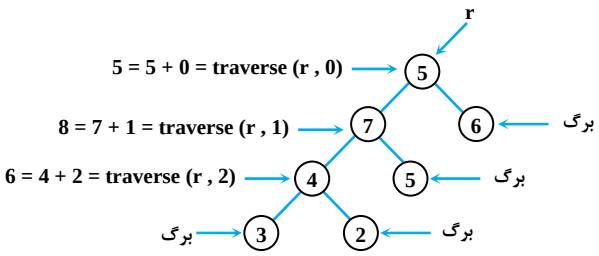
$$(n+h)(k-1)+1 = nk \Rightarrow nk - n(k-1) = h(k-1)+1 \Rightarrow \cancel{nk} - \cancel{nk} + n = h(k-1)+1 \Rightarrow n \bmod (k-1) = 1$$

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



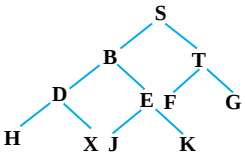
**۲۵ـ گزینه «۳»** در تابع ذکر شده، اگر آدرس دریافتی مربوط به برگ نباشد یک بار تابع با آدرس فرزند چپ و $inc+1$ و یک بار با آدرس فرزند راست و $inc+1$ فراخوانی می‌شود و اگر مقدار $key(left(r))$ بزرگ‌تر از $key(right(r))$ باشد، مقدار $key(r)$ برابر $key(r)+inc$ خواهد شد. یا به عبارت دیگر، اگر مقدار برگشتی key برای زیر درخت چپ گره بیشتر از زیر درخت راست گره باشد، مقدار key گره برابر مقدار قبلی به اضافهٔ inc می‌شود، در شکل مقادیر برگشتی key نشان داده شده است:




همان طور که در شکل نشان داده شده است بیشترین مقدار key عناصر درخت در انتها، 8 است.

**۲۶ـ گزینه «۲»** در پیمایش Postorder ابتدا زیر درخت چپ و بعد زیر درخت راست و در آخر ریشه را پیمایش می‌کنیم.

**۲۷ـ گزینه «۴»** در یک درخت دودویی کامل با n رأس عمق برابر $[\log_2^n]+1$ است، در درخت دودویی کامل ذکر شده در سؤال $n=11$ و عمق $[\log_2^n]+1=4$، خواهد بود و این درخت تا عمق 3 دقیقاً پُر و در سطح چهارم گره‌ها از چپ به راست چیده شده است. با توجه به این موارد شکل درخت عبارت است از:




با استفاده از شکل فوق داریم: $inorder: HDXBJEKSFTG \quad , \quad postorder: HXDJKEBFGTS$

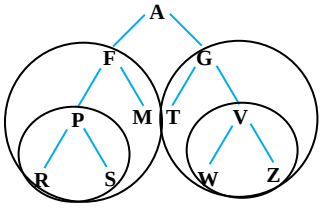
**۲۸ـ گزینه «۱»** در درخت 5 تایی کامل ذکر شده در سؤال، در سطح i ام، $5^{i-1}$ گره وجود دارد و فقط در سطح آخر ممکن است این تعداد گره وجود نداشته باشد، بنابراین داریم: $95=5^0+5^1+5^2+64$ پس در سطح آخر 64 برگ و در سطح سوم 12 برگ داریم که کلاً $64+12=76$ برگ در این درخت 5 تایی وجود دارد و گرهٔ 15 ام پدر گره‌های 72 تا 76 خواهد بود.

**۲۹ـ گزینه «۲»** در یک درخت جستجوی دودویی برای ریشه، تمامی کلیدهای (key) گره‌های زیر درخت چپ از ریشه کوچک‌تر و تمامی کلیدهای (key) گره‌های زیر درخت راست از ریشه بزرگ‌تر است، پس مرتبهٔ $key(x)$ برابر تعداد گره‌های زیر درخت چپ x به اضافهٔ یک است یا به عبارت دیگر مرتبهٔ $key(x)$ برابر $size[left[x]]+1$ خواهد بود.

**۳۰ـ گزینه «۳»** در این برنامهٔ بازگشتی ابتدا مقدار گره چاپ شده، سپس برنامهٔ بازگشتی با زیر درخت راست و در آخر با زیر درخت چپ فراخوانی می‌شود.

**۳۱ـ گزینه «۲»** در پیمایش پسوندی درخت، ابتدا زیر درخت چپ و بعد زیر درخت راست و در آخر ریشه را پیمایش می‌کنیم.

**۳۲ـ گزینه «۴»** در روش LRN، ابتدا زیر درخت چپ و بعد زیر درخت راست و در آخر ریشه را پیمایش می‌کنیم. با تفکیک زیر درخت‌های چپ و راست این کار به راحتی انجام می‌شود، داریم:




**۳۳ـ هیچ‌کدام از گزینه‌ها صحیح نیست.** می‌دانیم در یک درخت دودویی کامل به عمق h باید تا ارتفاع $h-1$ کاملاً پر باشد، بنابراین در درخت مورد سؤال باید تا عمق 8 دقیقاً $2^8-1=255$ گره باشد، پس حداقل تعداد گره‌ها می‌تواند 256 گره باشد.

**۳۴ـ گزینه «۲»** حداکثر گره‌ها در یک درخت دودویی به عمق h برابر است با: $2^h-1$ در سؤال $h=7$ است پس داریم: $2^7-1=128-1=127$

**۳۵ـ گزینه «۴»** در نمایش با پرانتزگذاری، ابتدا مشخصات گرهٔ ریشه می‌آید و درون پرانتز مشخصات فرزندان.

**۳۶ـ گزینه «۴»** اگر در یک درخت دودویی اولین گره در پیمایش postorder درخت با آخرین گره از پیمایش preorder درخت یکسان باشد، این درخت یک درخت مورب است و در درخت مورب با n گره ارتفاع درخت برابر n است.

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۳۷ـ گزینه «۲»** با توجه به مثال ذکر شده در سؤال فقط با داشتن عبارت میانوندی ساده شده می‌توان جواب منحصر به فرد ایجاد کرد.

---

**۳۸ـ گزینه «۲»** اگر هر گره حداکثر بتواند n فرزند داشته باشد، در سطح دوم حداکثر n عنصر و در سطح سوم حداکثر $n^2$ و در سطح hام با فرض آنکه ریشه در سطح $h=0$ باشد، حداکثر $n^h$ عنصر می‌تواند قرار گیرد.

---

**۳۹ـ گزینه «۳»** در یک درخت دودویی به عمق 5، حداکثر $2^5-1=31$ گره می‌تواند باشد.

---

**۴۰ـ گزینه «۴»** بنابر فرمول $\frac{1}{n+1}\binom{2n}{n}$ داریم:

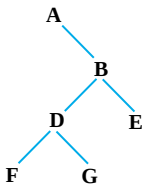
$$\frac{1}{4+1}\binom{2\times 4}{4}=\frac{1}{5}\times\frac{8!}{4!4!}=\frac{\cancel{8}\times 7\times\cancel{6}\times\cancel{5}}{\cancel{5}\times 4\times\cancel{3}\times\cancel{2}\times\cancel{1}}=2\times 7=14$$

---

**۴۱ـ گزینه «۲»** در یک درخت سه‌تایی، در ریشه یک گره، در سطح یک، 3 گره و در سطح دو 9 گره و در سطح سوّم، 27 گره وجود دارد که جمع این گره‌ها، $1+3+9+27=40$ گره می‌باشد و این نتیجه می‌دهد 27 گره در سطح آخر وجود دارد.

---

**۴۲ـ گزینه «۱»** شکل درخت به صورت زیر می‌باشد:




پیمایش Postorder: FGDEBA

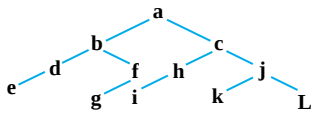
پیمایش Inorder: AFDGBE

---

**۴۳ـ گزینه «۴»** هر دو گره در یک درخت دارای حداقل یک جد مشترک می‌باشند، بنابراین کافیست با حرکت از طریق دو گره u و v به سمت ریشه اولین حد مشترک آن‌ها را یافت که در بدترین حالت دو گره u و v برگ می‌باشند و جد مشترک آن‌ها ریشه می‌باشد که در این حالت طول مسیر $2([\log_2^n]+1)$ می‌باشد و مرتبه یافتن مسیر $O(\log_2^n)$ خواهد بود.

---

**۴۴ـ گزینه «۳»** در پیمایش پیشوندی، ابتدا گره ریشه می‌آید و در پسوندی گره ریشه در آخر پیمایش می‌شود. با توجه به این مطلب می‌توان مکان گره‌های غیربرگ را مشخص کرد، اما در مورد برگ‌ها نمی‌توان مکان را معین کرد که تعداد درخت‌های حاصله برابر $2^{\text{(تعداد گره‌های تک فرزندی)}}$ خواهد بود. به طور مثال یکی از درخت‌ها می‌تواند به شکل مقابل باشد:




---

**۴۵ـ گزینه «۴»** همواره در درخت‌های دودویی غیر تهی تعداد برگ‌ها یکی بیشتر از تعداد گره‌های درجهٔ 2 می‌باشد.

---

**۴۶ـ گزینه «۳»** برای رسم یک درخت به صورت یکتا باید دو تا از پیمایش‌های درخت را داشته باشیم که حتماً باید یکی از پیمایش‌های درخت inorder باشد. در غیر این صورت تعداد درخت‌های به دست آمده برابر $2^{\text{(تعداد گره‌های تک فرزندی)}}$ خواهد بود.

---

**۴۷ـ گزینه «۳»**

I. درست است: درختی که همه گره‌های آن تنها یک فرزند داشته باشد. (درخت اریب)

II. درست است: یک درخت دودویی متوازن و پُر، ارتفاعی برابر با $\theta(\log n)$ دارد و تعداد برگ‌های آن در سطح آخر برابر با $\frac{n}{2}$ است و در نتیجه پهنای آن $\theta(n)$ خواهد بود.

III. درست است: اگر درخت کاملی را با $\frac{n}{2}$ عنصر فرض کنید. تعداد برگ‌های آن و در نتیجه پهنای آن حدوداً برابر $\frac{n}{4}$ است. حال $\frac{n}{2}$ عنصر باقیمانده را به شکل اریب در نظر بگیرید که ارتفاع آن برابر $\frac{n}{2}$ است و درخت کامل را به پایین درخت اریب متصل کنید. بنابراین یک درخت دودویی با ارتفاع $\frac{n}{2}+\log\frac{n}{2}\in\theta(n)$ و پهنای $\frac{n}{4}\in\theta(n)$ به دست خواهد آمد.

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir

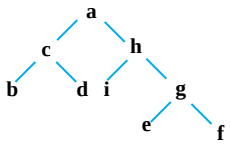


IV. غلط است: اگر پهنای درخت محدود به $\theta(\sqrt{n})$ شود، رشد ارتفاع درخت بسیار بیشتر از $\theta(\log n)$ خواهد شد. حتی اگر در همه سطح‌های درخت $\theta(\sqrt{n})$ عدد گره باشد، تعداد کل گره‌ها $\theta(\sqrt{n})\times\log n = O(n)$ می‌شود که تناقض است.

**تذکر:** چون در صورت سؤال $\theta$ در گزینه‌ها مطرح شده است، لذا پیچیدگی مجانبی و سرعت رشد ویژگی‌های درخت مورد نظر است و باید بررسی شوند. در نتیجه ارائه چند مثال با nهای کوچک دلیل بر صحت گزینه IV نیست.

---

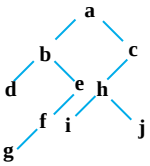
**۴۸ـ گزینه «۴»** در پیمایش Preorder اولین گره‌ای که پیمایش می‌شود ریشه است، بنابراین a ریشه‌ی درخت دودویی T می‌باشد و بعد زیر درخت چپ و در آخر زیر درخت راست پیمایش می‌شود. با توجه به اینکه $i,f,e,b,d$ در درخت T برگ می‌باشد، درخت T را می‌توان به شکل مقابل در نظر گرفت:




پیمایش Postorder درخت T عبارت است از: bdciefgha

---

**۴۹ـ گزینه «۱»** در پیمایش Preorder ابتدا ریشه و سپس زیر درخت چپ و در آخر زیردرخت راست پیمایش می‌شود و در پیمایش Postorder ابتدا زیر درخت‌های چپ و بعد زیر درخت راست و در آخر ریشه پیمایش می‌شود و عملیات ذکر شده باید در زیر درخت‌های چپ و راست به صورت یکسان انجام شود. باتوجه به موارد ذکر شده یکی از درخت‌هایی را که می‌توان به دست آورد، به صورت مقابل است:




در این درخت 3 گره تک فرزندی وجود دارد و می‌دانیم اگر پیمایش inorder برای رسم یک درخت را نداشته باشیم، با توجه به سایر پیمایش‌ها می‌توانیم $2^3 = 8$ (تعداد گره‌های تک فرزندی) درخت رسم کنیم که در این مسئله به صورت می‌توان درخت مورد نظر را رسم کرد.

---

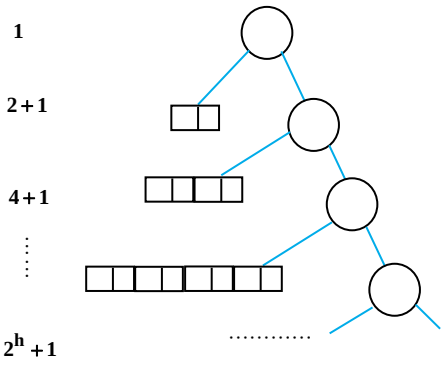
**۵۰ـ گزینه «۲»** یک درخت دودویی به عمق d را پر گویند هرگاه دقیقاً $2^d - 1$ گره داشته باشد و در این صورت در سطح d این درخت دقیقاً $2^{d-1}$ گره وجود دارد که همگی برگ هستند. بنابراین، $2^d - 1 - 2^{d-1}$ تعداد گره‌های غیربرگ می‌باشد.

---

**۵۱ـ گزینه «۱»** درخت گرافی است همبند و بدون دور یعنی دارای حلقه نمی‌باشد.

---

**۵۲ـ گزینه «۴»** با توجه به صورت سؤال، شکل روبه‌رو درختی با کمترین تعداد عنصر و بیشترین ارتفاع را نشان می‌دهد. با توجه به اینکه عمق ریشه صفر در نظر گرفته شده است، در این حالت: $n = 2^{h+1} - 1 + h$




در نتیجه در حالت کلی داریم: $n \le h + 2^{h+1} - 1$

---

**۵۳ـ گزینه «۱»** از بین عناصر موجود مینیمم یا ماکزیمم را به عنوان ریشه انتخاب می‌کنیم. (حاصل در شکل فرقی ایجاد نمی‌کند.) حال در هر سطح، برای انتخاب گرهٔ بعد، عنصری را پیدا می‌کنیم که یا ماکزیمم عناصر جاری باشد یا مینیمم عناصر جاری. از آنجایی که $n-1$ سطح برای انتخاب داریم، $2^{n-1}$ درخت دودویی با ساختار توپولوژی متفاوت خواهیم داشت.

با یک گره تنها می‌توان یک درخت ساخت، پس گزینه‌ی $2\times(n-1)$ حذف می‌شود.

با دو گره تنها دو درخت می‌توان ساخت، پس گزینه‌های $2^n - 1$ و $2\times n - 1$ نیز حذف می‌شوند.

---

**۵۴ـ هیچ‌کدام از گزینه‌ها صحیح نیست.**

هم سؤال و هم جواب‌ها غلط هستند. چرا که در سؤال V نیامده است و جواب‌ها نیز پاسخ صورت سؤالی متفاوت هستند!

---

**۵۵ـ گزینه «۱»** در یک درخت غیر تهی دودویی، اگر تعداد برگ‌ها $n_0$ و تعداد گره‌های درجهٔ 2، برابر $n_2$ باشد، همواره داریم: $n_2 = n_0 - 1$

---

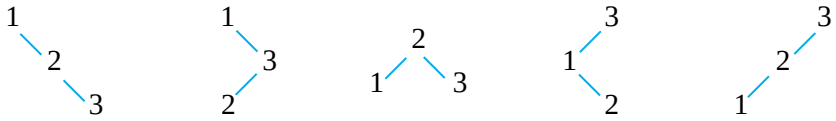
www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



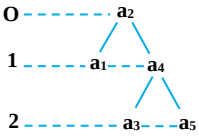
**«درخت جست‌وجوی دودویی»**

**۵۶ـ گزینه «۴»** با داشتن n مقدار متمایز، می‌توان $\frac{\binom{2n}{n}}{n+1}$ درخت جستجوی دودویی متفاوت تشکیل داد. بنابراین با داشتن سه مقدار متمایز می‌توان $\frac{6!}{3!3!(3+1)}=5$ درخت جستجوی دودویی متفاوت تشکیل داد. درخت‌های جستجوی دودویی حاصله عبارتند از:




---

**۵۷ـ گزینه «۱»** درخت زیر را در نظر بگیرید:




با فرض آنکه ریشه دارای عمق صفر باشد.

داریم:

$$C_T=\sum_{i=1}^{5}X_i(\text{depth}(a_i)+1)$$

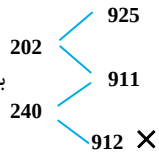
$$=X_1(\text{depth}(a_1)+1)+X_2(\text{depth}(a_2)+1)+X_3(\text{depth}(a_3)+1)+X_4(\text{depth}(a_4)+1)+X_5(\text{depth}(a_5)+1)$$

$$=2(1+1)+1(0+1)+1(2+1)+1(1+1)+1(2+1)=4+1+3+2+3=13$$

---

**۵۸ـ گزینه «۱»** یک درخت جستجوی دودویی، درخت دودویی است که بر چسب عناصر در آن منحصر به فرد بوده و برای هر گره، تمام برچسب‌های گره‌های زیر درخت راست بزرگ‌تر از آن گره و تمام بر چسب‌های گره‌های زیر درخت چپ کوچک‌تر از آن گره باشد. بنابراین باید تک تک گزینه‌ها را بررسی کرده و هر کدام که این قوانین برای آن صادق نباشد جواب مسئله است. در گزینهٔ یک داریم.

بنابر تعریف، درخت جستجوی دودویی 912 نمی‌تواند در زیر درخت چپ 911 باشد. بقیهٔ گزینه‌ها صحیح هستند.

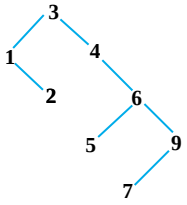



---

**۵۹ـ گزینه «۴»** با توجه به تعریف درخت جستجوی دودویی، چون x برگ و y پدر x است و $a=\text{key}[x]$ و $b=\text{key}[y]$، پس اگر a در زیر درخت راست باشد، a از b بزرگ‌تر است و اگر a در زیر درخت چپ باشد، a از b کوچک‌تر است. گزینه‌های ۱ و ۲ درصورتی صحیح هستند که x برگ سمت راست باشد و چنانچه x برگ سمت چپ y باشد. گزینهٔ ۳ صحیح خواهد بود.

---

**۶۰ـ گزینه «۱»** ابتدا مقادیر 7و 5و 2و 9و 6و 4و 1و 3 را به ترتیب از چپ به راست در یک درخت جستجوی دودویی خالی درج می‌کنیم، داریم:




سپس درخت مقابل را به صورت postorder ، یعنی ابتدا زیر درخت چپ و سپس زیر درخت راست و در آخر ریشه را پیمایش می‌کنیم، داریم:

$$\text{LRV}: 2,1,5,7,9,6,4,3$$

---

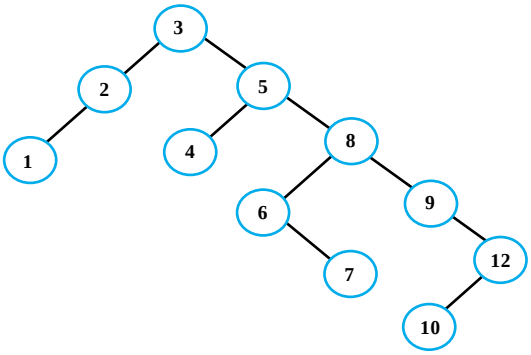
**۶۱ـ گزینه «۴»** در این سؤال سطح ریشه صفر در نظر گرفته شده است. برای اینکه درختی با n عنصر متفاوت، دارای بیش‌ترین ارتفاع ممکن (n-1) باشد، باید همه گره‌ها دارای حداکثر یک فرزند باشند و به عبارت دیگر درخت، n-1 گره تک فرزندی داشته باشد. تعداد کل حالت‌های درخت جستجوی دودویی اریب (با ارتفاع بیشینه) برابر $2^{n-1}$ است.

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir

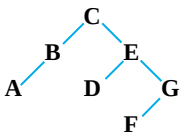


**۶۲ـ گزینه «۳»** از چپ به راست اعداد ذکر شده را در یک درخت جستجوی دودویی درج می‌کنیم:




---

**۶۳ـ گزینه «۴»** اگر CEBGADF را از چپ به راست در یک درخت جستجوی تهی درج کنیم، داریم:

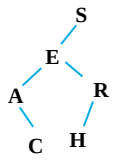



---

**۶۴ـ گزینه «۱»** اگر آدرس گره اول NULL باشد، درخت تهی است که این عمل با مرتبهٔ $O(1)$ انجام می‌شود. بقیهٔ موارد ذکر شده با مرتبهٔ $O(h)$ ($h$ عمق درخت است) انجام می‌شود.

---

**۶۵ـ گزینه «۱»** درخت جستجوی دودویی حاصل از رشته‌ی SEARCH برابر است با:

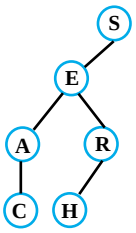



$$\Rightarrow \text{تعداد کل مقایسه‌ها} = 1+2+3+3+4+4 = 17$$

$$\text{متوسط مقایسه‌ها} = \frac{17}{6} = 2.83$$

---

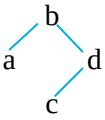
**۶۶ـ گزینه «۱»** روشن است که ابتدا باید درخت را بسازیم. اینکه در یک جستجو چه کلیدی داریم به تعداد مقایسه‌های مختلفی نیاز خواهد بود. با جمع تعداد مقایسه‌های مختلف برای هر کلید و تقسیم بر تعداد آنها میانگین را به دست می‌آوریم. شکل درخت در مقابل آمده است:



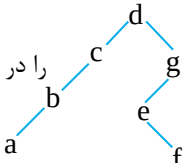
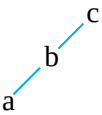
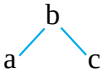
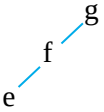
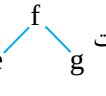
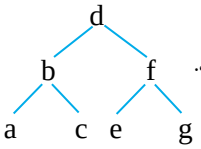

$$1\times 1$$
$$1\times 2$$
$$2\times 3$$
$$\frac{2\times 4}{17} = \text{تعداد کل مقایسه‌ها}$$

$$\text{میانگین} = \frac{17}{6} = 2.83$$

---

**۶۷ـ گزینه «۲»** با توجه به مقادیر $a<b<c<d$، در سه ترتیب badc، bdac، bdca، درخت  را نتیجه می‌دهد.

---

**۶۸ـ گزینه «۲»** درخت  را در نظر بگیرید. ابتدا در زیر درخت چپ  را طوری دوران می‌دهیم که به صورت  درآید، باید توجه داشته باشیم در هر دوران باید درخت همچنان به صورت درخت جستجوی دودویی باشد، پس در زیر درخت راست ابتدا  را به صورت  دوران داده و سپس  را به صورت  دوران می‌دهیم. با این سه دوران درخت به صورت روبرو درمی‌آید.

---

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۶۹ـ گزینه «۲»** در بدترین حالت، عمق درخت جستجوی دودویی برابر با تعداد گره‌های آن می‌باشد و چون حداکثر مقایسه، به اندازهٔ عمق درخت می‌باشد، بنابراین حداکثر تعداد مقایسه برای جستجوی یک عنصر در درخت جستجوی دودویی برابر $O(n)$ می‌شود.

**۷۰ـ گزینه «۴»** می‌دانیم در درخت جستجوی دودویی تمامی کلیدها منحصر به فرد هستند و نیز تمامی کلیدهای زیر درخت راست از کلید ریشه بزرگ‌تر و تمامی کلیدهای زیر درخت چپ از کلید ریشه کوچک‌تر می‌باشد و زیر درخت چپ و راست هر دو خود درخت جستجوی دودویی هستند. در پیمایش Preorder، کلید ریشه ابتدا پیمایش می‌شود، پس به راحتی می‌توان با مرتبهٔ $O(n)$ درخت را ساخت.

**۷۱ـ گزینه «۳»** در درخت جستجوی دودویی بعد از حذف یک گره می‌توان بزرگ‌ترین گره در زیر درخت چپ یا کوچک‌ترین گره در زیر درخت راست را به جای آن گره جایگزین کرد. در درخت ذکر شده در سؤال پس از حذف ریشه می‌توان بزرگ‌ترین گره در زیر درخت چپ، یعنی e و یا کوچک‌ترین گره در زیر درخت راست، یعنی f را جایگزین کرد.

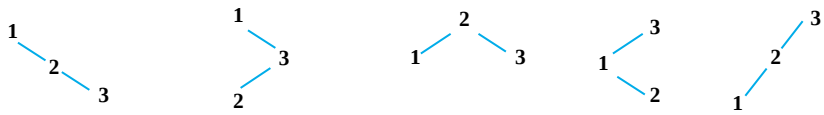
**۷۲ـ گزینه «۲»** حداکثر تعداد مقایسه برای جستجوی یک عنصر در درخت دودویی وابسته به ارتفاع درخت می‌باشد که در بدترین حالت این ارتفاع $n-1$ بوده بنابراین، پیچیدگی جستجو $O(n)$ می‌شود.

**۷۳ـ گزینه «۴»** مرتبه زمانی عمل درج در درخت جستجوی دودویی $O(h)$ است که h عمق درخت می‌باشد و در بهترین حالت که درخت متوازن است $O(\text{Log } n)$ می‌شود و چون n عنصر را باید درج کنیم پس حداقل مرتبه زمانی $\theta(n \text{ Log } n)$ است. در بدترین حالت $\theta(n^2)$ و به هر حال در کل حالات حداقل $\Omega(n\text{Log}n)$ خواهد بود.

**۷۴ـ گزینه «۱»** با داشتن فقط پیمایش inorder، نمی‌توان ریشه را مشخص کرد.

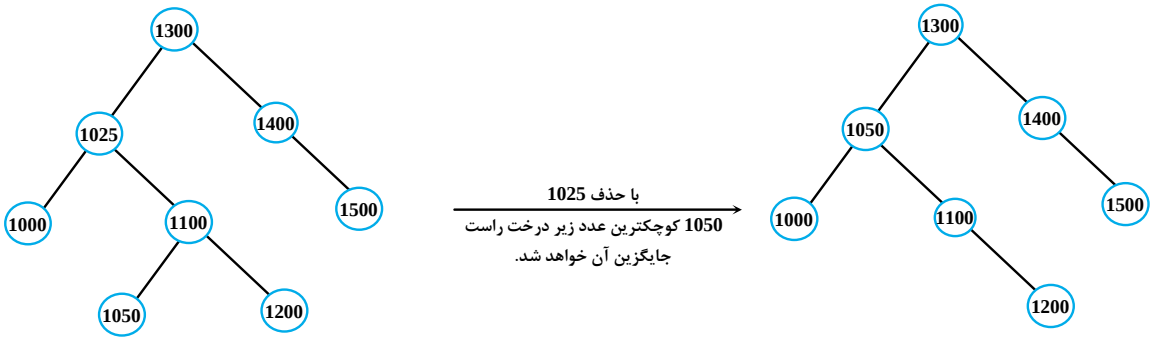
**۷۵ـ گزینه «۴»** برای اینکه یک درخت دودویی، درخت جستجوی دودویی باشد، باید عناصر زیر درخت چپ هر عنصر (عنصر جاری) از آن کوچک‌تر و عناصر زیر درخت راست هر عنصر، از آن بزرگ‌تر باشد. لذا، حداکثر مقدار عناصر زیر درخت چپ عنصر می‌تواند برابر مقدار آن $(t \rightarrow \text{data})$ باشد و کمترین مقدار عناصر زیر درخت سمت راست از مقدار خود عنصر بیشتر باشد.
(یعنی حداقل برابر $t \rightarrow \text{data} + 1$ باشد.) همچنین عناصر سمت چپ عنصر جاری باید حداقل دارای مقدار مجاز تعیین شده برای حداقل مقدار خود عنصر باشد و عناصر سمت راست عنصر جاری باید حداکثر دارای مقدار مجاز تعیین شده برای حداکثر مقدار خود عنصر باشد. بنابراین گزینه «۴» صحیح است.

**۷۶ـ گزینه «۳»**




**۷۷ـ گزینه «۲»** اگر عناصر یک درخت جستجوی دودویی را به صورت inorder پیمایش کنیم حاصل، مرتب شده صعودی مقادیر موجود در درخت است و با قرار دادن این عناصر در یک پشته، بزرگ‌ترین مقدار در بالای پشته قرار می‌گیرد. این مسئله باعث می‌شود بعد از حذف از پشته عناصر نزولی شوند و درخت جستجوی دودویی حاصله یک درخت مورب به چپ خواهد شد.

**۷۸ـ گزینه «۴»** درخت جستجوی دودویی حاصله به صورت مقابل است:




بعد از حذف گرهٔ 1025 کوچک‌ترین کلید در زیر درخت راست به جای آن جایگزینی می‌شود که گره 1050 می‌باشد.
و پیمایش preorder آن عبارت است از: 1300,1050,1000,1100,1200,1400,1500

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir

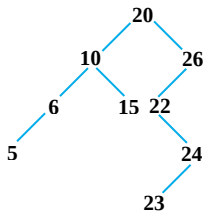


**۷۹ـ گزینه «۴»** در بدترین حالت درخت جستجوی دودویی با 150 گره، مورب بوده یعنی 150 سطح خواهد داشت. در بهترین حالت درخت دودویی کامل متوازن است و $[\log_2^{150}]+1=8$ سطح خواهد داشت.

**۸۰ـ گزینه «۲»** در درخت جستجوی دودویی کلید تکراری نداریم، پس گزینه ۴ غلط است. همچنین تمام کلیدهای زیر درخت راست باید از کلید ریشه بزرگ‌تر باشد، پس گزینهٔ ۱ نیز غلط است. تمام کلیدهای زیر درخت چپ باید از کلید ریشه کوچک‌تر باشد. پس گزینهٔ ۳ نیز غلط می‌باشد.

**۸۱ـ گزینه «۴»** در درخت جستجوی دودویی، تمام مقادیر زیر درخت چپ از ریشه کوچک‌تر و تمامی مقادیر زیر درخت راست از ریشه بزرگ‌تر می‌باشد. اگر درخت جستجویی دودویی را به صورت میانوندی پیمایش کنیم، عناصر به صورت صعودی به دست می‌آید. لذا درخت مورد نظر سؤال را به صورت میانوندی پیمایش کرده و چهارمین حرف سمت چپ معادل چهارمین کوچک‌ترین عنصر خواهد بود. پیمایش میانوندی: D B H E I A F C J G

**۸۲ـ گزینه «۴»** می‌دانیم در درخت جستجوی دودویی برای هر گره، تمام کلیدهای گره‌های زیر درخت راست از کلید آن گره بزرگ‌تر و تمام کلیدهای گره‌های زیر درخت چپ از کلید آن گره کوچک‌تر است. با استفاده از این مطلب و با داشتن پیمایش postorder می‌توان درخت را به راحتی رسم کرد، داریم:




پیمایش preorder این درخت عبارت است از: $20,10,6,5,15,26,22,24,23$

**۸۳ـ گزینه «۲»** در برنامه بازگشتی ISBST مقادیر max,min به ترتیب برابر حداکثر مقداری است که مقدار کلید جاری از آن بزرگ‌تر و حداقل مقداری که مقدار کلید جاری از آن کوچک‌تر است.

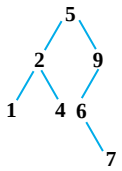
**۸۴ـ گزینه «۳»** تعداد درخت‌های BST ممکن با n کلید عبارتست از:

$$C_n=\frac{\binom{2n}{n}}{n+1}$$

بنابراین:

$$C_4=\frac{\binom{8}{4}}{5}=14$$

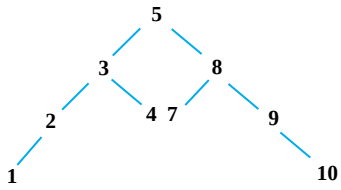
**۸۵ـ گزینه «۳»** ابتدا اعداد داده شده را یکی یکی در یک درخت جستجوی دودویی درج می‌کنیم، داریم:




سپس درخت حاصله را به صورت LRV پیمایش می‌کنیم.

**۸۶ـ گزینه «۱»** در کل، تعداد درخت‌های جستجوی دودویی که می‌توان با 5 کلید مجزا ساخت $\frac{1}{6}\binom{10}{5}=42$ می‌باشد، که البته از این تعداد بعضی از درخت‌های تولید شده دارای عمق 3 است، که با محاسبهٔ درخت‌های تولید شده به عمق 3 جواب حاصل می‌شود.

**۸۷ـ گزینه «۴»** درخت جستجوی دودویی زیر را در نظر بگیرید:




پیمایش پسوندی آن عبارت است از: 1, 2, 4, 3, 7, 10, 9, 8, 5

که آن را در آرایه postfix [1..9] ذخیره کرده‌ایم:

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|---|---|---|---|---|---|---|---|---|
| 1 | 2 | 4 | 3 | 7 | 10 | 9 | 8 | 5 |

به مثال مقابل دقت کنید:

$$i=3\Rightarrow x=4,b=3,a=2\Rightarrow\begin{cases}a<x\\x>b\Rightarrow n=4\ \text{برگ}\\a<b\end{cases}$$

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۸۸ـ گزینه «۳»** چون باید اختلاف عمق برگ‌های درخت حداکثر 1 باشد، بنابراین حداقل عمق برگ $[\text{Log}_2^{36}]=5$ و حداکثر عمق برگ‌ها $[\text{Log}_2^{36}]+1=6$ است، پس $2^5-1=31$ گروه در 5 سطح درخت ذخیره شده و 5 گره بعدی در سطح 6 ام درخت می‌تواند قرار گیرد که در سطح 6ام 32 مکان خالی وجود دارد و 5 گره را می‌توان در هرکدام از این 32 مکان قرار داد. بنابراین تعداد درخت‌های تشکیل شده $\binom{32}{5}$ می‌باشد.

---

**۸۹ـ گزینه «۴»** در اعداد داده شده دو مقدار تکراری داریم و می‌دانیم در درخت جستجوی دودویی کلید تکراری نداریم. بنابراین می‌خواهیم تعداد درخت‌های جستجوی دودویی که با 4 مقدار می‌توان ساخت را مشخص کنیم، بنابر فرمول داریم:

$$\frac{\binom{2n}{n}}{n+1}=\frac{\binom{2\times 4}{4}}{4+1}=\frac{\binom{8}{4}}{5}=\frac{\frac{8!}{4!4!}}{5}=\frac{8\times 7\times 6\times \cancel{5}\times \cancel{4!}}{4!\cancel{4!}\cancel{5}}=\frac{\cancel{8}\times 7\times \cancel{6}}{\cancel{4}\times 3!}=14$$

---

**۹۰ـ گزینه «۲»** در درخت تصمیم‌گیری دودویی، برگ‌ها حالت‌های مختلف ادغام برای لیست‌های 1 و 2 می‌باشد و چون این دو لیست بعد از ادغام ۷ عنصر می‌شوند، باید حالت‌های ادغام این دو لیست را در نظر بگیریم، داریم: $\binom{7}{2}=\binom{7}{5}=\frac{7!}{2!5!}=21$

در واقع هر برگ نشان‌دهنده یک ترتیب از 7عنصر است.

---

**۹۱ـ گزینه «۴»** در برنامه بازگشتی ISBST مقادیر min و max به ترتیب برابر حداکثر مقداری است که مقدار کلید جاری از آن بزرگ‌تر و حداقل مقداری که مقدار کلید جاری از آن کوچک‌تر است.

---

**۹۲ـ گزینه «۳»** می‌دانیم که تعداد درخت دودویی که با $n$ گره می‌توان ساخت برابر جمله $n+1$ از سری کاتالان می‌باشد و برابر است با:

$$b_n=\frac{1}{n+1}\binom{2n}{n}$$

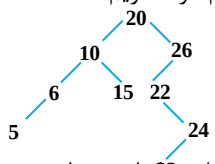
از طرف دیگر می‌دانیم که سری فوق را می‌توان به صورت بازگشتی روبرو نوشت: $b_n=\sum_{i=1}^{n-1} b_{k-1}\,b_{n-k}$

همچنین اعداد سری کاتالان به صورت فاکتوریل، رشد می‌کنند که از $2^n$ بزرگ‌تر هستند. پس عبارت $b_n\in\Omega\binom{n}{2}$ نیز صحیح است.

نادرستی گزینه ۳ بدیهی است. زیرا ترکیب $\binom{2}{n}$ برای $n$های بزرگ‌تر از 2 معنا ندارد.

---

**۹۳ـ گزینه «۳»** به راحتی می‌توان با استفاده از تعریف درخت جستجوی دودویی و پیمایش Postorder آن، درخت را رسم کرد، داریم:




پیمایش Preorder درخت فوق عبارت است از:
$20,10,6,5,15,26,22,24,23$ که از چپ به راست، هفتمین کلید 22 می‌باشد.

---

**۹۴ـ گزینه «۴»** $n$ عدد در بازهٔ $[1,\log n]$ نشان‌دهندهٔ تکراری بودن بعضی از مقادیر است. در درخت جستجوی دودویی مقادیر تکراری نداریم، بنابراین فقط $\log n$ عدد در درخت جستجوی دودویی می‌تواند وجود داشته باشد که در این صورت حداکثر ارتفاع درخت جستجوی دودویی می‌تواند برابر $\log n$ و حداقل برابر $\log(\log n)$ باشد که در این حالت درخت جستجوی دودویی متوازن خواهد بود (AVL).

---

**«هرم»**

**۹۵ـ گزینه «۴»** اگر heap ذکر شده در سؤال به صورت maxheap باشد، یعنی ریشه بزرگ‌ترین مقدار را دارد. پس یافتن ماکزیمم با $O(1)$ انجام می‌شود و به طور کلی اعمال درج و حذف در heap دارای پیچیدگی زمانی $O(\log n)$ است.

---

www.modaresanesharif.ac.ir

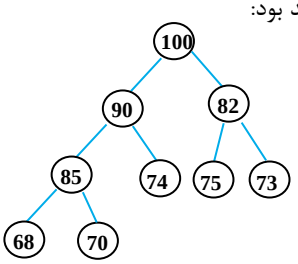
www.modaresanesharif.ac.ir



**۹۶ـ گزینه «۲»** می‌دانیم درخت heap یک درخت دودویی کامل است و بهترین روش برای نشان دادن یک درخت دودویی کامل آرایه می‌باشد. با استفاده از اصل موضوعی ۴، گره پدرِ اندیس i در آرایه در $\left\lfloor \frac{i}{2} \right\rfloor$ است. برای درج عنصری در heap باید ابتدا عنصر را به انتهای heap درج کرده و سپس آن را با اجدادش مقایسه کنیم. (در maxheap) از هر کدام از اجدادش که بزرگ‌تر بود مقدار آن را با آن گرهٔ جد جا به جا می‌کنیم. در الگوریتم داده شده temp عنصر جدید است که باید درج شود، پس باید حلقه تا زمانی که A[i]<temp باشد ادامه یابد.

---

**۹۷ـ گزینه «۱»** شکل درخت حاصل از سؤال به صورت زیر خواهد بود:




عدد 95 به سمت چپ عدد 74 اضافه می‌گردد و سپس با دو عمل تعویض (exchange) جای گره 90 قرار خواهد گرفت.

---

**۹۸ـ گزینه «۲»** می‌توانیم برای به دست آوردن عنصر مینیمم، برگ‌های heap را با هم مقایسه کنیم. کوچک‌ترین برگ عنصر مینیمم است و این کار با حداکثر $\frac{n}{2}$ مقایسه انجام می‌گیرد.

---

**۹۹ـ گزینه «۴»** چهارمین بزرگ‌ترین عنصر می‌تواند در درایه 2 تا 15 باشد. به عنوان مثال در heap زیر عنصر درایه 2 (عدد 8) چهارمین بزرگ‌ترین عنصر است.

| 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|
| 40 | 8 | 30 | 3 | 2 | 20 |

و در heap زیر، عنصر درایه 15 (عدد 65) چهارمین بزرگ‌ترین عنصر است.

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 85 | 60 | 75 | 56 | 50 | 40 | 70 | 45 | 35 | 30 | 26 | 20 | 10 | 2 | 65 |

**نکته:** در یک max-heap همواره شرایط زیر برقرار است:

بزرگ‌ترین عنصر در سطح اول (درایه 1) قرار دارد.

دومین بزرگ‌ترین عنصر در سطح دوم (درایه 2 یا 3) قرار دارد.

سومین بزرگ‌ترین عنصر در سطح دوم یا سوم (درایه 2، 3، 4، 5، 6 یا 7) قرار دارد.

چهارمین بزرگ‌ترین عنصر در سطح دوم، سوم یا چهارم (درایه 2 تا 15) قرار دارد.

....

i اُمین بزرگ‌ترین عنصر در سطح دوم تا i ام قرار دارد. $(i \le \frac{n}{2})$

---

**۱۰۰ـ گزینه «۲»** یک heap یک درخت دودویی کامل است و می‌دانیم عمق درخت دودویی کامل با n رأس، $\left\lfloor \log_2^n \right\rfloor + 1$ می‌باشد. بنابراین عمق درخت ذکر شده در مسئله، $\left\lfloor \log_2^{64} \right\rfloor + 1 = 6 + 1 = 7$ است و این درخت دارای 7 سطح است که در سطح هفتم تنها یک برگ وجود دارد. برای این که مقدار موجود در این برگ، بیشترین مقدار ممکن باشد، لازم است که مسیر این برگ تا ریشه، بزرگ‌ترین عناصر مجموعه را شامل شود. بنابراین میان ریشه و برگ مورد نظر، مسیر به صورت 64-63-62-61-60-59-58 خواهد بود. پس مقدار این برگ نمی‌تواند از 58 بزرگ‌تر باشد.

---

**۱۰۱ـ گزینه «۴»** برای تبدیل یک درخت دودویی به یک min – heap از انتهای آرایه (برگ‌ها) شروع کرده و هر گره را با اجدادش مقایسه می‌کنیم، اگر از اجدادش کوچک‌تر باشد مقدار آن را با مقدار کلید پدر جابجا می‌کنیم.

---

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۱۰۲ـ گزینه «۱»** عمق max–heap از فرمول $[\log_2^n]+1$ به دست می‌آید، max – heap ذکر شده در سؤال 1023 عضو دارد، بنابراین $[\log_2^{1023}]+1=9+1=10$ عمق این max – heap است. اگر در ریشه بزرگ‌ترین مقدار باشد (1023) با توجه به عمق درخت در آخرین سطح می‌تواند حداکثر 10 عدد بیشتر از 1000 باشد.

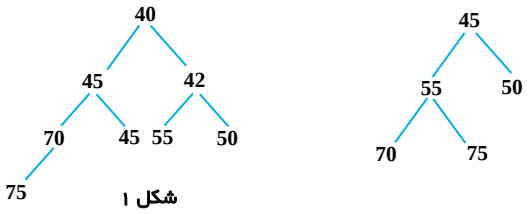
**۱۰۳ـ گزینه «۳»** چون اعداد متمایز هستند، دومین بزرگ‌تر می‌تواند فقط در درایه‌های 2 یا 3 باشد و بنابراین سومین بزرگ‌تر می‌تواند در درایه‌های 2 یا 3 یا 4 و 5 و 6 و یا 7 باشد و چهارمین بزرگ‌تر می‌تواند در درایه‌های 2 یا 3 یا 4 و 5 و 6 و 7 و یا 8 تا 15 باشد و نمی‌تواند در درایه‌های 16 تا 31 قرار گیرد.

**۱۰۴ـ گزینه «۳»** برای تبدیل یک max – heap به min – heap می‌توان از گره‌های برگ شروع کرد و هر گرهٔ برگ را با پدرش مقایسه کرد. در صورت کوچک‌تر بودن جابه‌جایی انجام داد که این جابه‌جایی با مرتبهٔ $O(\log n)$ انجام می‌شود و اگر برای تمام گره‌ها این کار انجام شود، مرتبهٔ اجرایی $O(n\log n)$ خواهد شد.

**۱۰۵ـ گزینه «۳»**

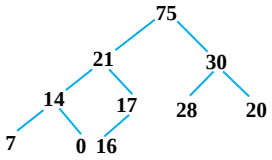
ابتدا کلیدها را در minheap درج می‌کنیم. (شکل ۱)

سپس از ریشه یکی یکی عمل حذف را انجام می‌دهیم (سه مرتبه) و در نتیجه داریم:



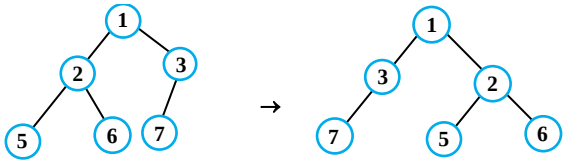
شکل ۱

**۱۰۶ـ گزینه «۴»** با قرار دادن گزینه‌ها می‌توان دریافت که فقط گزینهٔ (۴) سبب می‌شود که آرایهٔ T یک maxheap شود. با استفاده از گزینهٔ (۴) داریم:

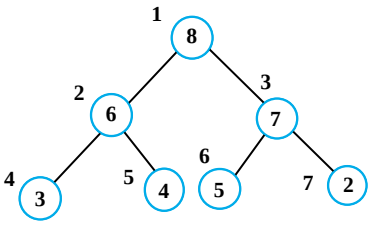



**۱۰۷ـ گزینه «۳»** چون در min – heap ریشه از فرزندان کوچک‌تر یا مساوی است، با تعویض زیر درخت‌های چپ و راست باز هم ریشه از فرزندان کوچک‌تر یا مساوی خواهد بود. پس اگر جای زیر درخت چپ و راست را عوض کنیم، شرط مذکور تغییر نمی‌کند ولی ممکن است شرط کامل بودن نقض شود و درخت حاصل دیگر کامل نباشد، بنابراین heap نخواهد بود.

**۱۰۸ـ گزینه «۳»** ممکن است درخت جدید یک درخت دودویی کامل نباشد.




**۱۰۹ـ گزینه «۲»**




$n=7$

$i=4 \rightarrow j=4+2^{[\text{Log}_2^4]-1}=6$

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir

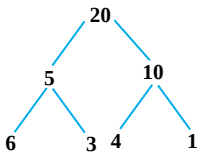


**۱۱۰ـ هیچ‌کدام از گزینه‌ها صحیح نیست.** در درخت دودویی جستجوی مقابل

، کوچک‌ترین گره 15 است که برگ نیست و نیز عمل درج در یک درخت دودویی جستجو در زمان حداکثر $O(h)$ ($h$ عمق درخت) انجام می‌شود. پیمایش inorder یک maxheap و preorder یک درخت دودویی جستجو، هیچ ترتیبی به وجود نمی‌آورد.

---

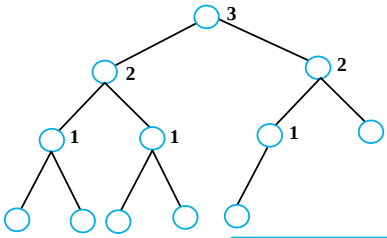
**۱۱۱ـ گزینه «۱»** برای گزینه یک درخت ذخیره شده را رسم می‌کنیم، داریم:




در این درخت گره با کلید 5 از فرزندش گره با کلید 6 کوچک‌تر است، بنابراین درخت مذکور Maxheap نمی‌باشد.

---

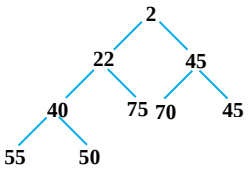
**۱۱۲ـ گزینه «۲»** درخت مورد نظر را می‌توان به صورت زیر در نظر گرفت، که حداکثر تعداد مقایسه‌های لازم برای هر گره بر روی آن نوشته شده است:




بنابراین حداکثر 10 مقایسه انجام خواهد شد.

---

**۱۱۳ـ گزینه «۱»** می‌دانیم برای درج در یک min-head ابتدا باید دودویی کامل بودن درخت حفظ شود، پس یک گره به آخرین سطح اضافه می‌کنیم، سپس کلید گره مورد نظر را با اجدادش مقایسه می‌کنیم، از هر کدام که کوچک‌تر بود با کلید آن گره باید جابجایی شود، داریم:




---

**۱۱۴ـ گزینه «۲»** سومین کوچک‌ترین کلید نمی‌تواند در اندیس یک باشد، زیرا کلیدها متمایز بوده و ریشه باید کوچک‌ترین مقدار را داشته باشد. ولی اگر دومین کوچک‌ترین عنصر در مکان دوم یا سوم باشد، سومین کوچک‌ترین عنصر می‌تواند در مکان سوّم یا دوّم قرار گیرد. بنابراین گزینه دو، بهترین جواب است.

---

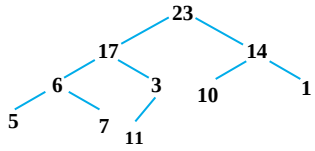
**۱۱۵ـ گزینه «۴»** ساختمان داده‌هایی نداریم که ساخت آن با مرتبه اجرایی $O(n)$ باشد و حذف بزرگ‌ترین عنصر در آن $O(1)$ و درج و حذف عنصر دلخواه در آن با مرتبه اجرایی $O(Logn)$ انجام شود.

---

**۱۱۶ـ گزینه «۴»** ابتدا گرهٔ 85 را به عنوان آخرین عنصر یعنی اندیس دهم آرایه در نظر می‌گیریم، سپس آن را با اجدادش یعنی $[i/2]$ مقایسه می‌کنیم. چون heap داده شده، max heap می‌باشد، باید 85 از اجدادش کوچک‌تر یا مساوی باشد، اگر نبود آن را با اجدادش جابجا می‌کنیم. پس اولین جابجایی با گرهٔ پنجم یعنی مقدار 64 خواهد بود و دومین جابجایی با گره دوم یعنی 80 می‌باشد.

---

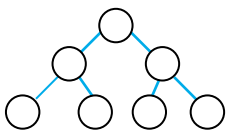
**۱۱۷ـ گزینه «۴»** با رسم درخت مورد نظر داریم:




بنابراین باید مقدار 6 به 12 و 3 به 15 تبدیل شود.

---

**۱۱۸ـ گزینه «۲»** یک $\min-heap$ با 7 گره یک درخت دودویی پر به عمق 3 است، مانند شکل مقابل:



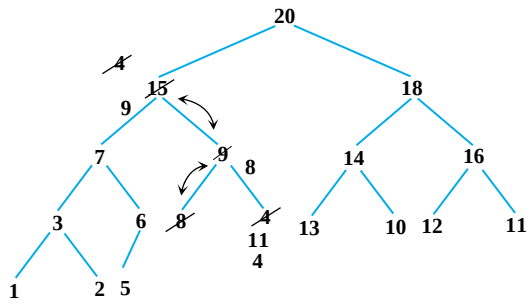
بنابراین عدد 1 فقط می‌تواند در ریشه باشد و عدد 2 فقط در سطح دوم و به عنوان فرزندان عدد 2 هر کدام از 5 عدد دیگر می‌تواند باشد. بنابراین کل حالات برابر است با: $5\times4\times2\times2=80$

---

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir






**۱۱۹ـ گزینه «۲»** در عمل Change(2,4) باید کلید A[2] را به 4 تغییر دهیم که در این صورت برای آنکه maxheap مجدداً تشکیل شود باید 4 با مقدار بزرگ‌ترین فرزند، یعنی 9 و بعد با 8 عوض شود و در عمل Change(11,16) باید کلید A[11] را به 16 تغییر دهیم که در این صورت برای آنکه maxheap مجدداً تشکیل شود، باید 16 با پدرش 8 و بعد با 9 عوض شود، بنابراین در کل 4 جابه‌جایی لازم است.

---

**۱۲۰ـ گزینه «۱»** یافتن یک عنصر با کلید مشخص در یک max-hap در بدترین حالت n مقایسه لازم دارد و از مرتبه $O(n)$ است.

---

**۱۲۱ـ گزینه «۲»** ابتدا باید اندیس 12را با اجدادش یعنی $\left[\frac{12}{2}\right]=6$ و $\left[\frac{6}{2}\right]=3$ و $\left[\frac{3}{2}\right]=1$ جابجا کنیم که نتیجه به صورت:

$12,2,1,4,5,3,7,8,9,10,11,6$ درمی‌آید. سپس اندیس 11 را با اجدادش یعنی $\left[\frac{11}{2}\right]=5$ و $\left[\frac{5}{2}\right]=2$، جابجا می‌کنیم که نتیجه به صورت:

$12,11,1,4,2,3,7,8,9,10,11,6$ در می‌آید.

اندیس 10را با پدرش یعنی $\left[\frac{10}{2}\right]=5$ جابجا می‌کنیم، داریم: $12,11,1,4,10,3,7,8,9,2,5,6$

اندیس 9 را با پدرش یعنی $\left[\frac{9}{2}\right]=4$ جابجا می‌کنیم، داریم: $12,11,1,9,10,3,7,8,4,2,5,6$

اندیس 8 نیاز به جابجایی ندارد و اندیس 7، با پدرش یعنی $\left[\frac{7}{2}\right]=3$ باید جابجا شود، داریم: $12,11,7,9,10,3,1,8,4,2,5,6$

و آخرین جابجایی مربوط به مقدار جدید اندیس 12یعنی عدد 6 و مقدار جدید اندیس 6 یعنی عدد 3 می‌باشد و داریم:

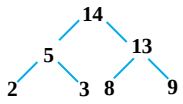
$12,11,7,9,10,6,1,8,4,2,5,3$ که در این لیست عنصر پنجم برابر عدد 10 می‌باشد.

---

**۱۲۲ـ گزینه «۱»** در فراخوانی متد A از ساختمان داده‌های X هر بار یک واحد به Count اضافه می‌شود و سپس عنصر $\langle Count, X\rangle$ در Maxheap درج می‌شود که چون معیار مقایسه عناصر Count است، پس هر بار عنصر جدید به ریشه Maxheap منتقل می‌شود، زیرا بزرگ‌ترین مقدار می‌باشد و می‌دانیم که در عمل حذف از Maxheap فقط می‌توان ریشه را حذف کرد. بنابراین آخرین عنصری که درج می‌شود اول باید حذف شود که این ساختمان داده stack می‌باشد.

---

**۱۲۳ـ گزینه «۲»** در یک درخت max heap مقدار کلید هر گره از فرزندانش بزرگ‌تر یا مساوی است، بنابراین هر گونه جابجایی بین فرزندان و گره پدر باعث می‌شود که دیگر درخت، max heap نباشد. پس گزینه‌های ۱و۳و۴ غلط است. البته گزینه ۲ نیز می‌تواند غلط باشد. به مثال روبرو دقت کنید:

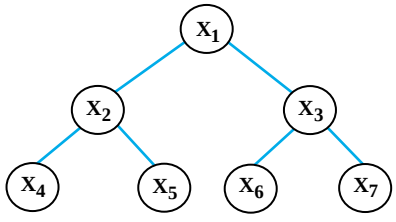



در مثال فوق، اگر جای دو فرزند گره 14 را عوض کنیم، دیگر درخت max heap نمی‌باشد.

---

**۱۲۴ـ گزینه «۴»** heap را به صورت زیر در نظر می‌گیریم.

با توجه به ساختار Min – heap داریم:




$x_1 = 1$

$2 \le x_2 < x_4 \qquad 2 \le x_3 < x_6$

$2 \le x_2 < x_5 \qquad 2 \le x_3 < x_7$

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



حالات مختلف را به صورت زیر شمارش می‌کنیم.

| مجموع حالات | $X_7$ | $X_6$ | $X_5$ | $X_4$ | $X_3$ | $X_2$ |
|---|---|---|---|---|---|---|
| ۲۴ | ۱ حالت | ۲ حالت | ۳ حالت | ۴ حالت | $=3$ | $=2$ |
| مشابه بالا = ۲۴ | — | — | — | — | $=2$ | $=3$ |
| ۶ | ۱ حالت | ۲ حالت | ۳ حالت | $=3$ | $=4$ | $=2$ |
| مشابه بالا = ۶ | — | — | $=3$ | — | $=4$ | $=2$ |
| ۱۲= مشابه دو حالت قبل | — | — | — | — | $=2$ | $=4$ |
| ۲ | ۱ حالت | ۲ حالت | $=4$ | $=3$ | $=5$ | $=2$ |
| ۲= مشابه حالت قبل | — | — | $=3$ | $=4$ | $=5$ | $=2$ |
| ۴= مشابه دو حالت قبل | — | — | — | — | $=2$ | $=5$ |

تعداد کل حالات $= 24+24+6+6+12+2+2+4=80$

---

**۱۲۵ـ گزینه «۴»** حذف گره از maxheap یعنی حذف ریشه و دوباره مرتب‌سازی maxheap که مرتبه زمانی آن $\log n$ است برای حذف عنصر I عناصر ما قبل آن نیز می‌بایست حذف شوند پس در بدترین حالت $i=n$ است و مرتبه زمانی آن $n\log n$ می‌شد.

---

**۱۲۶ـ گزینه «۱»** در $\min-heap$ همواره کوچک‌ترین مقدار در ریشه درخت است و حذف آن با مرتبهٔ زمانی $O(\log n)$ انجام می‌شود، ولی در $\max-heap$ برای جستجوی کوچک‌ترین مقدار باید n گره بررسی شود.

---

**۱۲۷ـ گزینه «۱»** انجام عمل حذف min از min heap دارای زمان $O(\log n)$ و از max heap دارای زمان $O(n)$ است.

---

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



# فصل ششم

# «داده‌ساختارهای پیشرفته»

## تست‌های طبقه‌بندی شده کنکوری فصل ششم

**۱ـ** درخت **AVL** یک درخت دودویی است که ارتفاع دو زیر درخت هر گره آن حداکثر یک واحد با هم اختلاف دارد. (ارتفاع یک درخت تهی را ۱- فرض کنید). اگر **T(h)** کمترین تعداد گره برای یک درخت **AVL** به ارتفاع **h** باشد، کدام یک از رابطه‌های بازگشتی زیر برای **T(h)** درست است؟ **(T(۰) = 1 و T(1) = 2)** (مهندسی کامپیوتر ـ سراسری ۷۹)

۱) $T(h) = 2T(h-2) + 1$

۲) $T(h) = T(h-1) + T(h-2) + 1$

۳) $T(h) = 2T(h-1) + 1$

۴) $T(h) = T(\lfloor h/2 \rfloor) + T(\lceil h/2 \rceil) + 1$

**۲ـ** ساختمان داده‌ای را بر روی مجموعه **A** از اعداد صحیح در نظر بگیرید که اعمال درج **(Insert)**، حذف **(Delete)** و پیدا کردن نزدیک‌ترین **(Findclosest)** را فراهم می‌آورد. منظور از **Findclosest(x)** پیدا کردن $y \in A$ است که $|x-y|$ نسبت به بقیه **y**ها کمینه باشد. فرض کنید که **T** بیشترین زمان اجرای اعمال فوق در بدترین حالت باشد. در این صورت کدام ساختمان داده کمترین مقدار **T** را خواهد داشت؟ (مهندسی کامپیوتر ـ سراسری ۸۱)

۱) heap

۲) لیست نامرتب

۳) لیست مرتب

۴) درخت جست‌وجوی دودویی متوازن

**۳ـ** درخت **AVL** یک درخت دودویی جستجو است که اختلاف ارتفاع دو زیر درخت هر عنصر در آن حداکثر **1** باشد. با عناصر **1، 2، 3، 4 و 5** حداکثر چند تا درخت **AVL** می‌توان ساخت ؟ ( ارتفاع درخت تهی **1**- فرض می‌شود ) (مهندسی کامپیوتر ـ سراسری ۸۵)

۱) 5

۲) 6

۳) 7

۴) 8

**۴ـ** یک درخت **AVL** درختی است متوازن که در تمامی گره‌ها، اختلاف عمق بین دو زیر درخت آن گره حداکثر یک باشد. اگر این درخت را از گره دلخواه **x**، **split** نمائیم. کمترین هزینه برای حفظ دو درخت به وجود آمده با ساختار **AVL** چقدر است ؟ ( **n** را تعداد عناصر درخت در نظر بگیرید). (مهندسی کامپیوتر ـ سراسری ۸۶)

۱) $n$

۲) $n^2$

۳) $n\log(n)$

۴) $\log(n)$

**۵ـ** یک درخت **AVL** با ارتفاع **4** حداقل چند گره خواهد داشت (یک درخت **AVL** درختی است دوتایی **(binary)** که اختلاف ارتفاع زیر درخت سمت راست و چپ آن حداکثر یک باشد)؟ (مهندسی فناوری اطلاعات (IT) و مهندسی کامپیوتر ـ آزاد ۸۷)

۱) 11

۲) 12

۳) 9

۴) 13

**۶ـ** یک درخت **Max Heap** تحت کدام یک از عملیات زیر همچنان **Max Heap** می‌ماند؟ فرض کنید که عمل خواسته شده روی یک یا چند گره دلخواه درخت اعمال شود؟ (علوم کامپیوتر ـ سراسری ۸۹)

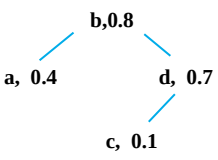
۱) Left Rotate (چرخش به چپ)

۲) Mirror (تعویض بچه‌های چپ و راست گره)

۳) Right rotate (چرخش به راست)

۴) Exchange (تعویض مقدار موجود در گره با بزرگ‌ترین فرزند)

**۷ـ** داده ساختار **treap** درخت دودویی است که بر روی **n** عنصر با کلیدهای متمایز ساخته می‌شود. هر عنصر علاوه بر کلید یک مؤلفه‌ی «امتیاز» دارد که عدد حقیقی بین **0, 1** می‌باشد. **treap** طوری طراحی می‌شود که از نظر کلیدها دودویی جست‌وجو و از نظر امتیازها یک هرم بیشینه **(max-heap)** است. مثلاً عناصر **d, c, b, a** با امتیازهای به ترتیب برابر **0.7, 0.1, 0.8, 0.4** در یک **treap** زیر قرار می‌گیرند.




توجه کنید که فقط خاصیت هرم **(heap)** بین امتیاز عناصر مهم است (یعنی امتیاز هر عنصر از پدرش بیشتر نیست). کدام یک از گزینه‌های زیر در مورد **treap** درست است؟ (مهندسی کامپیوتر ـ سراسری ۸۷)

۱) اگر کلیدها و امتیاز عناصر متمایز باشند، فقط یک treap برای آن وجود دارد.

۲) اگر کلیدها و امتیاز عناصر متمایز باشند، ممکن است treap برای آن وجود نداشته باشد.

۳) اگر کلیدها متمایز ولی امتیازها در حالت کلی باشند، ممکن است treap برای آن وجود نداشته باشد.

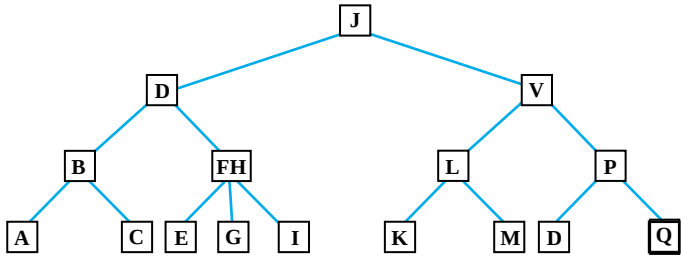
۴) اگر کلیدها و امتیاز عناصر متمایز باشند، treap برای آن وجود دارد ولی ممکن است جواب تک نباشد.

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۸ـ درخت زیر یک B-Tree با درجه مینیمم $t = 2$ است. بعد از حذف D نود E در چه عمقی قرار خواهد داشت؟** (مهندسی کامپیوتر ـ سراسری ۸۸)



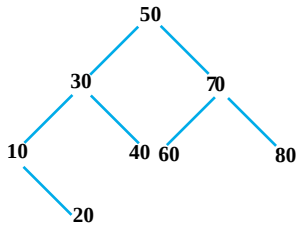

۱) 0
۲) 1
۳) 2
۴) 3

**۹ـ n عدد صحیح بین ۱ تا $\sqrt{n}$ داده شده که ممکن است یک عدد بیش‌تر از یک بار آمده باشد و فرقی بین آن‌ها نیست. داده ساختاری می‌خواهیم تا با مصرف حافظه‌ی کم، اعمال درج، حذف و جستجو را در زمان کمی انجام دهد. بهترین داده ساختار برای این مسئله کدام یک از ویژگی‌های زیر را دارد؟ (بهترین جواب ممکن را علامت بزنید.)** (مهندسی کامپیوتر ـ سراسری ۹۰)

۱) درج، حذف و جستجو در $O(\lg n)$ و میزان حافظه $\sqrt{n}$
۲) درج، حذف و جستجو در $O(1)$ و میزان حافظه $n$
۳) درج، حذف و جستجو در $O(1)$ و میزان حافظه $\sqrt{n}$
۴) درج، حذف از $O(\lg n)$ و جستجو در $O(1)$ و میزان حافظه $\sqrt{n}$

**۱۰ـ اگر به درخت AVL زیر کلید 15 اضافه شود، کدام گزینه بخشی از پیمایش Preorder درخت حاصل خواهد بود؟**

(مهندسی فناوری اطلاعات (IT) ـ سراسری ۹۰)




۱) 30 و 40 و 15 و 10 و 20
۲) 40 و 15 و 10 و 20 و 30
۳) 40 و 20 و 10 و 15 و 30
۴) 40 و 15 و 20 و 10 و 30

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



## پاسخنامه تست‌های طبقه‌بندی شده کنکوری فصل ششم

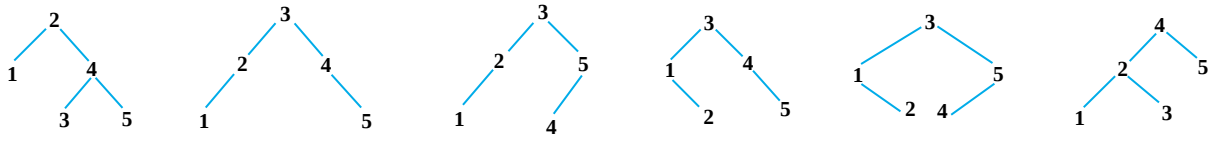
**۱ـ گزینه «۲»** درخت AVL، درخت جستجوی دودویی است که ارتفاع آن متوازن باشد، درخت با ارتفاع متوازن درختی است که حداکثر اختلاف ارتفاع دو زیر درخت چپ و راست آن یک است. با توجه به اینکه $T(1)=2$ ، $T(0)=1$ و ارتفاع یک درخت تهی برابر $-1$ ، یعنی: $T(-1)=0$ تنها گزینه (۲) صحیح می‌باشد.

---

**۲ـ گزینه «۴»** جستجو در یک heap، لیست پیوندی مرتب و نامرتب در بدترین حالت دارای پیچیدگی زمانی $O(n)$ است. ولی در یک درخت جستجوی دودویی متوازن به ارتفاع h برابر $O(h)$ است و می‌دانیم در درخت جستجوی دودویی متوازن h برابر $[\log_2^n]+1$ بوده پس پیچیدگی زمانی $O(\log n)$ می‌باشد.

---

**۳ـ گزینه «۲»** درخت‌های AVL ساخته شده توسط عناصر 1و2و3و4و5 عبارتند از:




---

**۴ـ گزینه «۴»** پس از حذف یک زیر درخت از AVL دو زیر درخت بوجود می‌آید که با زمان $O(\log_2^n)$ می‌توان آنها را متوازن نمود.

---

**۵ـ گزینه «۲»** حداقل تعداد گره‌های یک درخت AVL به عمق k از رابطه زیر بدست می‌آید:

$$T(k)=T(k-1)+T(k-2)+1;\quad T(0)=1\ ,\ T(1)=2$$
$$\Rightarrow T(2)=4\ ,\ T(3)=7\ ,\ T(4)=12$$

احتمالاً منظور طراح سوال از ارتفاع، عمق بوده است.

---

**۶ـ گزینه «۲»** یک درخت Max Heap با تعویض بچه‌های چپ و راست هرگره همچنان Max Heap می‌ماند. اما با چرخش به راست، چرخش به چپ و یا تعویض مقدار موجود در گره یا بزرگ‌ترین فرزند در صورتی که کلیدها متفاوت باشند خاصیت Max Heap از بین می‌رود.

---

**۷ـ گزینه «۱»** اگر فقط خاصیت heap یعنی هر عنصر از پدرش بیشتر نباشد مهم باشد، گزینه (۱) صحیح است، ولی اگر بخواهیم درخت دودویی کامل نیز باشد گزینه (۲) صحیح خواهد بود.

---

**۸ـ گزینه «۱»** درخت B-tree یک درخت جستجوی است و می‌دانیم برای حذف عنصری از یک درخت جستجو این عنصر باید با بزرگ‌ترین عنصر در زیر درخت چپ و یا کوچک‌ترین عنصر در زیر درخت راست جابه‌جا شود.

---

**۹ـ گزینه «۱»** چون اعداد بین 1 تا $\sqrt{n}$ می‌باشد، بنابراین در این n عدد، تکراری داریم و بهترین ساختمان داده‌ها برای نگهداری این n عدد، درخت جستجوی دودویی می‌باشد که متوازن باشد (AVL). در این درخت عملیات درج و حذف و جستجو با مرتبهٔ اجرایی $O(\log n)$ انجام می‌شود.

---

**۱۰ـ گزینه «۳»** در ابتدا گره 15 را به درخت اضافه می‌کنیم که فرزند چپ گره 20 می‌شود. درخت از حالت متوازن خارج شده پس با دوران درخت را متوازن می‌کنیم که پس از دوران گره 15 جای گره 10 قرار می‌گیرد و گره 10 فرزند چپ 15 و گره 20 فرزند راست 15 می‌گردد. حال پیمایش پیش ترتیب درخت به صورت زیر می‌شود:

50 , 30 , 15 , 10 , 20 , 40 , 70 , 60 , 80

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



# فصل هفتم

# «درهم‌سازی»

## تست‌های طبقه‌بندی شده کنکوری فصل هفتم

**۱ـ در جدول درهم سازی (hashing) با وارسی خطی (linear probing)، اگر تابع درهم سازی برای هفت عنصر ورودی به صورت زیر باشد،**

| key | A | B | C | D | E | F | G |
|---|---|---|---|---|---|---|---|
| hash | 3 | 5 | 3 | 4 | 5 | 6 | 3 |

**کدام یک از گزینه‌های زیر نمی‌تواند حاصل درج این عناصر با هر ترتیب دلخواه در آرایه 7 تایی H[0..6] (که در ابتدا تهی است) باشد؟**

(مهندسی کامپیوتر ـ سراسری ۸۲)

۱) H[0..6]=[E F G A C B D]
۲) H[0..6]=[C E B G F D A]
۳) H[0..6]=[B D F A C E G]
۴) H[0..6]=[C G B A D E F]

**۲ـ جدول درهم‌سازی (hashing table) با اندازه‌ی 2n را در نظر بگیرید که تصادم به صورت لیست پیوندی حل می‌شود. فرض کنید که به تعداد $\frac{n}{8}$ عنصر در جدول هستند. با فرض آن که تابع درهم‌سازی (hashing function) با احتمال برابر هر عنصر را به یک درایه از جدول نگاشت می‌کند، متوسط تعداد عناصر در هر درایه از جدول چند است؟**

(مهندسی کامپیوتر ـ سراسری ۸۴)

۱) 1/8
۲) 1/4
۳) 1/2
۴) 1/16

**۳ـ کدام یک از ترتیب‌های زیر ساختمان داده‌های همه منظوره برای ذخیره مرتب اشیاء (براساس مقدار کلید) را به ترتیب از کندترین به سریع‌ترین لیست کرده است؟**

(مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۵)

۱) جدول درهم‌سازی (hash)، آرایه‌های مرتب، لیست‌های زنجیره‌ای مرتب
۲) آرایه‌های مرتب، درخت‌های جستجوی دودویی، لیست‌های زنجیره‌ای، heap
۳) آرایه‌های مرتب، لیست‌های زنجیره‌ای مرتب، درخت‌های جستجوی دودویی
۴) آرایه‌های نامرتب، لیست‌های زنجیره‌ای نامرتب، درخت‌های جستجوی دودویی

**۴ـ اگر یک جدول درهم‌سازی (Hash) با اندازه M باشد، احتمال بروز تصادم (Collision) با ورود یک مقدار جدید در خانه iام چیست؟**

(مهندسی کامپیوتر ـ آزاد ۸۵)

۱) وابسته به تابع Hash استفاده شده می‌باشد.
۲) بستگی به ورودی جدید و بزرگی (M) دارد.
۳) $\frac{i}{M-1}$
۴) $\frac{i}{M}$

**۵ـ فرض کنیم اندازه یک جدول درهم‌سازی (Hash Table) برابر K باشد، در این‌صورت احتمال بروز تصادم (Collision) هنگام ورود یک مقدار جدید در خانه jام کدام است؟**

(مهندسی کامپیوتر ـ آزاد ۸۶)

۱) $\frac{j}{K}$
۲) $\frac{K}{j}$
۳) $\frac{j-1}{K}$
۴) با توجه به نوع تابع درهم‌سازی (Hash) استفاده شده این احتمال تعیین می‌گردد.

**۶ـ اگر اندازه یک hash table برابر با K باشد، احتمال بروز تصادم (collision) هنگام درج یک مقدار جدید در مکان iام جدول برابر است با:**

(مهندسی کامپیوتر ـ آزاد ۸۷)

۱) به تابع hash استفاده شده بستگی دارد.
۲) $\frac{i}{K}$
۳) $\frac{K}{i}$
۴) $\frac{i-1}{K}$

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۷ـ** فرض کنید که **n** عنصر با کلیدهای مجزا از هم را با استفاده از یک تابع در هم‌سازی ساده و یکنواخت در یک آرایه به اندازه‌ی **m** درج می‌کنیم. با این تابع، احتمال آن که دو عنصر دلخواه و متفاوت به یک درایه نگاشت شوند برابر است. میانگین تعداد برخوردهای دو عنصر چقدر است؟

(مهندسی کامپیوتر ـ سراسری ۸۸)

۱) $\Theta(\frac{n}{m})$　　۲) $\Theta(\frac{m}{n})$　　۳) $\Theta(\frac{n^2}{m})$　　۴) $\Theta(\frac{n^2}{m^2})$

**۸ـ** **H** یک جدول درهم‌سازی **(Hash table)** است که دارای **m** خانه است و در آن **n** عنصر ذخیره می‌شود. برای رفع برخورد، این جدول از رویه زنجیر **(collision resolution by chaining)** بهره می‌جوید. متوسط و بدترین زمان جستجو برای یک عنصر چقدر است؟

(مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۸)

۱) $\theta\left(\frac{n}{m}\right),\theta(n)$　　۲) $\theta\left(\frac{m}{n}\right),\theta(m)$　　۳) $\theta\left(\frac{n}{m}\right),\theta\left(\frac{n}{m}\right)$　　۴) $\theta\left(\frac{m}{n}\right),\left(\frac{n}{m}\right)$

**۹ـ** فرض کنید که یک جدول درهم سازی به اندازه **80** (هشتاد) از روش آدرس‌دهی خطی باز استفاده می‌کند. ابتدا جدول خالی بوده است و تنها عملیات اضافه کردن و جستجو روی جدول انجام شده است. در حال حاضر وضعیت درایه‌های **45** تا **56** جدول به صورت زیر است. اعداد بالای آرایه، اندیس درایه‌ها و اعداد پایین آرایه خروجی تابع درهم سازی است. اگر یک بار دیگر از جدول درهم‌سازی خالی شروع کنیم و همان عملیات را به غیر از اضافه کردن کلید **e** دوباره انجام دهیم، در درایه‌ی **50** چه کلیدی قرار می‌گیرد؟

(مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۹)

| 45 | 46 | 47 | 48 | 49 | 50 | 51 | 52 | 53 | 54 | 55 | 56 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | a | b | c | d | e | f | g | h | i | j | |
| | 46 | 46 | 46 | 47 | 46 | 51 | 47 | 46 | 48 | 49 | |

۱) i
۲) h
۳) f
۴) g

**۱۰ـ** فرض کنید که در ساخت جدول درهم‌سازی زیر از روش آدرس‌دهی خطی باز استفاده شده است. اگر تابع **Hash** به صورت باقیمانده تقسیم بر 5 تعریف شده باشد، کدام گزینه نمی‌تواند ترتیب صحیح درج عناصر در جدول باشد؟ ترتیب از چپ به راست است.

(مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۹)

| | |
|---|---|
| 0 | 10 |
| 1 | 11 |
| 2 | 6 |
| 3 | 7 |
| 4 | 15 |

۱) 11,6,10,7,15
۲) 11,6,7,10,15
۳) 10,11,6,7,15
۴) 11,10,6,15,7

**۱۱ـ** یک جدول درهم‌سازی به صورت آدرس‌دهی باز مدیریت می‌شود و درهم‌سازی به صورت یکنواخت صورت می‌گیرد. فرض کنید در حال حاضر **75** درصد جدول پر شده است. حداکثر تعداد **probe** برای جستجوی عنصری که در جدول موجود نیست (جستجوی ناموفق) چند تاست؟

(علوم کامپیوتر ـ سراسری ۸۹)

۱) بستگی به الگوریتم Probing دارد.
۲) بستگی به طول لیست دارد.
۳) 25 بار
۴) 4 بار

**۱۲ـ** اگر برای ذخیره‌سازی کلیدها در باکت‌ها از روش درهم‌سازی **(hashing)** استفاده شود و **n** کلید به طور یکنواخت در **m** باکت توزیع شده باشند و هر کلید با احتمال مشابهی به‌عنوان کلید جست‌وجو درنظر گرفته شود، آنگاه میانگین تعداد مقایسات در یک جست‌وجوی موفق کدام است؟

(مهندسی کامپیوتر ـ آزاد ۸۹)

۱) $\frac{n}{2m}+\frac{1}{2}$　　۲) $\frac{n}{2m}$　　۳) $\frac{n}{m}+1$　　۴) $\frac{n}{m}$

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



## پاسخنامه تست‌های طبقه‌بندی شده کنکوری فصل هفتم

**۱ـ گزینه «۲»** یکی از راه حل‌های تصادم، وارسی خطی است؛ بدین معنی که چنانچه مکان مورد نظر که توسط تابع در هم ساز مشخص شده پر باشد، به اولین مکان خالی بعد از محلی که مشخص شده مراجعه کرده و عنصر را درج می‌کنیم، البته این کار به صورت چرخشی انجام می‌شود. در گزینهٔ ۲، بعد از درج G، عناصر با هیچ ترتیب دلخواهی در H[0...6] به این شکل قرار نمی‌گیرند.

**۲ـ گزینه «۴»** یکی از راه حل‌های تصادم، استفاده از لیست پیوندی است. در این روش اگر مقدار محاسبه شده توسط تابع hash برای عناصر با هم یکسان باشد، این مقادیر به صورت یک لیست پیوندی با هم در ارتباط خواهند بود. در واقع در این روش، آرایه‌ای از لیست‌های پیوندی داریم. با توجه به اینکه تعداد عناصر این آرایه طبق فرض مسئله ۲n است و تعداد $\frac{n}{8}$ عنصر در جدول است و احتمال نگاشت عناصر در درایه‌های جدول مساوی است، بنابراین متوسط تعداد عناصر در هر درایه $\frac{\frac{n}{8}}{2n}=\frac{1}{16}$ خواهد بود.

**۳ـ گزینه «۳»** در آرایه‌های مرتب عملیات درج و حذف بسیار کند انجام می‌شود و در لیست‌های زنجیره‌ای مرتب عملیات حذف نیاز به پیمایش دارد و البته درخت‌های جستجوی دودویی اگر متوازن باشد، سریع‌ترین ساختمان داده‌ها برای عملیات مختلف می‌باشد.

**۴ـ گزینه «۱»** احتمال برخورد یک خانه خاص به تابع درهم‌سازی بستگی دارد.

**۵ـ گزینه «۴»** احتمال تصادم به نوع تابع درهم‌سازی وابسته است.

**۶ـ گزینه «۱»** احتمال بروز تصادم به تابع درهم‌سازی بستگی دارد.

**۷ـ گزینه «۳»** احتمال برخورد در زمان درج هر یک از عناصر را می‌توان به صورت زیر در نظر گرفت:

احتمال برخورد در هنگام درج اولین عنصر برابر صفر می‌باشد (زیرا جدول خالی است.)

احتمال برخورد در هنگام درج دومین عنصر برابر $\frac{1}{m}$ می‌باشد.

احتمال برخورد در هنگام درج سومین عنصر برابر $\frac{2}{m}$ می‌باشد.

احتمال برخورد در هنگام درج چهارمین عنصر برابر $\frac{3}{m}$ می‌باشد.

$\vdots$

احتمال برخورد در هنگام درج n امین عنصر برابر $\frac{n}{m}$ می‌باشد.

بنابراین میانگین تعداد برخوردها در کل عبارتست از: $0+\frac{1}{m}+\frac{2}{m}+\frac{3}{m}+\cdots+\frac{n-1}{m}\in\theta(\frac{n^2}{m})$

**۸ـ گزینه «۱»** در روش زنجیر، اگر دو عنصر با هم برخورد داشته باشند آن‌ها را به صورت لیست پیوندی در کنار هم ذخیره می‌کنیم. حال در بدترین حالت همه n عنصر با هم برخورد دارند که در نتیجه همه در کنار هم ذخیره شده و زمان جستجو برابر $\theta(n)$ خواهد بود. در حالت متوسط n عنصر در m خانه تقسیم می‌شود و زمان جستجو برابر $\theta(\frac{n}{m})$ خواهد بود.

**۹ـ گزینه «۴»** اگر کلید e با مقدار تابع درهم‌ساز 46 نباشد، پس مکان 50 در جدول درهم ساز خالی است و چون g با مقدار تابع در هم ساز 47 زودتر از j,i,h ظاهر شده پس به ترتیب ورود آن در جدول از j,i,h زودتر است و از طرفی به این دلیل که در مکان 47 قرار دارد. اولین مکان خالی بعد از 47، مکان 50 است که در این صورت g در مکان 50 قرار می‌گیرد.

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۱۰ـ گزینه «۴»** در گزینه چهارم از چپ به راست ابتدا 11 با توجه به تابع hash تعریف شده در سؤال در مکان 1 قرار می‌گیرد و سپس 10 در مکان صفر قرار می‌گیرد و 6 باید در مکان 1قرار بگیرد که چون قبلاً توسط 11پر شده است در اولین مکان خالی بعدی، یعنی 2 قرار می‌گیرد و 15 باید در مکان صفر قرار بگیرد که چون توسط 10 پر شده است در اولین مکان خالی بعدی، یعنی مکان 3 باید قرار بگیرد که با توجه به جدول درهم ساز ذکر شده این ترتیب صحیح نمی‌باشد.

---

**۱۱ـ گزینه «۴»** با توجه به این که درهم‌سازی به صورت یکنواخت انجام شده بنابراین، احتمال نگاشت شدن یک کلید به تمام خانه‌های جدول یکسان است و در نتیجه با در نظر گرفتن این که 75 درصد جدول پر شده است می‌توان گفت از هر چهار خانه متوالی 3 خانه پر است در نتیجه، حداکثر چهار مقایسه برای جست‌وجوی ناموفق نیاز داریم.

---

**۱۲ـ گزینه «۱»** اگر $n > m$ باشد و از روش زنجیره‌سازی برای برطرف نمودن مشکل تصادم استفاده شده باشد، آنگاه با توجه به یکنواخت بودن تابع درهم‌سازی در هر باکت به تعداد $\frac{n}{m}$ عنصر وجود دارد و بنابراین در یک جست‌وجوی موفق، میانگین تعداد مقایسه‌ها به‌صورت زیر قابل محاسبه است:

$$\frac{1+2+3+\ldots+\frac{n}{m}}{\frac{n}{m}} = \frac{\frac{\frac{n}{m}(\frac{n}{m}+1)}{2}}{\frac{n}{m}} = \frac{n}{2m}+\frac{1}{2}$$

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



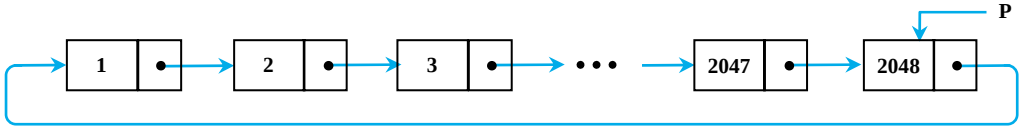
# بخش سوم: روش‌های حل مسأله

## فصل هشتم

## «الگوریتم‌های تقسیم و غلبه (Divide and Conquer)»

### تست‌های طبقه‌بندی شده کنکوری فصل هشتم

۱ـ در لیست پیوندی حلقوی شکل زیر، خروجی قطعه کد داده شده چیست؟ فرض کنید هر عنصر لیست پیوندی دارای دو مؤلفه **CellData** از نوع **Integer** و **Next** از نوع اشاره‌گر باشد. (مهندسی کامپیوتر ـ سراسری۷۶)




```
while P↑.Next < > P do
  begin
    P↑.Next := P↑.Next↑.Next;
    P := P↑.Next
  end;
write(P↑.CellData);
```

۱) 2047
۲) 1
۳) 2048
۴) 1024

۲ـ آرایهٔ مرتب **A** داده شده است. تابع زیر برای جستجوی دودویی در **A** (از اندیس **L** تا **R**) برای پیدا کردن **x** نوشته شده است.

```
Function search (x : real; L,R : integer) : boolean;
    var M : integer;
begin
    M : = (L+R) div 2;
    if (x=A[M]) then return (true);
    if (x <A[M] and (M>L)) then return (search (x, L, M-1));
    if (x > A[M] and (M<R)) then return (search (x, M+1, R));
    return (False)
end;
```

کدام یک از گزینه‌های زیر در مورد الگوریتم فوق برای جستجوی دودویی در **A[1..N]** غلط است؟ (منظور از مقایسه، مقایسهٔ **x** با یک عنصر از **A** است)

(مهندسی کامپیوتر ـ سراسری ۷۷)

۱) اگر x در A[1..N] باشد، همواره با حداکثر $O(\log_2 N)$ پیدا می‌شود.

۲) اگر x در A[1..N] نباشد، پاسخ False همواره با حداکثر $O(\log_2 N)$ مقایسه به دست می‌آید.

۳) در جستجوی ناموفق برای دو مقدار متفاوت x (که در A موجود نیستند) همواره تعداد مقایسه‌ها برابر است.

۴) حداکثر تعداد مقایسه‌ها (چه موفق و چه ناموفق) همواره از $O(N)$ کمتر است.

۳ـ یک آرایه **n** تایی **A** از اعداد صحیح، با شرط‌های زیر داده شده است: (مهندسی کامپیوتر ـ سراسری۷۹)

۱ـ $A(1) = x$ ، $A(n) = y$ و $x < y$ ۲ـ برای هر i ، $|A(i) - A(i+1)| \le 1$

یک الگوریتم کارا برای جستجوی $w\ (x \le w \le y)$ در آرایه **A** طراحی شده است. این الگوریتم اندیس **w** را در صورت وجود به دست می‌آورد. پیچیدگی این الگوریتم در بدترین حالت و حالت متوسط چیست؟

۱) بدترین حالت $O(n)$، حالت متوسط $O(\log n)$

۲) بدترین حالت $O(\log n)$ و حالت متوسط $O(n)$

۳) هر دو حالت $O(n)$

۴) هر دو حالت $O(\log n)$

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



۴ـ دستور حذف شده برنامهٔ جستجوی دودویی زیر کدام است؟ (علوم کامپیوتر ـ سراسری ۸۰)

```
Procedure Binsearch (a: element array; x:element; var left, right. j:integer)
var middle: integer;
begin
    if (left <= right) then begin
        middle := (left + right) div 2;
        case compare (x, a [middle]) of
            '>': Binsearch (a, x, middle +1 , right, j);
            '<': دستور حذف شده
            '=':  j := middle;
        end;
    end;
     j := −1;
end;
```

۱) BinSearch (a, x, middle −1, right, j);　　۲) BinSearch (a, x, middle −1, left, j);

۳) BinSearch (a, x, left, middle −1, j);　　۴) BinSearch (a, x, left, middle, right);

۵ـ یک ماشین انتزاعی که در مبنای **10** کار می‌کند، اعداد را به راحتی با یکدیگر جمع و تفریق می‌کند و ضرب در اعداد $10^x$ را از طریق **Shift** به راحتی عمل می‌کند، اما در ضرب تنها قادر است اعداد یک رقمی را در هم ضرب نماید. به نظر شما برای ضرب دو عدد $5618 * 1234$ حداقل تعداد عملیات ممکن ضرب چه تعداد می‌باشد تا حاصل ضرب فوق به دست آید؟ (راهنمائی: از ضرب اعداد صحیح بزرگ استفاده کنید.)

(مهندسی کامپیوتر ـ سراسری ۸۱)

۱) 8　　۲) 12　　۳) 13　　۴) 16

۶ـ برای محاسبهٔ عبارت روبرو حداقل چند عمل ضرب می‌توان استفاده کرد؟ $a_n x^n + a_{n-1}x^{n-1} + a_{n-2}x^{n-2} + \ldots + a_1 x + a_0$

(مهندسی کامپیوتر ـ سراسری ۸۱)

۱) $n$　　۲) $n^2$　　۳) $n + \frac{n(n+1)}{2}$　　۴) $n + \frac{n(n-1)}{2}$

۷ـ اگر الگوریتم جست‌وجوی دودویی را برای جست‌وجوی عناصر آرایه $A[1..8] = [5,10,15,20,25,30,35,40]$ به کار ببریم، میانگین تعداد مقایسه‌ها برای جست‌وجوی موفق، تقریباً کدام است؟ (مهندسی کامپیوتر ـ سراسری ۸۲)

۱) 2.2　　۲) 2.4　　۳) 2.6　　۴) 2.8

۸ـ بهترین الگوریتم ضرب دو عدد $n$ بیتی دارای کدام یک از پیچیدگی‌ها می‌باشد؟ (علوم کامپیوتر ـ سراسری ۸۲)

۱) $O(n)$　　۲) $O(n^2)$　　۳) $O(n \log n)$　　۴) $O(n^{\log_2^3})$

۹ـ برای ضرب دو عدد $A$ و $B$ به ترتیب با $n$ و $4n$ رقم برای $5000 \le n \le 10000$: (مهندسی کامپیوتر ـ سراسری ۸۳)

۱) بهتر است از روش معمولی و کلاسیک برای ضرب آنها استفاده کرد.

۲) بهتر است از الگوریتم تقسیم و حل استفاده کرد و فرقی نمی‌کند به چه ترتیب.

۳) بهتر است پس از قسمت‌کردن $B$ به 4 بخش $n$ رقمی هر کدام از این بخش‌ها را در $A$ و با روش تقسیم و حل ضرب کند.

۴) بهتر است پس از اضافه‌کردن $3n$ صفر به سمت چپ $A$، عدد حاصل را با کمک الگوریتم تقسیم و حل در $B$ ضرب کند.

۱۰ـ اگر $A$ یک ماتریس $5 \times 13$، $B$ ماتریس $89 \times 5$، $C$ ماتریس $3 \times 89$ و $D$ یک ماتریس $34 \times 3$ باشد کمترین تعداد ضرب برای بدست آوردن حاصل ضرب $A \times B \times C \times D$ چقدر است؟ (مهندسی IT ـ سراسری ۸۳)

۱) 4055　　۲) 2856　　۳) 2820　　۴) 1742

۱۱ـ تعداد **9** سکه کاملاً هم شکل داریم که فقط یکی از آن‌ها تقلبی است و ممکن است که سبک‌تر یا سنگین‌تر از بقیه باشد. با یک ترازوی شاهین‌دار (غیر عقربه‌ای) حداقل چند بار توزین لازم است تا سبکی یا سنگینی سکه تقلبی را تعیین کنیم؟ (علوم کامپیوتر ـ سراسری ۸۳)

۱) سه بار　　۲) چهار بار　　۳) هشت بار　　۴) ده بار

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



۱۲ـ تابع زیر به منظور جستجوی دودویی برای x در یک آرایه صعودی مرتب شده از اعداد صحیح نوشته شده است. به جای آرایه از اشاره‌گر به ابتدا و انتهای آرایه استفاده شده است. چه عبارتی باید به جای (A) نوشته شود؟ (علوم کامپیوتر ـ سراسری ۸۳)

```
int *bsearch (int *p,int *q,int x){
if (p == q)
return     (*p) == x  ? p : NULL;
int   *r = (A);
if ((*r) < x)
return bsearch (r + l,q,x);
else
return bsearch (p,r,x)
}
```

۱) $(p+q)/2$

۲) $(p+q+1)/2$

۳) $p+(q-p)/2$

۴) $p+(q-p)/2+1$

۱۳ـ چهار ماتریس زیر با ابعاد داده شده را در نظر بگیرید: $A_{10\times2}, B_{2\times25}, C_{25\times3}, D_{3\times4}$ کدام‌یک از گزینه‌های زیر مربوط به پرانتزبندی بهینه برای محاسبه حاصل‌ضرب آنهاست؟ (مهندسی IT ـ سراسری ۸۴)

۱) $(A\times B)\times(C\times D)$ ۲) $\big(A\times(B\times C)\big)\times D$ ۳) $A\times((B\times C)\times D)$ ۴) $((A\times B)\times C)\times D$

۱۴ـ فرض می‌کنیم n مجموعه تک عنصری متمایز از هم داریم. اگر u عمل union انجام دهیم حداقل تعداد مجموعه تک عنصر که باقی می‌ماند برابر است با: (علوم کامپیوتر ـ سراسری ۸۴)

۱) $n-2u$ ۲) $\max\{n-2u,0\}$ ۳) $\max\{2u,n\}$ ۴) $\min\{u,\frac{n}{2}\}$

۱۵ـ اگر در ضرب ماتریس‌ها به روش استراسن (Strassen)، مسئله کوچک ضرب ماتریس‌های 2*2 باشد، برای ضرب دو ماتریس 8*8 چند ضرب عددی صورت می‌پذیرد؟ (مهندسی IT ـ سراسری ۸۵)

۱) 57 ۲) 343 ۳) 392 ۴) 512

۱۶ـ یک کامپیوتر موازی با $\frac{n}{2}$ پردازنده مفروض است. اگر بخواهیم با استفاده از این کامپیوتر اعداد موجود در یک آرایه n عنصری را جمع نماییم، زمان مورد نیاز چیست؟ (مهندسی کامپیوتر ـ آزاد ۸۵)

۱) $O(\log_{10}^{n})$ ۲) $O(\log_{2}^{n})$ ۳) $O(n\log_{10}^{n})$ ۴) $O(\frac{n}{2})$

۱۷ـ یک کامپیوتر موازی دارای $\frac{n}{2}$ پردازنده می‌باشد. اگر بخواهیم اعداد موجود در یک آرایه n عنصری را ضرب نماییم، زمان موردنیاز چه خواهد بود؟ (مهندسی کامپیوتر ـ آزاد ۸۵)

۱) $O(n\log 2)$ ۲) $O(\frac{n}{2})$ ۳) $O(\log_{2}^{n})$ ۴) $O(n)-2$

۱۸ـ الگوریتم زیر را جهت محاسبه ضریب دو جمله‌ای که در آن K و n مثبت و $K \le n$ است درنظر بگیرید.

```
int Comb (int n, int k)
  {
    if ((k == 0) or (n == k))
      return 1;
    else
      return comb (n - 1, k - 1) + comb (n - 1, k)
  }
```

در این‌صورت تعداد جملاتی که الگوریتم فوق برای تعیین $\binom{n}{k}$ محاسبه می‌کند چیست؟ (مهندسی کامپیوتر ـ آزاد ۸۵)

۱) $2\binom{n}{k}-1$ ۲) $2\binom{n}{k-1}+1$ ۳) $2\binom{n-1}{k-1}+1$ ۴) $2\binom{n-1}{k}+3$

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



۱۹ـ در باشگاهی قرار است مسابقات والیبال برگزار شود به این ترتیب که بین $n = 2^k$ تیم شرکت کننده، اول $\frac{n}{2}$ مسابقه صورت می‌گیرد. سپس بین تیم‌های برنده و نیز بین تیم‌های بازنده دوباره به همین شکل مسابقاتی برگزار می‌شود (در هر مرحله هر تیم برنده با یک تیم برندهٔ دیگر از همان مرحله مسابقه می‌دهد و همینطور برای تیم‌های بازنده آن مرحله ) و در هر مرحله، مسابقات به همین منوال ادامه پیدا می‌کند تا زمانی که به گروه‌های یک تیمی برسیم. تعداد کل بازی‌ها برای $n = 128$ برابر است با : ( نتیجهٔ مساوی نداریم) (مهندسی کامپیوتر ـ سراسری ۸۶)

۱) 320 ۲) 448 ۳) 576 ۴) 704

۲۰ـ فرض کنید $L_n$ مجموعه‌ی نقاط $(i, j)$ با مختصات صحیح باشد به طوری که: $i \geq 0, j \geq 0, i + j \leq n$ مجموعه‌ی نقاط «همسایه‌ی» $N_n(i, j)$ نقطه $i, j$ است. می‌گوییم $(k, m)$ و $(i, j)$ همسایه هستند اگر مختصات این دو نقطه حداکثر یک واحد اختلاف داشته باشند. به عبارت دقیق‌تر:

$$N_n(i, j) = \{(k, m) \in L_n : |i - k| \leq 1 \text{ or } |j - m| \leq 1\}$$

برای $n > 3$ و برای نقاط $(i, j) \in L_n$، مقادیر مختلفی که مجموعه‌ی $N_n(i, j)$ می‌تواند داشته باشد چند تاست ؟ (مهندسی کامپیوتر ـ سراسری ۸۶)

۱) 4 ۲) 5 ۳) 6 ۴) 7

۲۱ـ اگر در ضرب ماتریس‌ها به روش استراسن (Strassen) مسئله کوچک ضرب ماتریس‌های $2 * 2$ باشد، با چند فراخوانی بازگشتی به الگوریتم استراسن عمل ضرب دو ماتریس $8 * 8$ انجام می‌پذیرد؟ (مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۶)

۱) 343 ۲) 57 ۳) 49 ۴) 7

۲۲ـ برای حل یک مسئله به اندازه $n$ با الگوریتم تقسیم و غلبه سه روش به شرح زیر امکان‌پذیر است:

روش اول: حل 3 زیر مسئله به اندازه $\frac{n}{2}$ و ترکیب آنها با هزینه $\theta(n^2\sqrt{n})$

روش دوم: حل 4 زیر مسئله به اندازه $\frac{n}{2}$ و ترکیب آنها با هزینه $\theta(n^2)$

روش سوم: حل 5 زیر مسئله به اندازه $\frac{n}{2}$ و ترکیب آنها با هزینه $\theta(n \log n)$

کدام روش دارای هزینه کمتری است؟ (مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۶)

۱) روش اول ۲) روش دوم ۳) روش سوم ۴) هر سه روش یکسانند

۲۳ـ $T(n) = 2T(\frac{n}{2}) + n - 1$ این فرمول بازگشتی دقیقاً زمان Merge sort در کدام حالت می‌باشد؟ (علوم کامپیوتر ـ سراسری ۸۶)

۱) بدترین حالت ۲) بهترین حالت ۳) در همه حالت ۴) بهترین و میانگین

۲۴ـ الگوریتم ضرب ماتریس استراسن (Strassen) به این صورت عمل می‌کند که ابتدا دو ماتریس $n \times n$ مفروض $A$ و $B$ را به عنوان ورودی دریافت نموده و ماتریس خروجی، که حاصلضرب $A$ و $B$ است را تولید می‌کند. تقسیم‌بندی ماتریس‌ها در این روش به صورت زیر است:

$$\begin{matrix} n/2 \updownarrow \\ \\ \end{matrix} \overset{\leftarrow n/2 \rightarrow}{\begin{bmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{bmatrix}} = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} \times \begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix}$$

تقسیمات متوالی تا رسیدن به دو ماتریس $1 \times 1$ ادامه می‌یابد. در این صورت پیچیدگی زمانی حالت معمول تعداد جمع‌ها/ تفریق‌ها در الگوریتم استراسن برابر است با: (فرض می‌کنیم $n > 1$ و $n$ توانی از 2 می‌باشد). (مهندسی کامپیوتر ـ آزاد ۸۶)

۱) $\begin{cases} T(n) = 7(T(\frac{n}{2}) + (\frac{n}{2})^2) \\ T(1) = 1 \end{cases}$

۲) $\begin{cases} T(n) = n(T(\frac{7n}{2}) + \frac{n^2}{2}) \\ T(1) = 0 \end{cases}$

۳) $\begin{cases} T(n) = 7T(\frac{n}{2}) + 18(\frac{n}{2})^2 \\ T(1) = 0 \end{cases}$

۴) $\begin{cases} T(n) = 7(T(\frac{n}{4}) + 9n^2 \\ T(1) = 1 \end{cases}$

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



۲۵ـ الگوریتم بازگشتی برای مشخص نمودن عنصر nام دنباله فیبوناچی به‌صورت زیر تعریف شده است:
اگر $T(n)$ پیچیدگی زمانی الگوریتم فیبوناچی فرض شود، در این‌صورت کدام‌یک از روابط زیر درست می‌باشد؟ (مهندسی کامپیوتر ـ آزاد ۸۶)

```
int fib (int n)
  {
   if (n <= 1)
       return n;
   else
       return fib (n – 1) + fib (n – 2);
  }
```

۱) $T(n) > 2^{\frac{n}{2}}$
۲) $T(n) > n^2$
۳) $T(n) = 2T(n-2)+1$
۴) $T(n) = 2T(n-1)+1$

۲۶ـ یک کامپیوتر موازی با تعداد K پردازنده مفروض است. می‌خواهیم اعداد موجود در آرایه A[1..2K] را با هم جمع کنیم. در این صورت مدت زمان لازم برابر خواهد بود با: (مهندسی کامپیوتر ـ آزاد ۸۶)

۱) $1+[\log K]$ ۲) $[\log \frac{K}{2}]$ ۳) $O(K)$ ۴) $O(2K)$

۲۷ـ آرایه‌ای با تعداد عناصر n مفروض است. می‌خواهیم عنصر $\chi$ را در آرایه جست‌وجو نماییم. اگر عنصر $\chi$ با احتمال $\frac{1}{2}$ در آرایه موجود باشد، در این‌صورت به‌طور میانگین چه تعداد از عناصر آرایه بایستی مورد جست‌وجو قرار گیرد؟ (مهندسی کامپیوتر ـ آزاد ۸۶)

۱) $\frac{1}{2}$ ۲) $\frac{3}{4}$ ۳) $\frac{2}{3}$ ۴) $\frac{3}{5}$

۲۸ـ اگر S(n) و U(n) به ترتیب متوسط مقایسه برای جستجوی موفق و ناموفق با روش جستجوی دودویی در یک آرایه مرتب n عضوی باشند کدام یک از گزینه‌های زیر درست است؟ (مهندسی IT ـ سراسری ۸۷)

۱) $S(n) = U(n) - 1$ ۲) $S(n) = (1+\frac{1}{n})U(n) - 1$ ۳) $S(n) = (1+\frac{1}{n})U(n) + 1$ ۴) $U(n) = S(n) + 2$

۲۹ـ آرایه مرتب A با n عدد صحیح مثبت و منفی ولی مجزا از هم داده شده است. می‌خواهیم در صورت وجود اندیس i را به دست آوریم که $A[i] = i$ باشد. و در صورت نبود جواب آن را اعلام کنیم. (مهندسی کامپیوتر ـ سراسری ۸۷)

۱) می‌توان نشان داد که هر راه حل این مسئله از $\Omega(n)$ است.
۲) برای این مسئله الگوریتمی از مرتبه‌ی $O(\sqrt{n})$ وجود دارد.
۳) برای این مسئله الگوریتمی از مرتبه‌ی $O(\lg n)$ وجود دارد.
۴) با استفاده از درخت تصمیم می‌توان نشان داد که هر درجه‌ی پیچیدگی هر راه‌حل این مسئله اکیداً بیش‌تر از $\lg n$ است.

۳۰ـ دو آرایه‌ی مرتب هر یک به طول n از اعداد مجزا از هم داده شده است. می‌خواهیم با کم‌ترین تعداد خواندن درایه‌های این دو آرایه میانه‌ی 2n عدد موجود در این دو آرایه را به دست آوریم. کدام‌یک از رابطه‌های بازگشتی نحوه‌ی رفتار این الگوریتم را نشان می‌دهد؟ فرض کنید T(2n) بیش‌ترین تعداد دسترسی به درایه‌های این دو آرایه برای حل بهینه‌ی این مسئله است. (مهندسی کامپیوتر ـ سراسری ۸۷)

۱) $T(2n) = T(n) + 2$ ۲) $T(2n) = 2T(n) + 2$ ۳) $T(2n) = 3T(\frac{n}{2}) + 2$ ۴) $T(2n) = T(\frac{3n}{2}) + 2$

۳۱ـ ماتریس $M_{\sqrt{n}\times\sqrt{n}}$ را با n عدد در نظر بگیرید به طوری که اعداد در سطر و ستون ماتریس صعودی مرتب شده باشند. الگوریتم زیر برای یافتن عدد k در ماتریس M داده شده است. پیچیدگی زمان این الگوریتم چند است؟ اگر الگوریتم را تغییر دهیم و در هر ستون از binary search استفاده شود آیا زمان بهتر می‌شود یا بدتر؟ (علوم کامپیوتر ـ سراسری ۸۷)

```
Search (K)
begin
      r = √n;
      for c = 1 to √n do
      begin
       while (M[r,c] > k & r > 1)
       r = r – 1;
       if  M[r,c] = k then return  true;
      end;
return false;
end;
```

۱) $O(n)$ و زمان بدتر می‌شود. ۲) $O(\sqrt{n})$ و زمان بدتر می‌شود. ۳) $O(\sqrt{n})$ و زمان بهتر می‌شود. ۴) $O(n)$ و زمان بهتر می‌شود.

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



۳۲ـ اگر در الگوریتم ضرب ماتریس‌ها، $T(n)$ برابر تعداد دستورالعمل ضرب در داخلی‌ترین حلقه **for** باشد، در این صورت $T(n)$ کدام است؟
(مهندسی کامپیوتر ـ آزاد ۸۷)

۱) $T(n)=n^2+2n$ ۲) $T(n)=n^3$ ۳) $T(n)=(n+1)^3$ ۴) $T(n)=3n^2$

۳۳ـ یک کامپیوتر موازی دارای $k$ عدد پردازنده می‌باشد. این کامپیوتر در چه زمانی قادر خواهد بود تعداد $k$ عدد صحیح را با هم جمع نماید؟ ($k$ توانی از $2$ می‌باشد).
(مهندسی کامپیوتر ـ آزاد ۸۷)

۱) $\lfloor \log_2^k \rfloor$ ۲) $[\log(\frac{k}{2})]$ ۳) $\lceil \log(k-2) \rceil$ ۴) $\log_2^k - 1$

۳۴ـ پیچیدگی زمانی حالت معمول تعداد ضرب‌های الگوریتم استراسن به صورت زیر تعریف شده است:

$T(n)=7T(\frac{n}{2})$

($n>1$ و $n$ توانی از دو و $T(1)=1$ می‌باشد)
در این‌صورت $T(n)$ معادل کدام گزینه است؟
(مهندسی کامپیوتر ـ آزاد ۸۷)

۱) $T(n)=2\log_2^{7n}$ ۲) $T(n)=n^{\log_2^7}$ ۳) $T(n)=7\log_2^n$ ۴) $T(n)=7^{\log_2^n}$

۳۵ـ الگوریتم فیبوناچی را به‌صورت زیر درنظر بگیرید. اگر تعداد عناصر درخت بازگشتی الگوریتم فوق را $T(n)$ بنامیم و $n \geq 2$ باشد، در این صورت کدام گزینه صحیح است؟
(مهندسی کامپیوتر ـ آزاد ۸۷)

```
int fib (int n)
  {
   if (n <= 1)
      return n;
   else
      return (fib (n – 1) + fib (n – 2));
```

۱) $T(n) > 2^{\frac{n}{2}}$ ۲) $T(n) > 2n^2 + 1$ ۳) $T(n) > 2^n - 2$ ۴) $T(n) > 2^{\frac{(n+1)}{2}}$

۳۶ـ فرض کنیم آرایه $B$ شامل $128$ عنصر و به‌صورت غیر نزولی مرتب شده باشد. در این‌صورت برای یافتن عدد $\chi$ (به‌طوری‌که $\chi$ از کلیه عناصر آرایه $B$ بزرگ‌تر است) با استفاده از الگوریتم جست‌وجوی دودویی چه تعداد مقایسه موردنیاز است؟
(مهندسی کامپیوتر ـ آزاد ۸۷)

۱) $7$ ۲) $8$ ۳) $6$ ۴) $9$

۳۷ـ یک کامپیوتر موازی دارای $n$ عدد پردازنده می‌باشد. این کامپیوتر در چه زمانی می‌تواند تعداد $2n$ عدد صحیح را باهم جمع نماید؟ ($n$ توانی از $2$ می‌باشد).
(مهندسی کامپیوتر ـ آزاد ۸۷)

۱) $[\log_2(2n)]$ ۲) $n\log_2(n)$ ۳) $\log_2^n + 1$ ۴) $[\log_2(\frac{n}{2})]$

۳۸ـ مرتبه اجرایی الگوریتم جمع ماتریس‌ها و ضرب ماتریس‌ها به‌ترتیب از راست به چپ کدام است؟
(مهندسی کامپیوتر ـ آزاد ۸۸)

۱) $O(n^3), O(n^2)$ ۲) $O(n^2), O(n\log n)$ ۳) $O(n^2\log n), O(n^2)$ ۴) $O(n^2\log n), O(n\log n)$

۳۹ـ در یک کامپیوتر موازی شامل چندین پردازنده در صورتی که بخواهیم اعداد از $1$ تا $n$ با زمان اجرای $O(\log_2^n)$ با هم ضرب شوند در این‌صورت تعداد پردازنده‌های موردنیاز کدام است؟
(مهندسی کامپیوتر ـ آزاد ۸۸)

۱) $\frac{n}{2}$ ۲) $2n$ ۳) $2n-1$ ۴) $\frac{n+1}{2}$

۴۰ـ ماتریس‌های $A_0, A_1, A_2, \ldots$ را، با تعریف زیر در نظر بگیرید:

$A_0$، یک ماتریس به شکل $1\times1$ $\quad A_0:[1]$ $\quad$ و $\quad A_K$، یک ماتریس به شکل $2^k \times 2^k$ $\quad A_k : k>0$

$$A_k=\begin{bmatrix} A_{k-1} & A_{k-1} \\ A_{k-1} & -A_{k-1} \end{bmatrix}$$

فرض کنید $V$، یک بردار ستونی به طول $n=2^k$ است. برای ضرب $V$ . $A_k$ زمان مصرفی کدام مورد است؟
(مهندسی کامپیوتر ـ سراسری ۸۹)

۱) $\theta(n\lg n)$ ۲) $\theta(n)$ ۳) $\theta(k\lg n)$ ۴) $\theta(n^2\lg n)$

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۴۱ـ** مسئله‌ی کلاسیک ضرب دو چند جمله‌ای $A=\sum_{i=0}^{n}a_ix^i$ در $B=\sum_{i=0}^{n}b_ix^i$ و محاسبه‌ی $C=A\times B=\sum_{k=0}^{2n}c_kx^k$ را در نظر بگیرید که با داشتن $n+1$ ضریب $a_i$ و $n+1$ ضریب $b_j$، هدف محاسبه‌ی $2n+1$ ضریب $c_k$ است. راه‌حل مبتنی بر «تقسیم و حل» هر دو مسئله را به دو قسمت تقسیم می‌کند و مسئله‌ی اصلی را حل می‌کند. در صورتی که $B'=\sum_{j=\frac{n}{2}}^{n}b_jx^j$ باشد (یعنی نیمی از ضرایب آن صفر باشند)، کدام یک از گزینه‌های زیر زمان اجرای الگوریتم تقسیم وحل برای به دست آوردن ضرایب چندجمله‌ای $C=A\times B'$ خواهد بود؟ (مهندسی کامپیوتر ـ سراسری ۸۹)

۱) رابطه‌ی بازگشتی راه‌حل تغییر می‌کند، اما جواب آن همان $O(n\log_2^3)$ است.

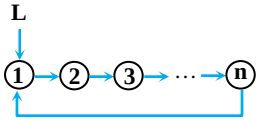
۲) رابطه‌ی بازگشتی راه‌حل تغییر نمی‌کند، و جواب هم همان $O(n\log_2^3)$ است.

۳) رابطه‌ی بازگشتی به $T(n)=2T(\frac{n}{2})+\Theta(n)$ تغییر می‌کند، ولی جواب آن $O(n\lg n)$ است.

۴) رابطه‌ی بازگشتی به $T(n)=2T(\frac{n}{2})+\Theta(n)$ تغییر می‌کند که جواب آن $O(n\lg n)$ است.

**۴۲ـ** با توجه به تابع زیر و لیست حلقوی مذکور به ازای مقادیر $n$ برابر 729 و 2200 مقدار خروجی به ترتیب برابر چند خواهد بود؟ (مهندسی کامپیوتر ـ سراسری ۸۹)

```
Int  SO(LIST*L){
While(L->next !=L){
L → next=L ->next->next
L=L → next:
}
return  L → data:
}
```




۱) 1 و 40 ۲) 1 و 1 ۳) 729 و 2200 ۴) هیچ‌کدام

**۴۳ـ** یک ماتریس $A$ به اندازه‌ی $n\times n$ را در نظر بگیرید که همه‌ی عناصر سطرهای آن از چپ به راست و همه‌ی ستون‌های آن از بالا به پایین به صورت غیرنزولی مرتب هستند. می‌خواهیم با مقایسه‌ی عناصر این ماتریس با $x$، در صورت وجود مکان $x$ را در ماتریس بیابیم. با حداکثر چند تا مقایسه می‌توان این کار را انجام داد؟ (مهندسی کامپیوتر ـ سراسری ۸۹)

۱) $2n$ ۲) $n^2$ ۳) $3n$ ۴) $n\log n$

**۴۴ـ** بهترین الگوریتم برای جستجوی مقداری مثل $z$ در یک ماتریسی به ابعاد $n\times n$ که در راستای سطرها و ستون‌ها به صورت غیرنزولی مرتب شده است، دارای چه هزینهٔ زمانی می‌باشد؟ (علوم کامپیوتر ـ سراسری ۸۹)

۱) $O(n^2)$ ۲) $O(n)$ ۳) $O(n^{\log_2^3})$ ۴) $O(n\log\frac{n}{2})$

**۴۵ـ** برای محاسبهٔ حاصل چندجمله‌ای زیر به روش تقسیم و حل چه مرتبهٔ زمانی لازم است؟ (علوم کامپیوتر ـ سراسری ۸۹)

$$P_{n-1}(x)=a_{n-1}x^{n-1}+a_{n-2}x^{n-2}+\cdots+a_1x+a_0$$

۱) $O(n\log\frac{n}{2})$ ۲) $O(\log\frac{n}{2})$ ۳) $O(n)$ ۴) $O(n^2)$

**۴۶ـ** $n$ نفر در یک کلاس ساختمان داده حضور دارند، می‌دانیم که تنها یکی از آن‌ها در درس ساختمان 20 گرفته است؛ می‌خواهیم این نابغه! را پیدا کنیم. بدین منظور تنها می‌توانیم این گونه عمل کنیم: «هر بار به دلخواه خود می‌بایست $k$ ($k$ عدد دلخواهی از 1 تا $n$ است) نفر از میان دانشجویان انتخاب نموده، در یک گروه قرار داده و از آن گروه سؤال کنیم که آیا نابغه در میان شما هست یا نه؟» و گروه در پاسخ تنها یک جواب بله و یا خیر می‌دهد. در بدترین حالت ممکن به چند پرسش نیاز داریم تا بتوانیم حتماً نابغه را پیدا کنیم؟ (یعنی در واقع پیچیدگی بهترین راه‌حل برحسب $n$ از چه درجه‌ای هست؟) (علوم کامپیوتر ـ سراسری ۸۹)

۱) $\sqrt{n}$ ۲) $n$ ۳) $\log n$ ۴) $n\log n$

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۴۷ـ** بهترین الگوریتم برای یافتن $n$ امین عنصر دنبالهٔ فیبوناچی به کمک ماتریس تبدیل $\begin{bmatrix} 0 & 1 \\ 1 & 1 \end{bmatrix}$ دارای چه هزینهٔ زمانی می‌باشد؟

(علوم کامپیوتر ـ سراسری ۸۹)

۱) $O(\varphi^n)\ ,\ \varphi = \frac{1+\sqrt{5}}{2}$ ۲) $O(\log_2 n)$ ۳) $O(n^2)$ ۴) $O(n)$

**۴۸ـ** کم رشدترین حد بالای زمان اجرای الگوریتم استراسن **(strassen)** برای ضرب دو ماتریس مربعی $n \times n$ کدام است؟

(مهندسی IT ـ سراسری ۹۰)

۱) $O(n^2)$ ۲) $O(n^3)$ ۳) $O(n^{\log_2^6})$ ۴) $O(n^{\log_2^7})$

**۴۹ـ** دو آرایه‌ی مرتب $A_1$ و $A_2$ با مجموع تعداد $n$ عنصر داده شده‌اند. فرض کنید که عناصر مجزا هستند. می‌خواهیم $k$ امین عنصر $A_1 \cup A_2$ را به دست آوریم. این کار را می‌توان در کدام زمان زیر انجام داد؟ (مهندسی کامپیوتر ـ سراسری ۹۰)

۱) $O(\lg n)$ ۲) $O(\lg^2 n)$ ۳) $O(\lg \lg n)$ ۴) $O(n + \lg n)$

**۵۰ـ** ماتریس $A_{n \times n}$ از اعداد را در نظر بگیرید که در آن همه سطرهای ماتریس (از چپ به راست) و همه ستون‌های ماتریس (از بالا به پایین) همگی صعودی هستند. برای پیداکردن عنصر $x$ در این ماتریس: (مهندسی کامپیوتر ـ آزاد ۹۰)

۱) با مقایسه $x$ با راست‌ترین عنصر سطر اول (بالاترین عنصر ستون آخر) می‌توان در هر مقایسه یک سطر یا ستون را از ماتریس کنار گذاشته و با بقیه ماتریس عمل جستوجو را ادامه داد.

۲) با مقایسه $x$ با چپ‌ترین عنصر سطر اول (بالاترین عنصر ستون اول) می‌توان در هر مقایسه یک سطر یا ستون را از ماتریس کنار گذاشته و با بقیه ماتریس عمل جستوجو را ادامه داد.

۳) با مقایسه $x$ با راست‌ترین عنصر سطر آخر (پایین‌ترین عنصر ستون آخر) می‌توان در هر مقایسه یک سطر یا ستون را از ماتریس کنار گذاشته و با بقیه ماتریس عمل جستوجو را ادامه داد.

۴) هیچ راهی وجود ندارد تا با یک مقایسه دسته‌ای از عناصر از محدوده قابل جستوجو خارج شوند.

**۵۱ـ** ماتریس $A_{n \times n}$ از اعداد را درنظر بگیرید که در آن همه سطرهای ماتریس (از چپ به راست) و همه ستون‌های ماتریس (از بالا به پایین) همگی صعودی هستند. برای پیداکردن عنصر $x$ در این ماتریس بهترین الگوریتم از کدام مرتبه زمانی است؟ (مهندسی کامپیوتر ـ آزاد ۹۰)

۱) $O(n \log n)$ ۲) $O(\log n)$ ۳) $O(n^2)$ ۴) $O(n)$

**۵۲ـ** مجموعه‌ای از $n$ نقطه در فضای دوبعدی مفروض است. بهترین الگوریتم برای پیداکردن نزدیک‌ترین زوج نقاط در فضای دوبعدی دارای کدام مرتبه زمانی است؟ (مهندسی کامپیوتر ـ آزاد ۹۰)

۱) $O(n^2)$ ۲) $O(n \log n)$ ۳) $O(n)$ ۴) $O(n^{\frac{3}{2}})$

**۵۳ـ** اگر $a$ یک عدد حقیقی و $b$ یک عدد صحیح مثبت $n$ بیتی باشد، بهترین الگوریتم موجود برای محاسبه $a^b$ دارای کدام مرتبه زمانی است؟ (فرض کنید عمل ضرب در یک واحد زمانی اجرا می‌شود) (مهندسی کامپیوتر ـ آزاد ۹۰)

۱) $O(\log n)$ ۲) $O(b)$ ۳) $O(a)$ ۴) $O(n)$

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



## «الگوریتم‌های تقسیم و غلبه (Divide and Conquer)»

### تست‌های طبقه‌بندی شده کنکوری فصل هشتم

**۱ـ گزینه «۳»** با توجه به دستور $p \uparrow .next := p \uparrow .next \uparrow .next$، همواره گرهٔ بعد از گرهٔ جاری حذف می‌شود و دستور $p := p \uparrow .next$ سبب می‌شود که p به گرهٔ بعدی اشاره کند. این عمل تا زمانی ادامه می‌یابد که $p \uparrow .next = p$ شود یا به عبارت دیگر تنها یک گره باقی بماند. فرض کنید لیست حاوی اعداد 1 تا $2n$ است و اشاره‌گر به عدد $2n$ اشاره می‌کند. پس از یک دور پیمایش لیست، اعداد فرد حذف می‌شوند و لیستی حاوی اعداد $2n,...,4,2$ خواهیم داشت که باز هم اشاره‌گر به $2n$ اشاره می‌کند. در واقع آخرین عنصر باقی‌مانده از لیست $2n$ عنصری با اعداد 1 تا $2n$ همان آخرین عنصر باقی‌مانده از لیست n عنصری با اعداد $2n,...,6,4,2$ است (یعنی لیستی که اعداد آن در 2 ضرب شده‌اند). از رابطه بازگشتی زیر می‌توان برای بررسی آخرین عنصر باقی‌مانده استفاده نمود:

$$W(2n) - 2W(n) \Rightarrow W(2^n) = 2W(2^{n-1}) = 2^n W(1)$$
$$W(1) = 1$$

با قرار دادن $2n = 2048$ در رابطه می‌توان به مقدار 2048 به عنوان عضو باقی‌مانده رسید.

---

**۲ـ گزینه «۳»** با توجه به الگوریتم جستجوی دودویی و مطالب ذکر شده در این فصل می‌دانیم مرتبهٔ اجرایی این الگوریتم در بدترین حالت $O(Log_2 n)$ می‌باشد، بدترین حالت یعنی x در آرایه نباشد یا در صورت وجود حداکثر $\left\lfloor Log_2^n \right\rfloor + 1$ مقایسه لازم داشته باشد. آرایهٔ 5 عنصری $[1,3,5,9,11]$ را در نظر بگیرید، برای جستجوی ناموفق عدد 4 در این آرایه، 3 مقایسه نیاز است ولی برای جستجوی ناموفق عدد 1- در این آرایه، دو مقایسه لازم می‌باشد. بنابراین در جستجوی ناموفق برای دو مقدار متفاوت x همواره تعداد مقایسه‌ها برابر نیست.

---

**۳ـ گزینه «۴»** با توجه به اینکه آرایهٔ A، آرایه‌ای از اعداد صحیح بوده و $|A(i) - A(i+1)| \le 1$ و $x \le w \le y$ الگوریتم کارا برای جستجوی w در آرایه، الگوریتم جستجوی دودویی می‌باشد، در نتیجه این الگوریتم در بدترین حالت و حالت متوسط دارای پیچیدگی زمانی $O(Logn)$ است.

---

**۴ـ گزینه «۳»** در اینجا جستجو یا به سمت چپ یا راست انشعاب داده می‌شود و یا با دستور '=' خاتمه پیدا می‌کند. با توجه به این‌که انشعاب به راست با دستور زیر معین شده لذا باید دستور انشعاب به چپ یعنی از left تا $middle - 1$ بررسی شود.

```
'>'; Binsearch(a, x, middle + 1, right, j);
Binsearch(a, x, left, middle - 1, j);
```

---

**۵ـ گزینه «۳»** برای ضرب دو عدد داده شده به صورت زیر عمل می‌کنیم:

$$1234 \times 5618 = (12 \times 10^2 + 34)(56 \times 10^2 + 18) = (12 \times 56) \times 10^4 + (46 \times 74 - 12 \times 56 - 34 \times 18) \times 10^2 + (34 \times 18)$$

بنابراین ضرب دو عدد چهار رقمی فعلاً به ضرب اعداد دو رقمی منجر شده است که باید آنها را به صورت بازگشتی ضرب کرد تا به اعداد یک رقمی برسیم.

$$(46 \times 74) = (4 \times 10 + 6)(7 \times 10 + 4) = (4 \times 7) \times 10^2 + (10 \times 11 - \boxed{4 \times 7} - \boxed{6 \times 4}) \times 10^1 + 6 \times 4$$

$$(12 \times 56) = (1 \times 10^1 + 2)(5 \times 10^1 + 6) = (1 \times 5) \times 10^2 + (3 \times 11 - \boxed{1 \times 5} - \boxed{2 \times 6} \times 10^1 + 2 \times 6$$

$$(34 \times 18) = (3 \times 10^1 + 4)(1 \times 10^1 + 8) = 1 \times 3 \times 10^2 + (\boxed{7 \times 9} - \boxed{3 \times 1} - \boxed{4 \times 8}) \times 10^1 + 4 \times 8$$

هنوز هم دو ضرب دو رقمی یعنی $3 \times 11$ و $10 \times 11$ باید به صورت بازگشتی انجام داد. دقت کنید که ضرب $10 \times 11$ را نمی‌توان به روش شیفت دادن انجام داد. زیرا به صورت $10^x$ توضیح داده شده در صورت مسئله نیست.

$$10 \times 11 = (1 \times 10^1 + 0)(1 \times 10^1 + 1) = (1 \times 1) \times 10^2 + \boxed{1 \times 2} - \boxed{1 \times 1} - \boxed{0 \times 1} \times 10^2 + 0 \times 1$$

$$3 \times 11 = (0 \times 10^1 + 3)(1 \times 10^1 + 1) = 0 \times 1 \times 10^2 + (\boxed{3 \times 2} - \boxed{0 \times 1} - \boxed{3 \times 1}) \times 10^1 + 1 \times 3$$

با شمارش نتایج داخل مستطیل به این نتیجه می‌رسیم که برای ضرب دو عدد 1234 و 5618 به 13 ضرب اعداد یک رقمی نیاز داریم.

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



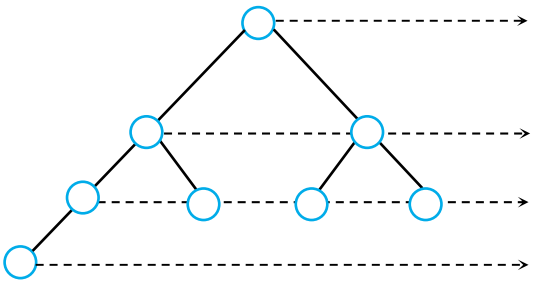
**۶ـ گزینه «۱»** برای محاسبه یک چند جمله‌ای بهترین روش، روش هورنر است که به صورت زیر چند جمله‌ای را محاسبه می‌کند.

$$a_0 + \times(a_1 + x(a_2 + \ldots + x(a_{n-1} + x(a_n)\ldots))$$

این عبارت به $n$ ضرب نیاز دارد.

---

**۷ـ گزینه «۳»** درخت مربوط به مقایسه لیست داده شده در جست‌وجوی دو دویی به صورت زیر می‌باشد:



1 مقایسه
2 مقایسه
3 مقایسه
4 مقایسه
$\Rightarrow$ 21 مقایسه

بنابراین میانگین تعداد مقایسه‌ها در جست‌وجوی موفق برابر است با: $\frac{21}{8} \simeq 2.6$

---

**۸ـ گزینه «۴»** چنین الگوریتمی با روش تقسیم و غلبه به دست می‌آید. (به متن درس مراجعه کنید.)

---

**۹ـ گزینه «۳»** راه‌حل مناسب ضرب اعداد بزرگ استفاده از روش تقسیم و غلبه می‌باشد، اما در صورتی که از روش بیان شده در گزینه چهارم استفاده شود آنگاه هزینه، بیشتر از روش گزینه سوم خواهد بود.

---

**۱۰ـ گزینه «۲»** ترتیب بهینه ضرب ماتریس‌های داده شده به صورت مقابل می‌باشد: $[A \times (B \times C)] \times D$

که تعداد ضرب‌های مورد نیاز برابر با 2856 می‌باشد.

$$B_{5\times89} \times C_{89\times3} \xrightarrow{\text{تعداد ضرب‌ها}} 5\times89\times3 \qquad A_{13\times5}(B\times C)_{5\times3} \xrightarrow{\text{تعداد ضرب‌ها}} 13\times5\times3$$

$$[A\times(B\times C)]_{13\times3} \times D_{3\times34} \xrightarrow{\text{تعداد ضرب‌ها}} 13\times3\times34$$

بنابراین: تعداد کل ضرب‌ها $: 5\times89\times3+13\times5\times3+13\times3\times34 = 2856$

---

**۱۱ـ گزینه «۱»** سکه را به سه مجموعه 3 تایی تقسیم کنید 2 مجموعه را با ترازو مقایسه کنید، اگر برابر شدند سکه تقلبی در مجموعه دیگر است. یکی از مجموعه‌ها را با آن مقایسه می‌کنیم. در می‌یابیم که سکه تقلبی سبک‌تر یا سنگین‌تر است. حال از مجموعه سکه‌هایی که سکه تقلبی در آنهاست دو تا را با هم مقایسه می‌کنیم، اگر برابر بودند سومی تقلبی است اگر نبودند با توجه به اینکه می‌دانیم سکه تقلبی سبک‌تر یا سنگین‌تر است آن را می‌یابیم. اما، فرض کنید دو مجموعه‌ای که در ابتدا با هم مقایسه شدند برابر نباشند؛ لذا یکی از آنها را با دیگری مقایسه کنید. اگر مساوی بودند، در می‌یابیم سکه در مجموعه دیگر است و وزن آن با هم با توجه به مقایسه اول به دست می‌آید و با مقایسه 2 سکه از آن مجموعه باز هم سکه تقلبی را می‌یابیم. اما باز فرض کنید بعد از اینکه در مقایسه اول با هم برابر نشدند و یکی از مجموعه را با مجموعه دیگر مقایسه کردیم برابر نشوند. باز هم در می‌یابیم سکه در همین مجموعه‌ی انتخاب شده است و وزن آن نیز کمتر (یا بیشتر) است و با مقایسه‌ی دو سکه از آن سکه تقلبی را می‌یابیم.

---

**۱۲ـ گزینه «۱»** در جستجوی دودویی همواره مقدار مورد جستجو، باید با عنصر وسط یعنی $\frac{(p+q)}{2}$ مقایسه شود.

---

**۱۳ـ گزینه «۳»** ترتیب بهینه ضرب و تعداد ضرب‌ها به صورت زیر تعیین می‌گردد:

$$B_{2\times25} \times C_{25\times3} \Rightarrow \text{تعداد ضرب‌ها} = 2\times25\times3 \qquad (B\times C)_{2\times3} \times D_{3\times4} \Rightarrow \text{تعداد ضرب ها} = 2\times3\times4$$

$$A_{10\times2} \times [(B\times C)\times D]_{2\times4} \Rightarrow \text{تعداد ضرب‌ها} = 10\times2\times4$$

بنابراین تعداد کل ضرب‌ها برابر است با: $150 + 24 + 80 = 254$

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۱۴ـ گزینه «۲»** با هر عمل Union دو مجموعه تک عنصر با هم ترکیب می‌شوند، در نتیجه بعد از u بار مجموعه‌های تک عنصر برابر $n-2u$ بوده که ممکن است منفی شود بنابراین بین صفر و این عدد ماکزیمم می‌گیریم.

**۱۵ـ گزینه «۳»** رابطهٔ بازگشتی محاسبه تعداد ضرب‌ها به صورت روبرو خواهد بود:

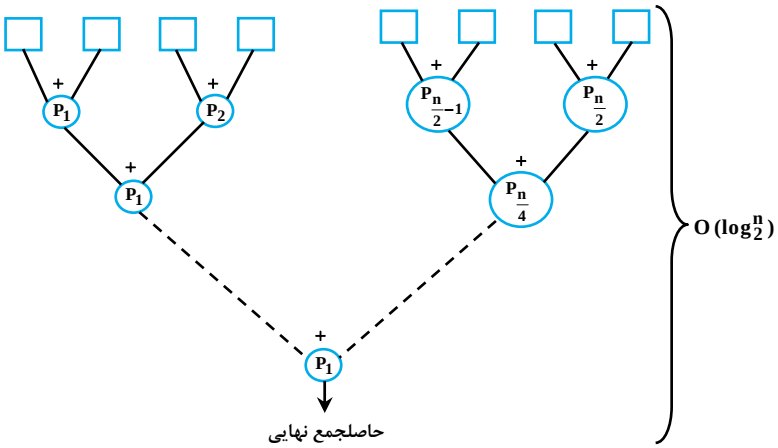
$$\begin{cases} T(n)=7T(\frac{n}{2}) & n>2 \\ T(2)=8 \end{cases}$$

حال با توجه به این که مسئله کوچک ضرب ماتریس‌ها ضرب ماتریس‌های $2\times 2$ در نظر گرفته شده است، بنابراین $T(2)=8$ و خواهیم داشت:

$$t(8)=7t(4)=7^2t(2)=7^2\times 8=392$$

بنابراین تعداد ضرب‌های عددی مورد نیاز عبارتست از: 392.

**۱۶ـ گزینه «۲»** اگر آرایه موردنظر را $A[1..n]$ درنظر بگیریم و پردازنده‌ها را با $P_1...P_{\frac{n}{2}}$ نشان دهیم، آنگاه می‌توان ساختار درختی مقابل را برای جمع این اعداد درنظر گرفت:




که با توجه به عمق این ساختار، زمان محاسبه از مرتبه $O(\lg_2^n)$ می‌باشد.

**۱۷ـ گزینه «۳»** با استفاده از روش درختی می‌توان در زمان $O(\log_2^n)$ ضرب‌ها را انجام داد.

**۱۸ـ گزینه «۱»** با جایگذاری $\binom{k}{k}$ به گزینه (۱) می‌رسیم. زیرا فقط یک جمله محاسبه می‌شود.

**۱۹ـ گزینه «۲»** برای رسیدن به گروه یک تیمی باید هفت بار دو دسته 64 تایی با یکدیگر مسابقه دهند.

**۲۰ـ گزینه «۳»** برای مؤلفه اول و دوم مختصات، سه حالت می‌توانند رخ دهند، یک واحد بیشتر، یک واحد کمتر و مساوی. منظور نقاط ربع اول است که زیر خط $y=n-x$ قرار دارند. حال کافی است یک مربع به مرکز نقطه مورد نظر که حداکثر ۸ نقطه روی محیط آن قرار دارند را حرکت داد و تعداد حالات نقاطی که روی آن قرار می‌گیرند را در نظر گرفت.

**۲۱ـ گزینه «۲»** رابطهٔ بازگشتی مربوط به ضرب استراسن به صورت مقابل می‌باشد: $T(n)=7T(\frac{n}{2})+1$

با توجه به این که مسئله کوچک، ماتریس‌های $2\times 2$ در نظر گرفته شده‌اند، بنابراین $T(2)=1$ و برای محاسبه $T(8)$ خواهیم داشت:

$$T(8)=7T(4)+1$$

$$T(4)=7T(2)+1$$

$$T(2)=1 \Rightarrow T(4)=8$$

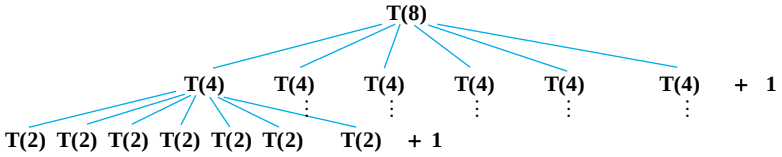
دقت کنید که این مسئله با مسئله مشابه که در آن تعداد ضرب‌های عددی خواسته شده، تفاوت دارد. در اینجا تعداد فراخوانی بازگشتی به الگوریتم خواسته شده، بنابراین $T(2)=1$ است، ولی در سؤال 16 تعداد ضرب‌های عددی $T(2)=8$ است. در نتیجه داریم: $\Rightarrow T(8)=7\times 8+1=57$

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



درخت بازگشت برای $T(8)$ را می‌توان به صورت زیر در نظر گرفت:




---

**۲۲ـ گزینه «۲»** در حالت مرتبهٔ زمانی الگوریتم‌های تقسیم و غلبه که در آن هربار $a$ زیر مسئله به اندازه $\frac{n}{b}$ ایجاد می‌شود، هزینه ترکیب زیر مسئله $f(n)$ می‌باشد که به صورت زیر خواهد بود:

$$T(n)=aT(\frac{n}{b})+f(n)$$

بنابراین با توجه به صورت تست داریم:

تابع پیچیدگی روش اول: $T(n)=3T(\frac{n}{2})+\theta(n^2\sqrt{n})$

تابع پیچیدگی روش دوم: $T(n)=4T(\frac{n}{2})+\theta(n^2)$

تابع پیچیدگی روش سوم: $T(n)=5T(\frac{n}{2})+\theta(n\log n)$

با استفاده از master theorem زمانهای سه روش به صورت زیر قابل محاسبه است:

روش اول: $T(n)\in\theta(n^2\sqrt{n})$　　روش دوم: $T(n)\in\theta(n^2\log n)$　　روش سوم: $T(n)\in\theta(n^{\log_2^5})$

بنابراین روش دوم دارای زمان کمتری نسبت به بقیه می‌باشد.

---

**۲۳ـ گزینه «۳»** الگوریتم مرتب‌سازی ادغامی، همواره از همین پیچیدگی برخوردار است.

---

**۲۴ـ گزینه «۳»** با توجه به متن درس، رابطه بازگشتی تعداد جمع و تفریق‌های الگوریتم استراسن در گزینه (۳) آمده است.

---

**۲۵ـ گزینه «۱»** با توجه به الگوریتم بازگشتی داده شده اگر رابطه بازگشتی مربوط به مرتبه زمانی آن را $T(n)$ درنظر بگیریم، می‌توان نوشت:

$$T(n)=T(n-1)+T(n-2)$$

بنابراین اگر دو رابطه بازگشتی زیر را درنظر بگیریم خواهیم داشت:

$$A(n)=2A(n-1)\ ,\ B(n)=2B(n-2)$$

لذا می‌توان نوشت:

$$B(n)<T(n)<A(n)$$

و در نتیجه:

$$\theta(2^{\frac{n}{2}})<T(n)<\theta(2^n)$$

---

**۲۶ـ گزینه «۱»** با روش درختی می‌توان در زمان $\lceil\log_2^{2k}\rceil$ عمل جمع $2k$ عنصر را انجام داد (دقت کنید که $\lceil\log_2^{2k}\rceil=1+\lceil\log_2^{k}\rceil$).

---

**۲۷ـ گزینه «۲»** دو حالت را در جستوجو (که در این حالت جستوجوی خطی است) می‌توان درنظر گرفت:

۱ـ در جستوجوی ناموفق همواره به $n$ مقایسه نیاز است.

۲ـ در جستوجوی موفق با درنظر گرفتن همه حالات میانگین تعداد مقایسه‌ها عبارتست از:

$$\frac{1+2+3+\ldots+n}{n}=\frac{n(n+1)}{n}=\frac{n+1}{2}$$

بنابراین با توجه به این‌که با احتمال $\frac{1}{2}$ جستوجوی موفق و با احتمال $\frac{1}{2}$ جستوجوی ناموفق انجام می‌شود، به‌طور میانگین تعداد مقایسه‌ها عبارتست از:

$$\frac{1}{2}\times n+\frac{1}{2}\times\frac{n+1}{2}\approx\frac{3n}{4}$$

بنابراین $\frac{3}{4}$ کل عناصر در حالت میانگین مورد جستوجو قرار می‌گیرند.

---

**۲۸ـ گزینه «۲»** گزینه دوم نشان‌دهندهٔ رابطه درست بین تعداد مقایسه‌ها در جستجوی موفق و ناموفق در الگوریتم جستجوی دودویی می‌باشد.

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۲۹ـ گزینه «۳»** با توجه به مرتب بودن آرایه A می‌توان از راه حل تقسیم غلبه در زمان $O(\log_2^n)$ مسئله را حل نمود.

برای این کار از روشی شبیه به جست‌وجوی دودویی استفاده می‌نماییم، اگر اندیس اولین مکان آرایه را $L$ و اندیس آخرین مکان آرایه را $R$ در نظر بگیریم، آنگاه ابتدا مقایسه زیر را انجام می‌دهیم:

$m = [\frac{L+R}{2}]$

if $A[m] = m$ $\Rightarrow$ عنصر مورد نظر وجود دارد.

if $A[m] > m$ $\Rightarrow$ مقدار تمام عناصر موجود در نیمه دوم آرایه از اندیس آنها بیشتر است بنابراین، جست‌وجو در نیمه اول آرایه انجام می‌شود.

بدون تغییر $L \rightarrow$ و $R = m-1$

if $A[m] < m$ $\Rightarrow$ مقدار تمام عناصر موجود در نیمه اول آرایه از اندیس آن‌ها کمتر است بنابراین، جست‌وجو در نیمه دوم آرایه انجام می‌شود..

$L = m+1$ و بدون تغییر $R \rightarrow$

if $L > R$ $\Rightarrow$ عنصر مورد نظر وجود ندارد.

---

**۳۰ـ گزینه «۱»** با استفاده از الگوریتم زیر می‌توان میانه دو آرایه مرتب را تعیین نمود:

```
Median(A, B, n)
{
  mid1 = A[n/2]   //   میانه آرایه A
  mid2 = B[n/2]   //   میانه آرایه B
  if (mid1 < mid2)
        return Median(A[n/2,...n], B[1...n/2], n/2)
  else if (mid1 > mid2)
        return Median(A[1...n/2], B[n/2...n], n/2)
  else
        return mid1
}
```

همان‌گونه که ملاحظه می‌گردد در این الگوریتم با مقایسه میانه دو آرایه $A$ و $B$ نیمی از هر یک از آرایه‌ها حذف می‌گردد و الگوریتم به صورت بازگشتی فراخوانی می‌شود، در هر فراخوانی یک عنصر از آرایه $A$ و یک عنصر از آرایه $B$ خوانده می‌شود. بنابراین در کل تعداد خواندن‌ها از آرایه به صورت زیر خواهد بود:

$T(2n) = T(n) + 2$

---

**۳۱ـ گزینه «۲»** مرتبه زمانی قطعه برنامه داده شده $O(2\sqrt{n})$ می‌باشد. اگر از جست‌وجوی دودویی استفاده کنیم آنگاه مرتبه زمانی $O(\sqrt{n}\log n)$ خواهد بود.

---

**۳۲ـ گزینه «۲»** برای محاسبه حاصلضرب دو ماتریس $n \times n$ باید $n^3$ ضرب عددی انجام شود.

---

**۳۳ـ گزینه «۱»** هر پردازنده می‌تواند 2 عدد را با هم جمع کند. هدف ما حداقل نمودن زمان محاسبه است و در تعداد پردازنده‌ها محدودیتی نداریم. به ازای اعداد ۱ تا $k$، پردازنده $i$ اعداد $2i-1$ و $2i$ را در مرحله اول جمع می‌کند. در مرحله دوم $\frac{k}{2}$ عدد داریم که هر یک حاصل جمع دو عدد از $k$ عدد اولیه بودند. در این مرحله همانند مرحله قبل، آن‌ها را دو به دو با هم جمع می‌کنیم. سپس از $\log_2 k$ مرحله، یک عدد داریم که برابر مجموع این $k$ عدد است.

---

**۳۴ـ گزینه «۲»** طبق قضیه اصلی زمانی، رابطه بازگشتی داده شده برابر $\theta(n^{\log_2^7})$ می‌باشد (البته گزینه ۴ نیز صحیح می‌باشد).

---

**۳۵ـ گزینه «۱»** مرتبه زمانی الگوریتم داده شده را می‌توان به‌صورت روبه‌رو درنظر گرفت:

$\theta(2^{\frac{n}{2}}) < T(n) < \theta(2^n)$

---

**۳۶ـ گزینه «۲»** اندیس‌های استفاده شده برای mid به‌صورت روبه‌رو می‌باشد:

64,96,112,120,124,126,127,128

---

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۳۷ـ گزینه «۳»** زمان مورد نیاز عبارتست از: $\log_2^{2n} = \log_2^n + 1$

با توجه به این که n توانی از 2 است گزینه ۱ هم درست است.

**۳۸ـ گزینه «۱»** مرتبه زمانی جمع و ضرب دو ماتریس $n \times n$ به ترتیب $\theta(n^2)$ و $\theta(n^3)$ می‌باشد.

**۳۹ـ گزینه «۱»** با داشتن $\frac{n}{2}$ پردازنده و به‌صورت درختی می‌توان n عدد را در زمان $O(\log_2^n)$ در یکدیگر ضرب نمود.

**۴۰ـ گزینه «۱»** می‌توان ضرب ماتریس $A_k$ در بردار V را با استفاده از روش تقسیم و حل به صورت زیر در نظر گرفت:

$$\left[\begin{array}{c|c} A_{k-1} & A_{k-1} \\ \hline A_{k-1} & -A_{k-1} \end{array}\right]\left[\begin{array}{c} V_1 \\ \hline V_2 \end{array}\right]$$

حال کافیست دو ضرب $A_{k-1}.V_1$ و $A_{k-1}.V_2$ محاسبه گردد و سپس با توجه به نتیجه می‌توان $A.V$ را محاسبه نمود. بنابراین مرتبه زمانی الگوریتم به صورت مقابل خواهد بود: $T(n) = 2T(\frac{n}{2}) + \theta(n) \Rightarrow T(n) \in \theta(n \log n)$

**۴۱ـ گزینه «۲»** به متن درس مراجعه نمایید. (برای ضرب‌های $\frac{n}{2} \times \frac{n}{2}$ باز همان محاسبات قبلی لازم است.)

**۴۲ـ گزینه «۴»** در این تابع و در حلقه while در گذر اول گره‌های مکان زوج حذف می‌شود و L یکی یکی جلو می‌رود و دوباره در لیست جدید یک‌درمیان گره‌ها حذف می‌شوند. می‌توان از روابط بازگشتی $W(2n) = 2W(n) - 1$ و $W(2n+1) = 2W(n) + 1$ و $W(1) = 1$ استفاده نمود. تابع $W(n)$ برای $n = 729$ مقدار 435 و برای $n = 2200$ مقدار 305 را برمی‌گرداند.

**۴۳ـ گزینه «۱»** در حالت کلی با حداکثر $2n$ مقایسه و شروع از عنصر گوشه سمت چپ و پایین و یا عنصر گوشه راست و بالا در ماتریس می‌توان جست‌وجو را انجام داد. به عنوان مثال ماتریس زیر را در نظر بگیرید:

$$\begin{bmatrix} 1 & 3 & 5 & 6 \\ 2 & 4 & 7 & 15 \\ 6 & 5 & 8 & 17 \\ 7 & 11 & 13 & 18 \end{bmatrix}$$

برای جست‌وجوی یک عنصر مانند 15 کافیست با 7 مقایسه شود. با توجه به این که $15 > 7$ بنابراین عنصر مورد جست‌وجو از تمام عناصر ستون اول بزرگتر است. حال 15 را با 11 مقایسه می‌کنیم و با توجه به این که $15 > 11$ بنابراین از تمام عناصر ستون دوم نیز بزرگتر می‌باشد. در نتیجه 15 را با 13 مقایسه می‌کنیم و با توجه به این که $15 > 13$، پس 15 تنها می‌تواند در ستون چهارم وجود داشته باشد که حداکثر با 4 مقایسه نتیجه مشخص می‌گردد. به عنوان یک مثال دیگر برای جست‌وجوی 4 ابتدا مقایسه $4 < 7$ انجام می‌شود و بنابراین 4 نمی‌تواند در سطر آخر باشد. حال مقایسه $4 < 6$ در سطر ماقبل آخر انجام می‌گیرد. پس مقایسه باید در سطر دوم انجام شود. با مقایسه $4 < 5$ به این نتیجه می‌رسیم که باید مقایسه در سطر دوم انجام شود و در سطر دوم نیز با یک جست‌وجوی عنصر مورد نظر پیدا می‌شود.

**۴۴ـ گزینه «۲»** کافیست جست‌وجو را از گوشه سمت راست بالا و یا سمت چپ پایین شروع کنیم و حداکثر با $2n-1$ مقایسه عمل جست‌وجو را انجام دهیم.

**۴۵ـ گزینه «۳»** با استفاده از روش هورنر می‌توان در زمان $O(n)$ چند جمله‌ای مذکور را ارزشیابی نمود.

**۴۶ـ گزینه «۳»** کافیست از ایده تقسیم و غلبه ساده زیر استفاده کنیم.

کل دانشجویان را به دو دسته با $\frac{n}{2}$ نفر تقسیم می‌کنیم و با سؤال از یک دسته یکی از دسته‌ها حذف می‌شود. سپس دانشجویان باقیمانده به دو دسته با اندازه $\frac{n}{4}$ تقسیم می‌شوند و با یک سؤال یکی از این گروه‌ها حذف می‌گردد. با ادامه همین روند با $\lceil \log_2^n \rceil$ سؤال می‌توان فرد موردنظر را یافت.

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۴۷ـ گزینه «۲»** با توجه به خواص اعداد فیبوناچی و با توجه به این که توان $n$ام ماتریس $\begin{bmatrix}0 & 1\\ 1 & 1\end{bmatrix}$ در زمان $O(\log_2^n)$ قابل محاسبه می‌باشد، نتیجه $F_n$ در زمان $O(\log_2^n)$ قابل محاسبه خواهد بود.

**۴۸ـ گزینه «۴»** رابطه بازگشتی مربوط به ضرب دو ماتریس $n \times n$ در روش Strassen به صورت مقابل می‌باشد: $T(n) = 7T(\frac{n}{2})\ ,\ T(1) = 1$

جواب این رابطه برابر است با: $T(n) = n^{\log_2^7} \approx n^{2/81}$

**۴۹ـ گزینه «۱»** زمان یافتن میانه دو لیست مرتب برابر $O(\log_2^n)$ می‌باشد و اگر $k = \frac{n}{2}$ باشد همان مسئله انتخاب میانه را خواهیم داشت.

**۵۰ـ گزینه «۱»** اگر همه سطرهای ماتریس (از چپ به راست) و همه ستون‌های ماتریس (از بالا به پایین) به صورت صعودی مرتب باشند، با مقایسه $x$ با راست‌ترین عنصر سطر اول (بالاترین عنصر ستون آخر) می‌توان در هر مقایسه یک سطر یا ستون را از ماتریس کنار گذاشته و با بقیه ماتریس عمل جست‌وجو را ادامه داد. (البته می‌توان همین عمل را با شروع از سمت چپ‌ترین عنصر سطر آخر نیز انجام داد.)

**۵۱ـ گزینه «۴»** با استفاده از توضیح بیان شده در تست‌های قبلی حداکثر با $2n$ مقایسه می‌توان جست‌وجو را انجام داد.

**۵۲ـ گزینه «۲»** با استفاده از الگوریتم تقسیم و غلبه می‌توان در زمان $O(n\log n)$ مسئله موردنظر را حل کرد.

**۵۳ـ گزینه «۴»** با استفاده از الگوریتم تقسیم و غلبه بیان شده در فصل اول، می‌توان این محاسبه را در زمان $O(\log_2^b)$ انجام داد. دقت کنید که با توجه به $n$ بیتی بودن عدد $b$ می‌توان نوشت $n = [\log_2^b]$. بنابراین: $O(\log_2^b) = O(n)$

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



# فصل نهم

## «برنامه‌ریزی پویا (Dynamic Programming)»

### تست‌های طبقه‌بندی شده کنکوری فصل نهم

**۱ـ** می‌خواهیم چهار ماتریس $A(30\times1)$، $B(1\times40)$، $C(40\times10)$ و $D(10\times25)$ را در هم ضرب کنیم $(A\times B\times C\times D)$. حداقل ممکن تعداد عمل ضرب عناصر این ماتریس‌ها چند تا است؟ (مهندسی کامپیوتر ـ سراسری ۷۷)

۱) 1250 ۲) 1400 ۳) 20700 ۴) 11750

**۲ـ** برنامهٔ زیر برای محاسبه سری فیبوناچی است. گزینهٔ صحیح را انتخاب کنید. (علوم کامپیوتر ـ سراسری ۸۰)

```
function fibo (n: integer): integer;
var f, f1, f2, i: Integer;
begin
    f1 = 1;
    f2 = 1;
    for  i :=  to n do begin
        f := f1 + f2,
        f2 := f2;
        f2 := f;
    end;
    fibo := f;
end;
```

۱) الگوریتم این برنامه از ردهٔ برنامه‌ریزی پویا و مرتبهٔ آن خطی است.

۲) الگوریتم این برنامه از ردهٔ تقسیم و غلبه و مرتبه آن خطی است.

۳) الگوریتم این برنامه از ردهٔ برنامه‌ریزی پویا است و مرتبهٔ آن بیش از خطی است.

۴) الگوریتم این برنامه از ردهٔ تقسیم و غلبه است و مرتبهٔ آن بیش از خطی است.

**۳ـ** $n$ بازه $[l_i, u_i]$ برای $i = 1, \ldots, n$ داده شده‌اند ($L_i$ ⟵ ⟶ $u_i$). می‌خواهیم $S$ مجموعه‌ی بازه‌های دو به دو ناهمپوشان با بیشترین طول (مجموع طول بازه‌های انتخاب شده) را پیدا کنیم. برای این کار، الگوریتم زیر پیشنهاد شده است:

بازه‌ها را به یک ترتیب انتخاب کن، برای هر بازه انتخابی، بازه‌های دیگر که با آن هم پوشانی دارند را حذف کن و این کار را تکرار کن. کدام گزینه ترتیب درست برای این الگوریتم حریصانه است؟ (مهندسی کامپیوتر ـ سراسری ۸۱)

۱) ترتیب بر حسب $l_i$ بازه‌ها

۲) ترتیب برحسب $u_i$ بازه‌ها

۳) ترتیب بر حسب طول بازه‌ها

۴) هیچ‌کدام از ترتیب‌های فوق درست نیست.

**۴ـ** در ضرب سه ماتریس به ابعاد (علوم کامپیوتر ـ سراسری ۸۱)

| ماتریس | ابعاد |
|---|---|
| $N_1$ | $w\times x$ |
| $N_2$ | $x\times y$ |
| $N_3$ | $y\times z$ |

تحت چه شرایطی ضرب $(N_1N_2)N_3$ سریع‌تر از ضریب $N_1(N_2N_3)$ است؟

۱) $x > y$ ۲) $\frac{1}{x}+\frac{1}{z}<\frac{1}{w}+\frac{1}{y}$ ۳) $\frac{1}{w}+\frac{1}{x}<\frac{1}{y}+\frac{1}{z}$ ۴) $w+x>y+z$

**۵ـ** فرض کنید $T(n)$ برابر تعداد پرانتزگذاری‌های مختلف برای ضرب $n$ ماتریس در هم باشد. در این صورت، $T(1)=T(2)=1$ و $T(n)=?$: (مهندسی کامپیوتر ـ سراسری ۸۲)

۱) $T(n)=\sum_{i=1}^{n} T(i)\times T(n-i)$

۲) $T(n)=\sum_{i=1}^{n-1} T(i)\times T(n-i)$

۳) $T(n)=\sum_{i=1}^{n} T(i)\times T(n-i+1)$

۴) $T(n)=\sum_{i=1}^{n-1} T(i)\times T(n-i+1)$

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



۶ـ می‌خواهیم یک درخت دودویی جست‌وجو با 6 عنصر $a_1 < a_2 < \ldots < a_6$ بسازیم تا متوسط عمق عناصر در آن کمینه شود. اگر $p_i$ احتمال جستجوی $a_i$ باشد، متوسط عمق برابر $\sum_{i=1}^{6} p_i \, depth[a_i]$ تعریف می‌شود. اگر $p_1 = \frac{2}{7}$ و $p_i = \frac{1}{7}$ باشد، متوسط عمق درخت بهینه چند است؟ (عمق ریشه، صفر فرض می‌شود.) (علوم کامپیوتر ـ سراسری ۸۲)

۱) $\frac{8}{7}$ ۲) $\frac{9}{7}$ ۳) $\frac{10}{7}$ ۴) $\frac{11}{7}$

۷ـ فرمول بهینه‌سازی بازگشتی زیر: (علوم کامپیوتر ـ سراسری ۸۳)

$$m[j,j] = \begin{cases} 0 & \text{if} \quad i = j \\ \min\{m[i,k] + m[k+1,j] + p_{i-1}p_k p_j\} & \text{if} \quad i < j \end{cases}$$

برای کدام مسئله می‌باشد و بهترین روش نوشتن الگوریتم آن کدام است؟

۱) مسئله ضرب دو ماتریس Dynamic programming

۲) مسئله پرانترگذاری ضرب n ماتریس Divide and conqure

۳) مسئله بهینه‌سازی درخت جستجوی باینری Dynamic programming

۴) مسئله پرانتزگذاری ضرب n ماتریس Dynamic programming

۸ـ کدام گزینه زیر صحیح نمی‌باشد؟ (مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۴)

۱) الگوریتم بازگشتی، یک الگوریتم از بالا به پایین است که همه زیر مسائل را حداقل یک بار حل می‌کند.

۲) الگوریتم طراحی شده توسط روش برنامه‌ریزی پویا، یک الگوریتم از پایین به بالاست که همه زیر مسائل را تنها یک بار حل می‌کند.

۳) الگوریتم‌های طراحی شده توسط روش برنامه‌ریزی پویا (نوع استاندارد یا Memoized) زیر مسائل حل شده را فقط یک بار حل می‌کنند.

۴) الگوریتم طراحی شده توسط روش برنامه‌ریزی پویا و از نوع Memoized یک الگوریتم بالا به پایین است که همه زیرمسائل لازم را تنها یک بار حل می‌کند.

۹ـ می‌خواهیم برای ماتریس‌های $M_{1_{(10\times20)}}$ و $M_{2_{(20\times50)}}$ و $M_{3_{(50\times1)}}$ و $M_{4_{(1\times100)}}$ ترکیب بهینه پرانتزبندی را پیدا نماییم تا تعداد ضرب‌های کل جهت محاسبه عبارت روبرو حداقل گردد: $M = M_1 \times M_2 \times M_3 \times M_4$

این ترکیب بهینه عبارت است از: (مهندسی کامپیوتر ـ آزاد ۸۴)

۱) $(M_1 \times (M_2 \times M_3)) \times M_4$ ۲) $((M_1 \times M_2) \times M_3) \times M_4$

۳) $M_1 \times ((M_2 \times M_3) \times M_4)$ ۴) هیچ‌کدام

۱۰ـ الگوریتم زیر هزینه ضرب بهینه n ماتریس $M_1 \times M_2 \times \cdots \times M_n$ را که ابعاد $M_i$ برابر $d_{i-1} \times d_i$ است مشخص می‌کند. مجموع تعداد خواندن درایه‌های C در این الگوریتم چندتاست؟ (مهندسی کامپیوتر ـ سراسری ۸۵)

```
for i =:1 to n do
c[i,i] =0
for ℓ:= 2 to n do
for i :=1 to n− ℓ + 1 do
{ j := i + ℓ − 1
c[i,j]=min(c[i,k] + c[k + 1, j] + d_{i−1} * d_k * d_j)
i<k≤j
}
```

۱) $\sum_{\ell=1}^{n} (2\ell - 1)(n - \ell)$

۲) $\sum_{\ell=2}^{n} (2\ell - 1)(n - \ell)$

۳) $\sum_{\ell=2}^{n} 2(\ell - 1)(n - \ell + 1)$

۴) $\sum_{\ell=2}^{n} 2(\ell - 1)(n - \ell)$

۱۱ـ تعداد پرانتزگذاری یک عبارت جبری با n عملوند (operand) که مقدار آن تغییر نکند چند است؟ (علوم کامپیوتر ـ سراسری ۸۵)

۱) $n!$ ۲) $C_n = \frac{1}{n+1}\binom{2n}{n}$ ۳) $C_{n-1} = \frac{1}{n}\binom{2(n-1)}{n-1}$ ۴) مشخص نمی‌باشد.

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



۱۲ـ فرض کنید که $X = x_1x_2....x_m$ و $Y = y_1y_2...y_n$ دو رشته با الفبای $\{A,C,G,T\}$ باشند. طولانی‌ترین زیر دنباله مشترک (LCS) رشته $X$ و $Y$ با برنامه‌ریزی پویا به دست می‌آید. برای محاسبه $C[i,j]$ که $1 \le i \le m$ یا $1 \le j \le n$ است، در بدترین حالت چند خانه بررسی می‌شود؟

(مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۶)

۱) n ۲) 2 ۳) 3 ۴) n+m

۱۳ـ برای سه ماتریس با ابعاد زیر

(علوم کامپیوتر ـ سراسری ۸۶)

| | ستون | سطر |
|---|---|---|
| $N_1$ | m | n |
| $N_2$ | n | p |
| $N_3$ | p | q |

اگر بخواهیم تعداد ضرب‌های $N_1 \times (N_2 \times N_3)$ با $(N_1 \times N_2) \times N_3$ یکسان باشد باید:

۱) فقط $m = n = p = q$ باشد. ۲) $m = n$ یا $p = q$ ۳) $\frac{1}{m}+\frac{1}{n}=\frac{1}{q}+\frac{1}{p}$ ۴) $\frac{1}{m}-\frac{1}{n}=\frac{1}{q}-\frac{1}{p}$

۱۴ـ یک تیر چوبی به طول $L$ متر را می‌خواهیم از فاصله‌های $I_1, I_2, ..., I_n$ (نسبت به سر سمت چپ آن تیر) ببریم. یعنی $0 < I_1 < I_2 < ... < I_n < L$ می‌دانیم که هزینه‌ی برش یک تیر به طول $r$ از هر نقطه‌ای بر روی آن تیر برابر $r$ است. می‌خواهیم تیر را از این $n$ نقطه به ترتیبی ببریم که کل هزینه‌ی صرف شده برای این کار کمینه شود.

(مهندسی کامپیوتر ـ سراسری ۸۷)

۱) این مسئله راه‌حل چند جمله‌ای ندارد.
۲) این مسئله راه‌حل پویا از مرتبه‌ی چند جمله‌ای دارد.
۳) این مسئله راه‌حل حریصانه از مرتبه‌ی چند جمله‌ای دارد.
۴) این مسئله راه‌حل تقسیم و حل از مرتبه‌ی چند جمله‌ای دارد.

۱۵ـ کدام عبارت در مورد روش برنامه‌ریزی پویا نادرست است؟

(مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۷)

۱) این روش جزو روش‌های بالا به پایین است.
۲) مسائل قابل حل با این روش باید تعریف بازگشتی داشته باشند.
۳) در این روش از یک حافظه کمکی برای ثبت نتایج میانی استفاده می‌شود.
۴) این روش برای حل مسائلی مناسب است که در آن‌ها زیر مسائل روی هم افتادگی داشته باشند.

۱۶ـ محاسبه کدام تابع با روش برنامه‌نویسی پویا می‌تواند میزان محاسبات مورد نیاز را نسبت به روش بازگشتی بیش از بقیه کاهش دهد؟

(مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۸)

۱) $F(n) = \sum_{i=1}^{n-1} F(i)$ ۲) $F(n) = aF(n-1) + b$
۳) $F(n) = aF(n-1) + bF(n-2)$ ۴) $F(n) = aF\left(\left\lfloor \frac{n}{2} \right\rfloor\right) + bF\left(\left\lceil \frac{n}{2} \right\rceil\right)$

۱۷ـ گراف جهت‌دار $G = (V,E)$ نشان‌دهنده‌ی یک شبکه‌ی کامپیوتری با $n$ رأس است که وزن هر یال $(u,v)$ برابر احتمال خرابی (قطع کامل) آن یال است که با $p(u,v)$ نشان می‌دهیم و داریم $0 \le p(u,v) \le 1$. می‌خواهیم در این گراف احتمال خرابی قابل اعتمادترین مسیر از هر رأس $i$ به یک رأس $j$ را پیدا کنیم. این مسیری است که احتمال خرابی آن کمینه است. فرض کنید که احتمال خرابی یال‌ها مستقل از هم هستند. می‌خواهیم از الگوریتم Floyd برای حل این مسئله به صورت زیر استفاده کنیم.

```
for k = 1 to n do
  for i = 1 to n do
    for j = 1 to n do
      (a).....
```

اگر $p_{ij}$ احتمال خرابی یک مسیر بین دو رأس $i$ و $j$ با کم‌ترین احتمال خرابی باشد، چه عبارتی در سطر (a) قرار دهیم تا الگوریتم کار کند؟

(مهندسی کامپیوتر ـ سراسری ۸۹)

۱) $p_{ij} = \min\{p_{ij}, \max\{p_{ik}, p_{kj}\}\}$ ۲) $p_{ij} = \max\{p_{ij}, \min\{p_{ik}, p_{kj}\}\}$
۳) $p_{ij} = \min\{p_{ij}, (p_{ik} + p_{kj})\}$ ۴) $p_{ij} = \min\{p_{ij}, (p_{ik} * p_{kj})\}$

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



۱۸ـ یک دنباله‌ی $X=(x_1,x_2,...,x_n)$ داده شده است. $Y=(y_1,...,y_k)$ یک «زیردنباله» از $X$ است. اگر برای هر $1\le i\le k$، $y_i=x_j$ و نیز $j_1<j_2<...<j_k$. مثلاً $(2,3,7,9)$ و $(2,3,9)$ زیر دنباله‌های $(2,3,1,7,4,9)$ هستند. با داشتن یک دنباله‌ی $L$ با $n$ عنصر، می‌خواهیم بزرگترین (از نظر تعداد عناصر) زیردنباله‌ی افزایشی $L$ را به دست آوریم. زیردنباله‌ای افزایشی است که هر عنصر آن از عنصر بعدی‌اش اکیداً کوچکتر باشد. این مسئله:

(مهندسی کامپیوتر ـ سراسری ۸۹)

۱) با استفاده از الگوریتم یافتن بزرگترین دنباله‌ی مشترک LCS (Longest common substring) می‌توان این مسئله را هم حل کرد.

۲) راه حل حریصانه و از مرتبه $O(n^2)$ دارد.

۳) راه حلی جز backtracking ندارد.

۴) راه حل حریصانه و از مرتبه $O(n\log n)$ دارد.

۱۹ـ جواب حاصل از رابطهٔ بازگشتی زیر فضای حالات کدام یک از مسائل را می‌توان بشمارد؟ (علوم کامپیوتر ـ سراسری ۸۹)

$$\begin{cases} T(1)=1 \\ T(n)=\sum_{i=1}^{n-1} T(i)T(n-i) \end{cases}$$

۱) تعداد ضرب زنجیره‌ای $n$ ماتریس

۲) تعداد پرانتزگذاری یک عبارت با $n$ عملگر

۳) تعداد درخت‌های باینری با $n-1$ گره

۴) تعداد درختان باینری با $n$ گره

۲۰ـ جهت محاسبه ضرب سه ماتریس $A_{k\times m}.B_{m\times n}.C_{n\times p}$ برای آنکه ترتیب $(A.B).C$ زمان کمتری نسبت به $A.(B.C)$ صرف نماید، کدام یک از گزینه‌های زیر می‌بایست برقرار باشد؟ (مهندسی کامپیوتر ـ آزاد ۸۹)

۱) $\frac{1}{m}+\frac{1}{p}<\frac{1}{k}+\frac{1}{n}$

۲) $\frac{1}{m}+\frac{1}{p}>\frac{1}{k}+\frac{1}{n}$

۳) $\frac{1}{m}+\frac{1}{p}=\frac{1}{k}+\frac{1}{n}$

۴) $\frac{1}{m}>\frac{1}{k}$

۲۱ـ برنامه زیر برای یافتن عدد فیبوناچی $n$ام تدوین شده است. مرتبه زمانی آن چیست؟ فرض کنید که ftab آرایه یک بعدی است که کلیه مقادیر آن در ابتدا 1- است. (مهندسی کامپیوتر ـ سراسری ۹۰)

```
int fib(int n)
{ if (ftab[n]>=0) return ftab[n];
  If (n==0||n==1) {ftab[n]=n;return n;}
  ftab[n]=fib(n-1)+fib(n-2);
  return ftab[n];
}
```

۱) $n$

۲) $2^n$

۳) $(1.75)^n$

۴) $(\frac{1+\sqrt{5}}{2})^n$

۲۲ـ اگر یک مسئله هم به روش «برنامه‌ریزی پویا» و هم به روش «تقسیم و غلبه» قابل حل باشد آنگاه کدام گزینه صحیح است؟

(علوم کامپیوتر ـ سراسری ۹۰)

۱) استفاده از روش «تقسیم و غلبه» بهتر است زیرا پیاده‌سازی آن آسان است.

۲) استفاده از روش «برنامه‌ریزی پویا» بهتر است زیرا حافظهٔ مصرفی آن کمتر است.

۳) روش «برنامه‌ریزی پویا» ممکن است نسبت به روش «تقسیم و غلبه» مسئله را در زمان کمتری حل کند.

۴) روش «تقسیم و غلبه» همواره نسبت به روش «برنامه‌ریزی پویا» مسئله را در زمان کمتری حل کند.

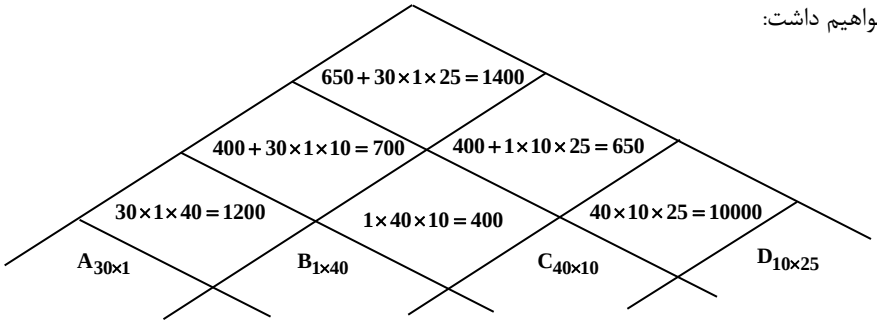
www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



## پاسخنامه تست‌های طبقه‌بندی شده کنکوری فصل نهم

**۱ـ گزینه «۲»** برای ضرب دو ماتریس $A_{m\times n}\times B_{n\times k}$ دقیقاً $m\times n\times k$ ضرب نیاز است و حاصل‌ضرب این دو ماتریس یک ماتریس $m\times k$ می‌باشد.
با استفاده از برنامه‌ریزی پویا خواهیم داشت:




مقدار ریشه برابر 1400 است که جواب مسأله خواهد بود.

**۲ـ گزینه «۱»** مرتبه این برنامه با توجه به حلقه استفاده شده خطی است. از طرفی این الگوریتم کاملاً ایدهٔ برنامه‌ریزی پویا را پشتیبانی می‌کند. (با ذخیره زیر جواب‌ها، آن‌ها را برای محاسبات بعدی بکار می‌گیرد.)

**۳ـ گزینه «۴»** بازه‌های $[3,20]$و $[2,4]$ را در نظر بگیرید. اگر برحسب $L_i$ها انتخاب انجام شود بنابراین، ابتدا بازه $[2,4]$ انتخاب می‌شود و با توجه به این که بازه $[3,20]$ با آن همپوشانی دارد، بازه $[3,20]$ حذف می‌گردد. تنها بازه انتخاب شده بازه $[2,4]$ می‌باشد که دارای بیشترین طول نیست. اگر همین دو بازه را در نظر بگیریم و برحسب $u_i$ها انتخاب انجام پذیرد، مجدداً بازه، $[2,4]$ انتخاب خواهد شد؛ بنابراین گزینه‌های اول و دوم نمی‌توانند درست باشند.
حال به عنوان یک مثال نقض برای گزینه سوم بازه‌های $[8,12],[4,9],[1,5]$ را در نظر بگیرید. طول این سه بازه به ترتیب برابر $5,6,5$ می‌باشد. بنابراین، اگر طبق روش بیان شده در گزینه سوم عمل شود بازه $[4,9]$ انتخاب می‌شود و دو بازه $[8,12]$و $[1,5]$ حذف می‌شوند، در صورتی که جواب بهینه حذف بازه $[4,9]$ و انتخاب بازه‌های $[1,5]$ و $[8,12]$ می‌باشد.

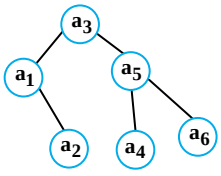
**۴ـ گزینه «۲»** با توجه به اینکه زمان ضرب دو ماتریس با ابعاد $a\times b$ و $b\times e$ برابر $a\times b\times c$ می‌باشد، داریم:

$$(N_1N_2)N_3 = wxy + wyz$$
$$N_1(N_2N_3) = xyz + wxz$$
$$wxy + wyz =< xyz + wxz \xrightarrow[xyzw]{\text{تقسیم بر}} \frac{1}{z}+\frac{1}{x}<\frac{1}{w}+\frac{1}{y}$$

**۵ـ گزینه «۲»** تعداد روش‌های ضرب $n$ ماتریس برابر عدد کاتالان $C_{n-1}$ می‌باشد که به وسیله رابطه بازگشتی زیر تعریف می‌شود:

$$T(n)=\sum_{i=1}^{n-1}T(i)T(n-i)$$

**۶ـ گزینه «۲»** شکل درخت جست‌وجوی دودویی بهینه به صورت زیر خواهد بود:




متوسط عمق برابر است با:

$$\frac{1}{7}\times 0+\frac{2}{7}\times(+\frac{1}{7}\times(+(\frac{1}{7}\times 2)\times 3=\frac{9}{7}$$

**۷ـ گزینه «۴»** این فرمول مربوط به رابطه بازگشتی اعداد کاتالان است که در آن استفاده از برنامه‌ریزی پویا ارجعت دارد. با توجه به مقدار ناهمگن و شرط پایانی، این جواب به مسأله پرانتزگذاری شباهت دارد.

**۸ـ گزینه «۴»** الگوریتم‌های طراحی شده توسط روش‌های (dynamic programming) و از نوع Memoized هر زیر مسئله ایجادشده را فقط یک بار حل می‌کنند؛ اما روش memoized یک روش بالا به پایین است و روش dynamic programming یک روش پایین به بالاست.

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۹ـ گزینه «۱»** تعداد ضرب‌های موردنیاز در سه گزینه به‌صورت زیر قابل محاسبه است.

گزینه ۱: $20\times50\times1+10\times20\times1+10\times1\times100=1000+200+1000=2200$

گزینه ۲: $10\times20\times50+10\times50\times1+10\times1\times100=10000+500+1000=11500$

گزینه ۳: $20\times50\times1+20\times1\times100+10\times20\times100=1000+2000+20000=23000$

---

**۱۰ـ گزینه «۳»** به ازای یک مقدار دلخواه و ثابت برای $\ell$، تعداد دفعات اجرای حلقه for داخلی (با شمارنده i) برابر $n-\ell+1$ می‌باشد، همچنین دستور زیر برای پیدا کردن مینیمم به ازای تمام مقادیر k که $i<k\le j$ دو عنصر $C[i,k]$ و $C[k+1,j]$ را از آرایه C می‌خواند:

$$C[i,j]=\min(C[i,k]+C[k+1,j]+d_{i-1}\times d_k\times d_j)$$

بنابراین به ازای یک مقدار ثابت برای $\ell$ تعداد خواندن‌ها در آرایه C عبارتست از:

$$2\times\underbrace{(n-\ell+1)}_{\text{تعداد دفعات تکرار حلقه for داخلی}}\times\underbrace{(\ell-1)}_{\text{متغیر k می‌تواند } \ell-1 \text{ مقدار مختلف را اختیار کند.}}$$

در نتیجه با توجه به این‌که حلقه for خارجی مقدار $\ell$ را بین 2 تا n تغییر می‌دهد، تعداد کل خواندن‌ها از آرایه C برابر است با:

$$\sum_{l=2}^{n}2\times(n-\ell+1)\times(\ell-1)$$

---

**۱۱ـ گزینه «۴»** بستگی به این دارد که عملکردها خاصیت جابجایی دارند یا خیر. اگر همه آنها این خاصیت را داشته باشند، پاسخ گزینه ۲ خواهد بود. ولی در سؤال به این موضوع اشاره‌ای نشده است.

---

**۱۲ـ گزینه «۲»** در الگوریتم LCS به منظور محاسبه C[i,j] از رابطه زیر استفاده می‌گردد:

$$C[i,j]=\begin{cases}C[i-1,j-1] & \text{if} \quad x_i=y_j\\ \max(C[i-1,j]\,,\,C[i,j-1]) & \text{otherwise}\end{cases}$$

همان گونه که مشخص است، اگر $x_i=y_j$ باشد، آنگاه نیاز به خواندن یک عنصر از آرایه C می‌باشد و اگر $x_i\neq y_j$ باشد، آنگاه نیاز به خواندن دو عنصر از آرایه می‌باشد (دقت نمایید که مطلب بیان شده برای سطر اول و ستون اول صادق نیست، بنابراین تست با فرض $i\neq0$ و $j\neq0$ بیان شده است.)

---

**۱۳ـ گزینه «۴»** در حالت $N_1\times(N_2\times N_3)$ تعداد ضربها برابر $(m\times n\times q)+(n\times p\times q)$ می‌باشد و در حالت $(N_1\times N_2)\times N_3$ تعداد ضربها برابر $(m\times n\times p)+(m\times p\times q)$ است و نتیجه از برابری زیر به دست می‌آید:

$$mnq+npq=mnp+mpq\Rightarrow nq(m+p)=mp(n+q)\Rightarrow\frac{m+p}{mp}=\frac{n+q}{nq}\Rightarrow\frac{1}{m}+\frac{1}{p}=\frac{1}{n}+\frac{1}{q}\Rightarrow\frac{1}{m}-\frac{1}{n}=\frac{1}{q}-\frac{1}{p}$$

---

**۱۴ـ گزینه «۲»** مسئله دارای زیر ساختار بهینه است و راه حل آن روش برنامه‌ریزی پویا می‌باشد.

---

**۱۵ـ گزینه «۱»** روش برنامه‌نویسی پویا یک الگوریتم پایین به بالا $(\text{Bottom}-\text{up})$ محسوب می‌گردد. سایر گزینه‌ها موارد درستی را در مورد روش برنامه‌نویسی پویا ارائه می‌کنند.

---

**۱۶ـ گزینه «۱»** کارایی روش برنامه‌نویسی پویا نسبت به روش بازگشتی در مسائلی که زیر مسئله‌های تکراری زیادی داشته باشند بهتر است. این کارایی به میزان اشتراک زیر مسئله‌ها بستگی دارد.

گزینه اول تابعی را معرفی می‌نماید که بیشترین تکرار زیر مسئله‌ها در آن نسبت به سایر گزینه‌ها وجود دارد.

در گزینه دوم زیر مسائل تکراری رخ نمی‌دهد و استفاده از روش برنامه‌نویسی پویا و روش تقسیم و حل، تفاوت چندانی نخواهد داشت.

در گزینه سوم تابع معرفی شده منجر به ایجاد زیر مسائل تکراری خواهد شد اما، تعداد زیر مسائل تکراری نسبت به گزینه اول بسیار کم‌تر خواهد بود.

در گزینه چهارم تابع معرفی شده به زیر مسائل تکراری منجر نمی‌گردد و تا حدی شبیه به الگوریتم جست و جوی دودویی خواهد بود.

---

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۱۷ـ هیچ‌کدام از گزینه‌ها صحیح نمی‌باشد.** از آنجایی که احتمال خرابی یال‌ها مستقل از یکدیگر است بنابراین، با توجه به قانون استقلال احتمال‌ها، احتمال این که مسیر $i$ به $k$ و مسیر $k$ به $j$ خراب باشد برابر حاصلضرب احتمال آنهاست پس، برای محاسبه $P_{ij}$ کافیست بین $P_{ij}$ و $P_{ik}\times P_{kj}$ مینیمم در نظر گرفته شود. حال احتمال خرابی مسیر $V_iV_kV_j$ برابر احتمال خرابی $V_iV_k$ یا $V_kV_j$ و یا هر دو می‌باشد بنابراین، این احتمال عبارت است از:

$$P_{ik}+P_{kj}-P_{ik}P_{kj}$$

و در نتیجه داریم:

$$P_{ij}=\min(P_{ij},P_{ik}+P_{kj}-P_{ik}P_{kj})$$

**۱۸ـ گزینه «۱»** اگر الگوریتم LCS را طوری تغییر دهیم که بزرگترین زیر رشته مشترک بین $X$ و $[1,2,3,...,n]$ را بدست آورد آنگاه مسئله موردنظر نیز حل شده است.

**۱۹ـ گزینه «۱و۳»** گزینه‌های ۱و۳ نشان دهنده عدد کاتالان $C_{n-1}$ می‌باشند.

**۲۰ـ گزینه «۱»** باید در هر کدام از حالات تعداد ضرب‌ها را محاسبه کرده و برای آنکه تعداد ضرب‌ها در حالت $(A.B).C$ زمان کمتری از $A.(B.C)$ صرف نماید، تعداد ضرب‌ها در حالت $(A.B).C$ کمتر باشد، داریم:

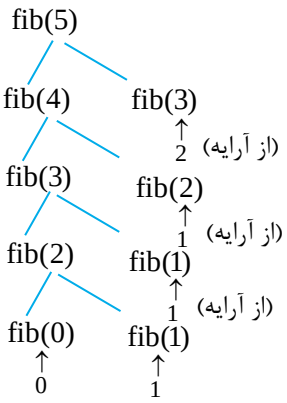
$$\begin{pmatrix} A\times B \text{ تعداد ضربها} = k\times m\times n & \text{ضرب} \\ (A.B)\times C \text{ تعداد ضربها} = k\times n\times p & \text{ضرب}\end{pmatrix},\quad \begin{pmatrix} B\times C \text{ تعداد ضربها} = m\times n\times p & \text{ضرب} \\ A\times (B.C) \text{ تعداد ضربها} = k\times m\times p & \text{ضرب}\end{pmatrix}$$

طرفین را بر $kmnp$ تقسیم می‌کنیم داریم:

$$\Rightarrow k\times m\times n+k\times n\times p<m\times n\times p+k\times m\times p$$

$$\Rightarrow \frac{kmn+knp}{kmnp}<\frac{mnp+kmp}{kmnp}\Rightarrow \frac{1}{p}+\frac{1}{m}<\frac{1}{k}+\frac{1}{n}$$

**۲۱ـ گزینه «۱»** الگوریتم داده شده از یک جدول پویا (Dynamic table) استفاده می‌نماید (آرایه ftab) و مانند نسخه بازگشتی این الگوریتم همه مؤلفه‌ها را مستقل محاسبه و فراخوانی نمی‌کند، بلکه از مقادیر محاسبه شده (با ذخیره‌سازی آنها در آرایه ftab) مجدداً استفاده می‌کند. به عنوان مثال نحوه محاسبه (۵) fib به‌صورت زیر است.




و بنابراین زمان $O(n)$ خواهد بود.

**۲۲ـ گزینه «۳»** در حالت کلی ممکن است هیچ‌یک از گزینه‌ها صحیح نباشند، اما گزینه سوم محتمل‌تر از بقیه می‌باشد.

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



# فصل دهم

## «الگوریتم‌های حریصانه (Greedy Algorithms)»

### تست‌های طبقه‌بندی شده کنکوری فصل دهم

۱ـ در فشرده‌سازی با استفاده از کد هافمن اگر **x** و **y** دو کاراکتری باشند که کمترین تکرار را داشته باشند در آن صورت کد بهینه برای این دو نویسه **(character)** ..... . (علوم کامپیوتر ـ سراسری ۸۰)

۱) دارای پیشوند یکسان می‌باشند فقط در آخرین لیست کد آن‌ها متفاوت است.

۲) دارای پسوند یکسان می‌باشند فقط در اولین لیست کد متفاوت است.

۳) دارای پیشوند یکسان نمی‌باشند و فقط کد آخر آن‌ها برابر است.

۴) دارای پسوند یکسان نمی‌باشند و فقط کد اول آن‌ها با هم برابر است.

۲ـ رشته متنی **(a b a a b a c a d c a d e)** را در نظر بگیرید. کد هافمن حاصل برای هر یک از نویسه‌ها برابر است با ..... (علوم کامپیوتر ـ سراسری ۸۰)

| ۱) | ۲) | ۳) | ۴) |
|---|---|---|---|
| $a=1$ | $a=0$ | $a=000$ | $a=00$ |
| $b=01$ | $b=100$ | $b=001$ | $b=01$ |
| $c=001$ | $c=101$ | $c=010$ | $c=10$ |
| $d=0001$ | $d=110$ | $d=011$ | $d=11$ |
| $e=0000$ | $e=111$ | $e=100$ | $e=100$ |

۳ـ متنی شامل **7000** حرف از حروف **a,b,c,d,e,f** با دفعات تکرار **a = 1000**، **b = 1200**، **c = 800**، **d = 1500**، **e = 1800** و **f = 700** موجود است. چنانچه کدی بهینه برای حروف بالا انتخاب نماییم، تعداد کل بیتهای لازم برای تبدیل متن مذکور به مجموعه‌ای از بیت‌ها چقدر است؟ (مهندسی کامپیوتر ـ سراسری ۸۱)

۱) 14600 ۲) 17700 ۳) 24300 ۴) 35200

۴ـ حداکثر طول یک کد برای **n** عنصر که به روش هافمن کدگذاری می‌شوند چقدر است؟ (مهندسی کامپیوتر ـ سراسری ۸۲)

۱) $n-2$ ۲) $\lceil \lg n \rceil$ ۳) $n-1$ ۴) $\left\lceil \frac{n}{2} \right\rceil$

۵ـ جمله زیر را اگر به روش هافمن فشرده‌سازی کنیم، اندازه‌ی فشرده شده آن چند بیت خواهد بود؟ ( ∪ فاصله خالی است.) (علوم کامپیوتر ـ سراسری ۸۲)

**this ∪ is ∪ the ∪ set ∪ that ∪ has ∪ set**

۱) 60 ۲) 68 ۳) 76 ۴) 92

۶ـ **n** دانشجو به صورت تصادفی در یک ردیف نشسته‌اند. فرض کنید دانشجوی **i** در مختصات $(x_i, 0)$ قرار دارد. مربی می‌خواهد در نقطه‌ای بایستد که مجموع فاصله‌اش با همهٔ دانشجویان کمینه شود. با داشتن همهٔ **n** مختصات، محل قرار گرفتن مربی را در چه زمانی می‌توان به دست آورد؟ (مهندسی کامپیوتر ـ سراسری ۸۳)

۱) $O(n)$ ۲) $O(n^2)$ ۳) $O(n \log n)$ ۴) $\log n$

۷ـ رویه زیر را در نظر بگیرید:

```
Function power(x,k)
    y ← x
    i ← k
    a ← 1
    while i > 0
        do if i is odd
              then a ← a × y
           y ← y × y
           i ← ⌊i/2⌋
    return a
```

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



این الگوریتم قرار است مقدار $x^k$ را محاسبه کند. مقدار مستقل از حلقه **(loop invariant)** این الگوریتم چیست؟ (یعنی چه عبارتی همواره در ابتدای حلقه درست است؟) (مهندسی کامپیوتر ـ سراسری ۸۳)

۱) $y^i a = x^k$ ۲) $y^i a^i = x^k$ ۳) $a(ay)^i = x^k$ ۴) $y(ay)^i = x^k$

۸ـ تابع زیر را در نظر بگیرید، فرض کنیم $T(n)$ تعداد سطرهایی است که در اثر اجرای این تابع روی صفحه، نمایش داده خواهند شد و داریم $n \le 1$ و $T(n) = 1$. کدام یک از تعاریف زیر صحیح است؟ (مهندسی کامپیوتر ـ سراسری ۸۳)

```
function Concow (n)
    if n ≤ 1 then
        writeln('Stop')
        return 1
    else
        for i ← 1 to n do
            writeln(concow(n div 2))
        return n
```

۱) $T(n) = \sum_{i=1}^{n} T(\lfloor \frac{1}{2} \rfloor)$

۲) $T(n) = nT(\lfloor \frac{n}{2} \rfloor) + n$

۳) $T(n) = nT(\lfloor \frac{n}{2} \rfloor)$

۴) $T(n) = T(\lfloor \frac{n}{2} \rfloor) + n$

۹ـ $n$ کار با زمان‌های پردازش $p_1 < p_2 < \ldots < p_n$ قرار است در $m$ پردازنده کاملاً یکسان اجرا شوند، به این صورت که کارهایی را که هر کدام از پردازنده‌ها قرار است به ترتیب اجرا کند از قبل مشخص می‌کنیم. بدیهی است که زمان اتمام کارهای یک پردازنده جمع زمان‌های پردازش کارهایی است که به آن پردازنده تخصیص داده شده است. برای تخصیص، کارها را به ترتیب شماره‌شان (از زمان کم تا زیاد) انتخاب می‌کنیم و هر کار را به پردازنده‌ای تخصیص می‌دهیم که همهٔ کارهای تخصیص داده شده به آن نسبت به بقیهٔ پردازنده‌ها زودتر تمام شوند. حداکثر زمان اتمام همهٔ کارها چقدر است؟ (فرض کنید $A = \sum_{i=1}^{n} p_i$) (مهندسی کامپیوتر ـ سراسری ۸۴)

۱) $\frac{A}{m}$ ۲) $\frac{A}{m} - p_n$ ۳) $\frac{A}{m} + p_n$ ۴) $\frac{A - p_n}{m} + p_n$

۱۰ـ برای اجرای $n$ برنامه آماده به اجرا روی یک کامپیوتر مسئله بهینه‌سازی به شکل زیر تعریف شده است. اجرای هر برنامه یک واحد زمانی به طول می‌انجامد. حداکثر زمان انتظار برای شروع به اجرای هر برنامه $D_i$ واحد زمانی می‌باشد و در صورت انتظار بیش از این مقدار، اجرای برنامه دیگر سودمند نخواهد بود. در اثر اجرای هر برنامه سود $P_i$ بدست می‌آید. بدیهی است که با توجه به زمان‌های انتظار اجرای همه برنامه‌ها میسر نخواهد بود. اما هدف آن است که برنامه‌ها به گونه‌ای اجرا شوند که بیشترین سود بدست آید. با توجه به شرح فوق اگر برای حل مسئله از روش حریصانه **(Greedy)** استفاده شود، درجه پیچیدگی زمانی الگوریتم چیست؟ (مهندسی IT ـ سراسری ۸۵)

۱) $\theta(n^2)$ ۲) $\theta(n)$ ۳) $\theta(n^3)$ ۴) $\theta(n \log n)$

۱۱ـ در مسئله **knapsack 0 − 1** با 5 شیء به شرح روبرو ($P_i$ سود و $W_i$ وزن شی $i$ام) ظرفیت کیسه برابر **13kg** می‌باشد. مقدار سود ماکزیمم چیست؟ (مهندسی IT ـ سراسری ۸۵)

| i | 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|---|
| $P_i$ | $35 | $30 | $20 | $12 | $3 |
| $W_i$ | 7 | 5 | 2 | 3 | 1 |

۱) $65
۲) $68
۳) $70
۴) $80

۱۲ـ اگر قرار باشد 6 سخنرانی زیر با شروع و پایان مشخص را طوری برنامه‌ریزی کنیم که بیشترین تعداد سخنرانی در یک سالن قابل ارائه باشد، بیشترین تعداد سخنرانی‌های ممکن کدام است؟ (مهندسی IT ـ سراسری ۸۵)

| شماره سخنرانی | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|
| شروع | 2 | 1 | 3 | 3 | 4 | 6 |
| پایان | 9 | 2 | 4 | 5 | 7 | 8 |

۱) 2
۲) 5
۳) 4
۴) 3

۱۳ـ اگر رشته **abcabbaccaabdffe** را با روش کدینگ هافمن کد کنیم. طول کد چند بیت خواهد شد؟ (مهندسی IT ـ سراسری ۸۵)

۱) 38 ۲) 36 ۳) 34 ۴) هیچکدام

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



۱۴ـ مجموعه الفبای $\{a,b,c,d,e,f\}$ را در نظر بگیرید و فرض کنید فرکانس رخدادهای این حروف در یک فایل داده‌ای به ترتیب برابر $\{0.01,0.25,0.35,0.2,0.03,0.16\}$ باشد. اگر در این حالت درخت کدگذاری بهینه هافمن تشکیل شود، کدام یک از زوج حروف ذیل در یک سطح درخت قرار ندارند؟ (مهندسی IT ـ سراسری ۸۵)

۱) $d,f$ ۲) $b,d$ ۳) $a,e$ ۴) $b,c$

۱۵ـ می‌خواهیم قطر یک درخت آزاد $G=(V,E)$ (گراف بدون جهت و غیرسیکلی و بدون وزن ) را پیدا کنیم. قطر بیشترین فاصله دو گره در گراف است. این کار در بهترین حالت با چه مرتبه‌ای می‌تواند انجام شود؟ (مهندسی کامپیوتر ـ سراسری ۸۶)

۱) $O(|V||E|)$ ۲) $O(|V|)$ ۳) $O(|V|\lg|V|)$ ۴) $O(|V|^3)$

۱۶ـ در متنی $n$ کاراکتر وجود دارد که فراوانی آنها از تصاعد هندسی با جمله اول $a$ و قدر نسبت $q=2$ تبعیت می‌کند . در مورد هزینه کد پیشوندی هافمن کدام گزینه صحیح است؟ (مهندسی کامپیوتر ـ سراسری ۸۶)

۱) $a(2^{n+1}-n-3)$ ۲) $a(2^{n+1}+n-3)$ ۳) $a(2^{n+1}-2n-2)$ ۴) $a(2^{n+1}-n+3)$

۱۷ـ کد هافمن عبارت AABAABAACAABAACACABA چند بیت است؟ (مهندسی IT ـ سراسری ۸۸)

۱) 23 ۲) 24 ۳) 27 ۴) 30

۱۸ـ در الگوریتم فشرده‌سازی هافمن، اگر برای یافتن دو نویسه با کمترین فراوانی از جست‌وجوی خطی به جای هرم (heap) استفاده شود، زمان اجرای آن چه خواهد بود؟ ( $n$ تعداد نویسه‌هایی است که قرار است کدگذاری شوند.) (مهندسی کامپیوتر ـ سراسری ۹۰)

۱) $\Theta(n)$ ۲) $\Theta(n^2)$ ۳) $\Theta(n\lg n)$ ۴) $\Theta(n^2\lg n)$

۱۹ـ $n$ بازه‌ی $I_i=[x_i,y_i)$ که $1\le i\le n$ داده شده‌اند. هر بازه نشان‌دهنده‌ی زمان یک کلاس درس است و هر کلاس به صورت مستقل احتیاج به یک اتاق دارد. می‌خواهیم کمترین تعداد اتاق‌های لازم را پیدا کنیم تا بتوان بدون تداخل زمانی، به همه‌ی کلاس‌ها اتاق تخصیص داد. سریع‌ترین الگوریتم برای یافتن این تعداد اتاق کمینه، از چه مرتبه‌ی زمانی است؟ (مهندسی کامپیوتر ـ سراسری ۹۰)

۱) $O(n^2)$ ۲) $O(n^3)$ ۳) $O(n\lg n)$ ۴) راه‌حل چند جمله‌ای ندارد.

۲۰ـ اگر در روش کدگذاری هافمن، فراوانی دادهٔ اول از مجموع فراوانی $n-1$ داده دیگر بیشتر بوده و فراوانی $n-1$ عنصر ذکر شده با هم برابر باشد آنگاه طول کدهای تولید شده حداکثر چقدر است؟ (علوم کامپیوتر ـ سراسری ۹۰)

۱) $\lfloor\log n\rfloor+1$ ۲) $\lceil\log n\rceil+1$ ۳) $\lceil\log(n-1)\rceil+1$ ۴) $\lfloor\log(n-1)\rfloor+1$

۲۱ـ ارتفاع درخت هافمن اگر ورودی 10 نشانه با بسامدهای ۱ تا 10 باشد، چه قدر است؟ (مهندسی کامپیوتر، هوش مصنوعی ـ سراسری ۹۴)

۱) 3 ۲) 4 ۳) 5 ۴) 6

www.modaresanesharif.ac.ir

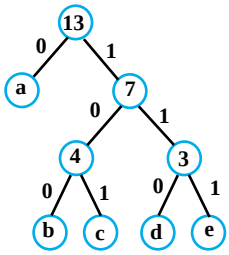
www.modaresanesharif.ac.ir



## پاسخنامه تست‌های طبقه‌بندی شده کنکوری فصل دهم

**۱ـ گزینه «۱»** در این صورت آن‌ها حتماً پیشوند یکسان دارند و تنها در بیت آخر دارای کد متفاوت هستند.

**۲ـ گزینه «۲»** ابتدا تکرار هر حرف را مشخص می‌کنیم و سپس درخت موردنظر را ترسیم می‌نماییم:

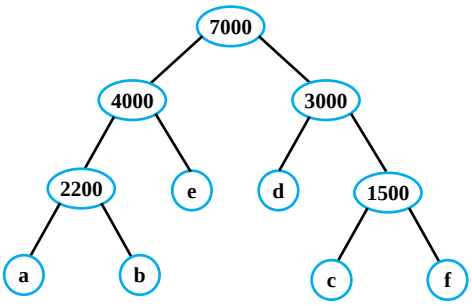



| a | b | c | d | e |
|---|---|---|---|---|
| 6 | 2 | 2 | 2 | 1 |

$a:0$
$b:100$
$c:101$
$d:110$
$e:111$

بنابراین پاسخ درست گزینه ۲ می‌باشد.

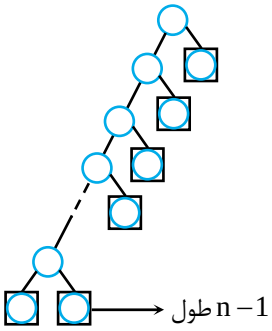
**۳ـ گزینه «۲»** در حقیقت این سوال همان مسئله Huffman Code می‌باشد و درخت آن به صورت زیر خواهد بود.




با توجه به درخت کد هافمن روبرو مشخص است که $d$ و $e$ دارای کدی به طول 2 می‌باشند و $c$، $f$، $a$ و $b$ دارای کدی به طول 3 هستند بنابراین، طول کل کد عبارت است از:

$$(1000+800+700+1200)\times 3+(1500+1800)\times 2=17700$$

**۴ـ گزینه «۳»** درخت کد هافمن می‌تواند حداکثر دارای عمق $n-1$ باشد.




**۵ـ گزینه «۳»** فراوانی هر یک از کاراکترها به صورت زیر می‌باشد:

$N(t)=6$　$N(h)=4$　$N(i)=2$　$N(s)=5$　$N(e)=3$　$N(a)=2$　$N(u)=6$

بنابراین تعداد کل بیت‌ها برابر است با:

$$\sum N(x)\text{depth}(x)=12+9+16+12+15+12=76$$

**۶ـ گزینه «۱»** در حقیقت مربی باید در محل میانگین بایستد تا مجموع فاصله‌اش کمترین باشد (اگر فاصله‌ها را به صورت داده‌های آماری فرض کنیم) و با توجه به این که میانگین را می‌توان در زمان $O(n)$ به دست آورد گزینه ۱ درست است.

**۷ـ گزینه «۱»** در هر بار اجرای حلقه مقدار $y$ برابر عددی است که اگر $i$ بار در خود ضرب شود (یعنی $y^i$) و در نهایت در $a$ ضرب شود همان $x^k$ را ایجاد می‌نماید؛ بنابراین: $y^i\times a=x^k$

به عنوان مثال اگر $k=2$ باشد، آنگاه در ابتدای اولین اجرای حلقه داریم:

$$y=x,a=1,i=2$$

بنابراین: $y^i\times a=y^2\times a=x^2=x^k$

حال اگر $k=3$ باشد، آنگاه پس از اجرای اول حلقه و در ابتدای حلقه برای بار دوم داریم:

$$y=x^2,a=x,i=1$$

$$y^i\times a=x^3$$

**۸ـ گزینه «۲»** به غیر از $nT(\frac{n}{2})$، $n$ عدد write هم داریم.

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۹ـ گزینه «۴»** کافیست دو حالت زیر را در نظر بگیرید.

۱) $p_1=1, p_2=2, p_3=3, m=2$ در این حالت زمان لازم برابر 4 است بنابراین، گزینه‌های ۱ و۲ غلط می‌باشند.

۲) $p_1=1, p_2=2, p_3=3, p_4=4, m=2$ در این حالت نیز مشخص می‌شود که گزینهٔ سوم غلط است.

**۱۰ـ گزینه «۱»** مسئله طرح شده مسئله زمان‌بندی با مهلت معین است که در آن کارها بر اساس سود مرتب می‌شوند و حداکثر کارهای قابل اجرا انتخاب می‌گردد، بنابراین زمان حل مسئله $n^2$ خواهد بود.

**۱۱ـ گزینه «۳»** اگر به غیر از قطعه دوم تمام قطعات دیگر را انتخاب کنیم، آنگاه سود کل برابر $70 خواهد بود. اما با انتخاب هر ترکیب دیگری سود کمتر از 70 حاصل می‌گردد یا وزن قطعات از ظرفیت کوله‌پشتی بیشتر خواهد شد.

**۱۲ـ گزینه «۴»** مسئله داده شده در حقیقت همان مسئله زمان‌بندی است و بنابراین کافیست کارها بر اساس زمان خاتمه مرتب شوند و در نهایت تعداد کل کارهای ممکن از ابتدای لیست انتخاب گردند:

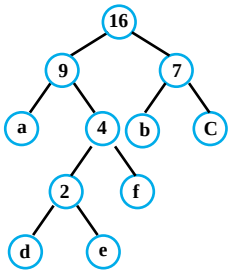
| شمارهٔ کار | 1 | 2 | 3 | 4 | 5 | 6 |
|---|---|---|---|---|---|---|
| زمان شروع | 2 | 1 | 3 | 3 | 4 | 6 |
| زمان پایان | 9 | 8 | 7 | 5 | 4 | 2 |

بنابراین می‌توان کارهای 2، 3 و 5 را انجام داد.

**۱۳ـ گزینه «۱»** در ابتدا فراوانی کاراکترهای موجود در متن محاسبه می‌گردد:

| کاراکتر | فراوانی |
|---|---|
| a | 5 |
| b | 4 |
| c | 3 |
| d | 1 |
| e | 1 |
| f | 2 |

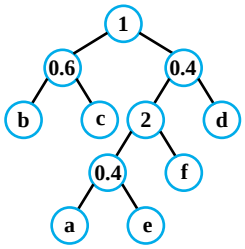
سپس درخت متناظر را رسم می‌نماییم:




همان‌گونه که مشاهده می‌گردد تعداد بیت‌های لازم برای کاراکترها به صورت روبه‌رو است:

$a:2, b:2, C:2, f:3, d:4, e:4$ بنابراین، برای کل متن داده شده به 38 بیت نیاز است.

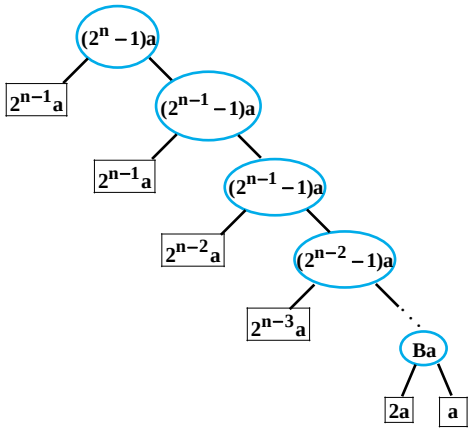
**۱۴ـ گزینه «۱»** درخت متناظر با داده‌های ذکر شده به صورت زیر خواهد بود:




همان گونه که ملاحظه می‌شود d و f در یک سطح قرار ندارند.

**۱۵ـ گزینه «۲»** محاسبه قطر گراف در زمان $O(|V|)$ امکان‌پذیر است. با الگوریتم DFS (پیمایش عمقی) که مرتبه زمانی آن $O(|V|)$ است، می‌توان این مسئله را حل کرد، دو گره‌ای که در پیمایش DFS بیشترین فاصله را داشته باشند قطر درخت را تشکیل می‌دهند.

**۱۶ـ گزینه «۱»** با توجه به فراوانی کاراکترها شکل کلی درخت هافمن به صورت زیر خواهد بود:




بنابراین هزینه کل کد عبارتست از:

$$2^{n-1}a+2^{n-2}a\times 2+\cdots+2a(n-1)+a\times(n-1)=a(2^{n+1}-n-3)$$

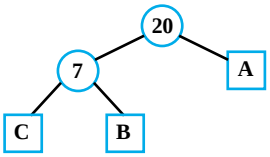
www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۱۷ـ گزینه «۳»** ابتدا فراوانی هر کاراکتر را در متن داده شده بدست می‌آوریم:

A : 13　　B : 4　　C : 3




بنابراین درخت ایجاد شده در کد هافمن به صورت مقابل خواهد بود.

بنابراین تعداد بیت‌های لازم برابر است با: $13\times1+4\times2+3\times2=27\,\text{bit}$.

---

**۱۸ـ گزینه «۲»** در الگوریتم اصلی هافمن n بار عمل EXTAC-MIN انجام می‌شود که این عمل برروی یک binary-heap انجام می‌پذیرد و زمان آن $O(\log_2^n)$ می‌باشد، بنابراین اگر به جای heap از جست‌وجوی خطی استفاده شود زمان $O(n\times n)=O(n^2)$ خواهد بود.

---

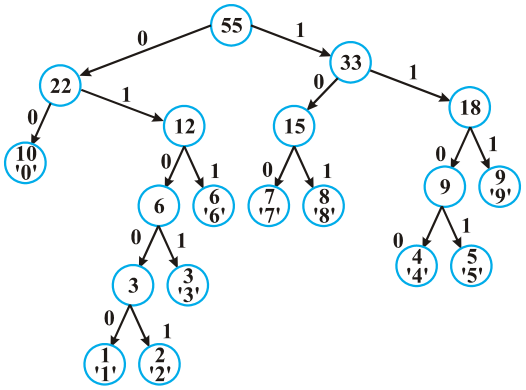
**۱۹ـ گزینه «۳»** در کل به یک مرتب‌سازی نیاز داریم که زمان آن $O(n\log n)$ می‌باشد.

---

**۲۰ـ گزینه «۳»** در اینجا $n-1$ عنصر دیگر یک درخت دودویی به عمق $\text{Log}(n-1)$ تشکیل می‌دهند که در آخر با گره مذکور عمق درخت یا طول کدها برابر $1+\lceil \text{Log}\,n-1\rceil$ می‌شود.

---

**۲۱ـ گزینه «۳»** درخت هافمن مورد نظر به صورت زیر خواهد بود:




دقت کنید که فراوانی‌ها به صورت زیر در نظر گرفته شده است.

| کاراکتر | 1 | 2 | 3 | ⋯ | 9 | 0 |
|---|---|---|---|---|---|---|
| فراوانی | 1 | 2 | 3 | | 9 | 10 |

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



# بخش چهارم: الگوریتم‌های گراف

## فصل دوازدهم

## «الگوریتم‌های پیمایش گراف»

### تست‌های طبقه‌بندی شده کنکوری فصل دوازدهم

**۱ـ کدام عبارت صحیح نیست؟** (مهندسی کامپیوتر ـ آزاد ۷۷)

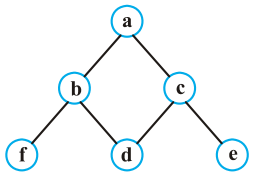
۱) درختی که تعدادی از لبه‌ها و تعدادی از رئوس را در بر دارد درخت پوشا گفته می‌شود.

۲) الگوریتم DFS لبه‌های T را به شکل یک درخت پوشا تعیین می‌کند.

۳) الگوریتم BFS لبه‌های T را به شکل یک درخت پوشا تعیین می‌کند.

۴) همه گراف‌های متصل با n-1 لبه درخت هستند.

**۲ـ کدام گزینه نمی‌تواند خروجی پیمایش اول عمق (dfs) گراف زیر باشد؟** (علوم کامپیوتر ـ سراسری ۸۰)



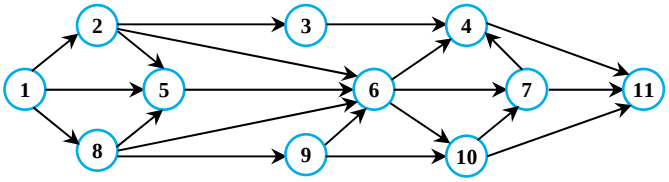

۱) a b d c e f

۲) a b d c f e

۳) a c d b f e

۴) a b f d c e

**۳ـ در جست‌وجوی عمق اول (Depth-First search) گراف جهت‌دار زیر، فرض کنید که گره 1، گره شروع باشد و گره‌های مجاور یک گره به ترتیب مقدار عددی‌شان ملاقات شوند. کدام گزینه (از چپ به راست) ترتیب گره‌های ملاقات شده را نشان می‌دهد؟** (مهندسی کامپیوتر ـ سراسری ۸۱)




۱) 1,2,3,4,11,5,6,7,10,8,9

۲) 1,2,8,5,3,9,6,4,11,10,7

۳) 1,2,5,8,3,9,6,4,10,7,11

۴) 1,2,3,4,5,6,7,8,9,10,11

**۴ـ به دست آوردن تعداد سیکل‌های ساده در یک گراف مسطح با الگوریتمی با درجه ..... قابل محاسبه می‌باشد.** (مهندسی کامپیوتر ـ سراسری ۸۱)

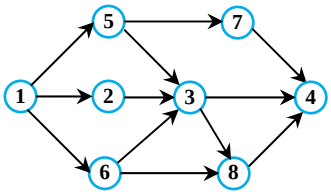
۱) 1

۲) n

۳) $n^2$

۴) $2^n$

**۵ـ اگر رأس‌های گراف زیر را به صورت topological مرتب کنیم، چند دنباله متفاوت به دست می‌آید؟** (مهندسی کامپیوتر ـ سراسری ۸۲)



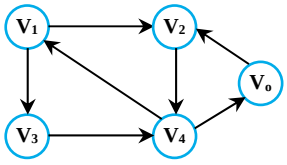

۱) 6

۲) 1

۳) 2

۴) 48

**۶ـ اجزای قویاً هم بند (Strongly connected) در گراف زیر کدام است؟** (مهندسی کامپیوتر ـ سراسری ۸۲)




۱) $\{v_0, v_1, v_2, v_3, v_4\}$

۲) $\{v_0, v_2, v_3, v_4\}, \{v_0\}$

۳) $\{v_0, v_2, v_4\}, \{v_1, v_3, v_4\}$

۴) $\{v_1, v_3, v_4\}, \{v_2, v_1, v_4\}, \{v_0, v_2, v_4\}$

**۷ـ وجود دور در یک گراف بدون جهت را در بهترین حالت با چه مرتبه‌ای، می‌توان به دست آورد؟** (علوم کامپیوتر ـ سراسری ۸۲)

۱) $O(|E|+|V|)$

۲) $O(|V|)$

۳) $O(|V||E|)$

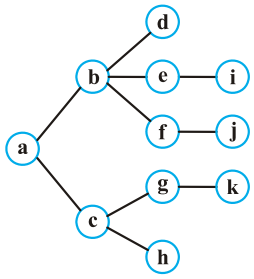
۴) $O(|E|\log|V|)$

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۸ـ با اجرای یک پیمایش DFS بر روی یک گراف بدون جهت همبند مثل G درخت DFS به صورت شکل زیر به دست آمده است، با فرض این که a رأس مبدأ پیمایش است و دانستن این که درجه a برابر 4 است، کدام گزاره درست است؟** (علوم کامپیوتر ـ سراسری ۸۳)




۱) $G-a$ سه مؤلفه همبندی دارد.

۲) G دارای دور نمی‌باشد.

۳) $G-a$ همبند است.

۴) $G-b$ نمی‌تواند همبند باشد.

**۹ـ فرض کنید یک نوع الگوریتم پیمایش گراف بر روی دو گراف ساده همبند $G_1$ و $G_2$ اجرا شده است و $T_1=a_1.....a_n$ و $T_2=b_1.....b_n$ نشان‌دهنده دنباله رأس‌های ملاقات شده به ترتیب در $G_1$ و $G_2$ هستند. کدام گزاره نادرست است؟** (علوم کامپیوتر ـ سراسری ۸۳)

۱) اگر $T_1$ و $T_2$ از پیمایش BFS به دست آمده باشند، آنگاه $T=T_1T_2$ نشان‌دهنده یک پیمایش BFS در گرافی است که از اتصال رأس‌های $a_n$ و $b_1$ به دست آمده است.

۲) اگر $T_1$ و $T_2$ از پیمایش BFS به دست آمده باشند، آنگاه $T=T_1T_2$ نشان‌دهنده یک پیمایش BFS در گرافی است که از اتصال رأس‌های $a_1$ و $b_1$ به دست آمده است.

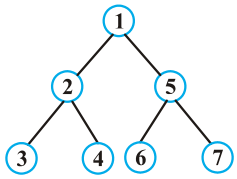
۳) اگر $T_1$ و $T_2$ از پیمایش DFS به دست آمده باشند، آنگاه $T=T_1T_2$ نشان‌دهنده یک پیمایش DFS در گرافی است که از اتصال رأس‌های $a_1$ و $b_1$ به دست آمده است.

۴) اگر $T_1$ و $T_2$ از پیمایش DFS به دست آمده باشند، آنگاه $T=T_1T_2$ نشان‌دهنده یک پیمایش DFS در گرافی است که از اتصال رأس‌های $a_n$ و $b_1$ به دست آمده است.

**۱۰ـ اگر دنبالهٔ درجه یک گراف بدون جهت 3, 3, 3, 3, 2, 2 باشد، در آن صورت تعداد یال‌های پشتی (back edge) این گراف در جستجوی DFS برابر است با:** (علوم کامپیوتر ـ سراسری ۸۴)

۱) 2 ۲) 3 ۳) 4 ۴) 5

**۱۱ـ اگر در پیمایش DFS در گراف غیر جهت‌دار $G=(V,E)$ درخت فراگیر آن برابر با درخت زیر باشد، در آن صورت کدام گزینه صحیح است؟ در گراف $G=(V,E)$:** (علوم کامپیوتر ـ سراسری ۸۴)



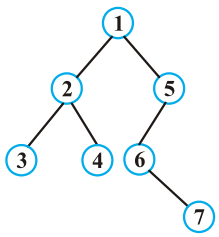

۱) بین دو رأس 5 و 2 می‌تواند یال وجود داشته باشد.

۲) بین دو رأس 6 و 2 و دو رأس 7 و 6 می‌تواند یال وجود داشته باشد.

۳) بین دو رأس 1 و 4 می‌تواند یال وجود داشته باشد.

۴) بین دو رأس 7 و 6 می‌تواند یال وجود داشته باشد.

**۱۲ـ اگر در پیمایش BFS یک گراف غیر جهت‌دار $G=(V,E)$ درخت فراگیر حاصل برابر با درخت زیر باشد، در آن صورت در گراف $G=(V,E)$ کدام گزینه صحیح است؟** (علوم کامپیوتر ـ سراسری ۸۴)



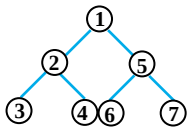

۱) بین دو رأس 5 و 4 می‌تواند یال وجود داشته باشد.

۲) بین دو رأس 5 و 1 می‌تواند یال وجود داشته باشد.

۳) بین دو رأس 7 و 3 و دو رأس 4 و 1 می‌تواند یال وجود داشته باشد.

۴) بین دو رأس 7 و 1 و دو رأس 5 و 1 می‌تواند یال وجود داشته باشد.

**۱۳ـ اگر در پیمایش DFS در گراف غیر جهت‌دار $G=(V,E)$ درخت فراگیر آن برابر با درخت زیر باشد، در آن صورت کدام گزینه صحیح است؟در گراف $G=(V,E)$:** (علوم کامپیوتر ـ سراسری ۸۴)




۱) بین دو رأس 5 و 2 می‌تواند یال وجود داشته باشد.

۲) بین دو رأس 6 و 2 و دو رأس 7 و 6 می‌تواند یال وجود داشته باشد.

۳) بین دو رأس 1 و 4 می‌تواند یال وجود داشته باشد.

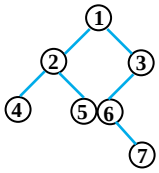
۴) بین دو رأس 7 و 6 می‌تواند یال وجود داشته باشد.

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۱۴ـ اگر در پیمایش BFS یک گراف غیر جهت‌دار $G=(V,E)$ درخت فراگیر حاصل برابر با درخت زیر باشد، در آن صورت در گراف $G=(V,E)$ کدام گزینه صحیح است؟** (علوم کامپیوتر ـ سراسری ۸۴)




۱) بین دو رأس 5 و 4 می‌تواند یال وجود داشته باشد.
۲) بین دو رأس 5 و 1 می‌تواند یال وجود داشته باشد.
۳) بین دو رأس 7 و 3 و دو رأس 4 و 1 می‌تواند یال وجود داشته باشد.
۴) بین دو رأس 7 و 1 و دو رأس 5 و 1 می‌تواند یال وجود داشته باشد.

**۱۵ـ در یک گراف جهت‌دار که در هر گره یک عدد صحیح قرار دارد یک یال جهت‌دار رابطه کوچکتری را بیان می‌کند. تعداد سیکل‌های ساده گرافی با مشخصات بالا با n گره برابر است با:** (مهندسی کامپیوتر ـ سراسری ۸۵)

۱) $2n-n+1$
۲) $n^2-3n+2$
۳) $n^2-\frac{3}{2}n+1$
۴) $\frac{n^2-3n+2}{2}$

**۱۶ـ کدام‌یک از گزینه‌های زیر غلط است؟** (مهندسی کامپیوتر ـ سراسری ۸۵)

۱) وجود یک یال cross در پیمایش عمق اول معرف عدم وجود سیکل هامیلتونی است.
۲) وجود یک یال cross در پیمایش عمق اول نشانگر وجود گره مفصلی در گراف است.
۳) وجود یک یال cross در پیمایش عمق اول را نمی‌توان دلالت بر عدم وجود مسیر هامیلتونی دانست.
۴) وجود یک یال cross در پیمایش عمق اول معرف عدم وجود مسیر اویلری در گراف است.

**۱۷ـ در یک گراف کامل بی‌جهت با n رأس تعداد یال‌های visit نشده در پیمایش DFS برابر است با:** (علوم کامپیوتر ـ سراسری ۸۵)

۱) $n-1$
۲) $\frac{1}{2}(n-1)(n-2)$
۳) $\frac{1}{2}n(n-1)$
۴) $\frac{1}{2}n(n-3)$

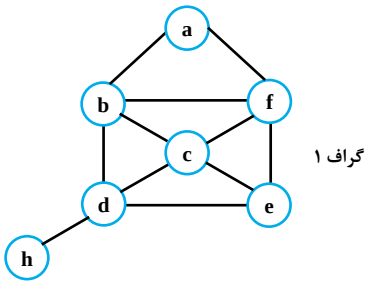
**۱۸ـ فرض کنید G یک گراف غیر جهت‌دار بدون وزن باشد. کدام یک از روش‌های جست‌وجوی زیر در گراف G، گره‌های G را به صورتی ملاقات می‌کند که گره‌هایی که فاصله آن‌ها از گره شروع کمتر است زودتر ملاقات شوند؟** (علوم کامپیوتر ـ سراسری ۸۵)

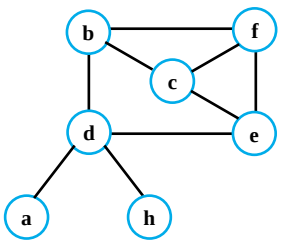
۱) روش BFS
۲) روش DFS
۳) روش‌های BFS و DFS
۴) با پیمایش‌های BFS و BFS نمی‌توان نتیجه‌ای به دست آورد.

**۱۹ـ فرض کنید G یک گراف جهت‌دار باشد. کدام یک از شرایط زیر کافی است تا تضمین کند در جستجوی DFS، برای هر دو رأس V و W در G اگر یک مسیر از V به W وجود داشته باشد DFS (W) سریع‌تر از DFS (V) پایان می‌یابد؟** (علوم کامپیوتر ـ سراسری ۸۵)

۱) همبند نباشد.
۲) همبند باشد.
۳) G دارای یک Cycle جهت‌دار باشد.
۴) G دارای هیچ Cycle جهت‌دار نباشد.

**۲۰ـ دو گراف زیر و دو پیمایش BFS,DFS روبرو مفروض هستند: BFS : cbfedah ; DFS : cbfedha** (مهندسی کامپیوتر ـ سراسری ۸۶)




۱) این دو پیمایش برای هر دو گراف درست هستند.
۲) این دو پیمایش برای هر دو گراف غلط است.
۳) DFS داده شده برای هر دو درست است ولی BFS فقط برای گراف 1 درست است.
۴) BFS برای هر دو درست است ولی DFS فقط برای گراف 2 درست است.

**۲۱ـ ماتریس B از روی گراف $G=(V,E)$ به صورت زیر به دست آمده است:**

$$b_{ij}=\begin{cases}-1 & \text{اگر یال } j \text{، گره } i \text{ را ترک کند.}\\ 1 & \text{اگر یال } j \text{، به گره } i \text{ وارد شود.}\\ 0 & \text{در غیر این صورت}\end{cases}$$

**ماتریس $C=B.t_B$ ( $t_B$، ترانهاده B است ) چه را نشان می‌دهد ؟ می‌توان:** (مهندسی کامپیوتر ـ سراسری ۸۶)

۱) با مقدار $C_{ij}$ تعداد دورهای بین گره‌های $V_i,V_j$ را تعیین کرد.
۲) با مقدار $C_{ij}$ تعداد یال‌های از $V_i,V_j$ را تعیین کرد.
۳) با مقدار $C_{ij}$ تعداد یال‌های از $V_j,V_i$ را تعیین کرد.
۴) با مقدار $C_{ij}$ تعداد یال‌های بین گره $j,i$ را تعیین کرد.

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۲۲ـ** اگر گراف $G=(V,E)$ به صورت لیست مجاورت (Adjacency list) پیاده‌سازی شده باشد، پیچیدگی پیمایش Breadth – first چیست؟

(مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۶)

۱) $O(|V^2|)$ ۲) $O(|V|.|E|)$ ۳) $O(|V|^2.|E|)$ ۴) $O(|V|+|E|)$

**۲۳ـ** کدام گزاره نادرست است؟

(علوم کامپیوتر ـ سراسری ۸۶)

۱) الگوریتم DFS از استک استفاده می‌کند.

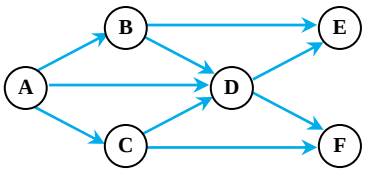
۲) الگوریتم DFS برای پیدا کردن Topological sort استفاده می‌شود.

۳) پیچیدگی زمان الگوریتم DFS در یک گراف سریع‌تر از BFS است.

۴) در الگوریتم DFS یک گراف اگر Backedge وجود نداشته باشد گراف بدون دور است.

**۲۴ـ** کدام گزینه یک ترتیب درست از رأس‌های گراف زیر را که در جستجوی اول ـ عمق (Depth-First Search) علامت ملاقات شدن (Visited) می‌خورند، نشان نمی‌دهد؟

(مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۷)




۱) ABCDEF

۲) ABEDFC

۳) ABDFEC

۴) ADEFBC

**۲۵ـ** الگوریتم تشخیص این که یک گراف جهت‌دار $G(V,E)$ بدون دور است دارای کدام پیچیدگی می‌باشد؟

(علوم کامپیوتر ـ سراسری ۸۷)

۱) $O(|V|\times|E|)$ ۲) $O(|V|)$ ۳) $O(|E|)$ ۴) $O(|V|+|E|)$

**۲۶ـ** پیچیدگی زمان پیدا کردن قطر یک گراف $G=(V,E)$ برابر است با:

(علوم کامپیوتر ـ سراسری ۸۷)

۱) $O(|V|^2)$ ۲) $O(|V|^2+|V|.|E|)$ ۳) $O(|V|^2.|E|)$ ۴) $O(|V|.|E|^2)$

**۲۷ـ** اگر $d[u]$ نخستین زمان ملاقات گره $u$ در جست‌وجوی عمق اول و $f[v]$ آخرین زمان ملاقات گره $v$ و یال $(u,v)$ یک Cross edge باشد، کدام رابطه درست است؟

(مهندسی کامپیوتر ـ سراسری ۸۸)

۱) $d[v]<d[u]<f[u]<f[v]$ ۲) $d[v]<f[v]<d[u]<f[u]$

۳) $d[u]<f[u]<d[v]<f[v]$ ۴) $d[v]<d[u]<f[v]<f[u]$

**۲۸ـ** در صورتی که $d[v]$ نخستین زمان ملاقات گره $v$ در جست‌وجوی سطح اول باشد، کدام عبارت در مورد جست‌وجوی سطح اول یک گراف جهت‌دار نادرست است؟

(مهندسی کامپیوتر ـ سراسری ۸۸)

۱) برای هر Back edge $(u,v)$ داریم، $0\le d[v]\le d[u]$

۲) برای هر Cross edge $(u,v)$ داریم، $d[v]\le d[u]+1$

۳) تعداد Tree Edge های حاصل کمتر از بقیه یال‌ها هستند.

۴) هیچ forward edge ای وجود ندارد.

**۲۹ـ** گراف جهت‌دار با مجموعه گره‌های $V=\{a,b,c,d,e\}$ و مجموعه یال‌های $E=\{(a,c),(a,e),(c,d),(d,e),(e,b),(e,c)\}$ را در نظر بگیرید. اگر این گراف با روش پیمایش عمق اول، پیمایش شود به ترتیب از چپ به راست، تعداد کمان‌های درخت $(tree-edges)$، کمان‌های برگشتی $(back-edges)$، کمان‌های ضربدری (cross edges) و کمان‌های جلورو (forward edges) برابر کدام است؟

(مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۸)

۱) $(4,0,1,1)$ ۲) $(4,1,1,0)$ ۳) $(3,2,1,0)$ ۴) $(4,1,0,1)$

**۳۰ـ** گراف بدون جهت $G(V,E)$ مفروض است. می‌خواهیم مشخص کنیم که آیا گراف شامل حلقه است یا نه. کوچکترین حد بالای زمان اجرای سریع‌ترین الگوریتم برای حل این مسأله کدام است؟

(مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۸)

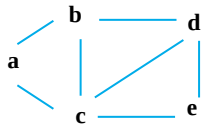
۱) $O(|E|)$ ۲) $O(|V|.|E|)$ ۳) $O(|V|)$ ۴) $O(|V|+|E|)$

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir

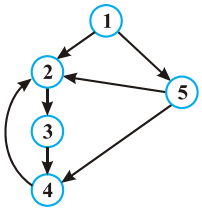


**۳۱ـ کدام مورد نمی‌تواند ترتیب اجرای الگوریتم BFS روی گراف روبه‌رو باشد؟** (مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۸)




۱) cedab
۲) bcdae
۳) abced
۴) acbde

**۳۲ـ پیمایش DFS گراف جهت‌دار زیر به ترتیب از چپ به راست $1,2,3,4,5$ می‌باشد. یال $<5,2>$ چه یالی می‌باشد؟** (علوم کامپیوتر ـ سراسری ۸۸)




۱) Back edge
۲) Cross edge
۳) Tree edge
۴) Forward edge

**۳۳ـ در جست و جوی BFS یک گراف جهت‌دار $G=(V,E)$ در طبقه‌بندی یال‌ها در زمان پیمایش کدام نوع از یال‌ها به هیچ وجه ایجاد نمی‌شود؟** (علوم کامپیوتر ـ سراسری ۸۸)

۱) Back edge ۲) Tree edge ۳) Cross edge ۴) Forward edge

**۳۴ـ کدام یک از الگوریتم‌ها در داخل خود الگوریتم DFS را دو بار صدا می‌کند؟** (علوم کامپیوتر ـ سراسری ۸۸)

۱) Topological sort
۲) پیدا کردن مؤلفه‌های مرتبط
۳) پیدا کردن مؤلفه‌های قویاً مرتبط
۴) پیدا کردن دور در یک گراف

**۳۵ـ الگوریتم dfs به روی گراف بدون جهت G اجرا شده است. زمان ورود و خروج از هر رأس به صورت زیر است. کدام یال در گراف G وجود ندارد؟** (مهندسی کامپیوتر ـ سراسری ۸۹)

| | a | b | c | d | e | f | g | h | i |
|---|---|---|---|---|---|---|---|---|---|
| ورود | 1 | 9 | 8 | 7 | 5 | 4 | 2 | 12 | 13 |
| خروج | 18 | 10 | 11 | 16 | 6 | 17 | 3 | 15 | 14 |

۱) fh ۲) di ۳) bd ۴) ch

**۳۶ـ کدام‌یک از گراف‌های زیر Bipartite است؟ (تعریف: گراف Bipartite گرافی است که نودهای آن قابل دسته‌بندی به دو گروه x و y هستند به گونه‌ای که نودهای گروه‌های x و y به همگروهی خودشان یال ندارند اما به گروه دیگر یال دارند.)** (مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۹)

(۱) (۲) (۳) (۴)

**۳۷ـ ماتریس تلاقی گراف جهت‌دار $G=(V,E)$ یک ماتریس $|V|\times|E|$ به نام $B=(b_{ij})$ است به‌طوریکه:**

$$b_{ij}=\begin{cases}-1 & \text{اگر یال } j \text{ رأس } i \text{ را ترک کند.}\\ 1 & \text{اگر یال } j \text{ وارد رأس } i \text{ شود.}\\ 0 & \text{در غیر این صورت}\end{cases}$$

**در این صورت اعضاء ماتریس $C=B.t_B$ چه چیزی را نشان می‌دهد؟ $t_B$ ترانهاده ماتریس B است؟** (مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۹)

۱) $C_{ij}$ تعداد یالهای بین دو رأس $i$ و $j$ را نشان می‌دهد.
۲) تعداد رئوس مانند $v_k$ را نشان می‌دهد که $(v_k,v_i)$ و $(v_k,v_j)$ یال‌های گراف هستند.
۳) $C_{ij}$ تعداد یال‌هایی را نشان می‌دهد که از رئوس $i$ و $j$ شروع شده و پایان مشترک دارند.
۴) $C_{ij}$ تعداد یال‌هایی را نشان می‌دهد که شروع مشترک دارند و پایان آنها رئوس $i$ و $j$ است.

www.modaresanesharif.ac.ir

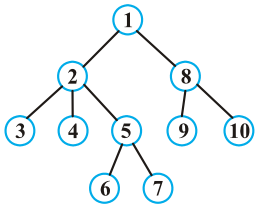
www.modaresanesharif.ac.ir



**۳۸ـ** در نمایش یالی (Adjacency Multilist) یک گراف $G=(V,E)$ با $|V|=n$ و $|E|=m$. ۱) چند Linked list و ۲) چه تعداد node وجود دارد (بدون در نظر گرفتن Head Node) (از چپ به راست) (علوم کامپیوتر ـ سراسری ۸۹)

۱) $|V|+|E|,|V|$ ۲) $|V|,|E|$ ۳) $|E|,|E|$ ۴) $|V|,|V|+|E|$

**۳۹ـ** اگر درخت حاصل از جستجوی DFS یک گراف درخت زیر باشد یال a می‌تواند Forward edge و یال b می‌تواند یک cross edge باشد. یال‌های a و b کدامند؟ (علوم کامپیوتر ـ سراسری ۸۹)




۱) $a=(1,4)\ ,\ b=(6,7)$
۲) $a=(3,4)\ ,\ b=(9,10)$
۳) $a=(6,1)\ ,\ b=(6,2)$
۴) $a=(9,10)\ ,\ b=(7,1)$

**۴۰ـ** در یک گراف هم‌بند، بدون جهت، بدون حلقه و بدون یال چندگانه، کدام‌یک از نامساوی‌های زیر ممکن است برقرار نباشد؟ (مهندسی کامپیوتر ـ سراسری ۹۰)

۱) $|E|\leq|V|^2$ ۲) $|V|\geq|E|/2$ ۳) $|E|\geq|V|-1$ ۴) $|V|\leq|E|^2+1$

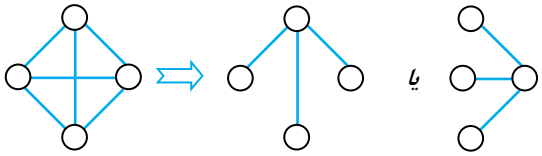
www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir

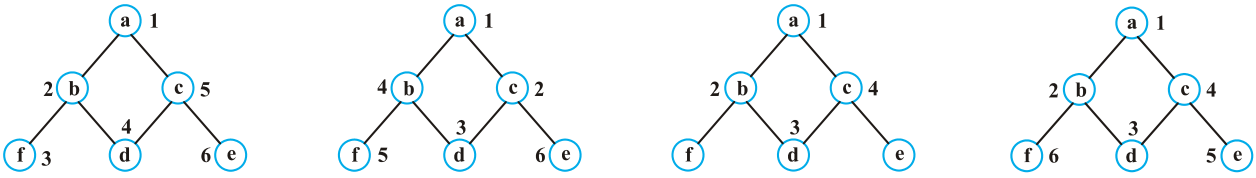


## پاسخنامه تست‌های طبقه‌بندی شده کنکوری فصل دوازدهم

**۱ـ گزینه «۱»** درخت پوشا باید تعدادی از یال‌ها و همه گره‌ها را پوشش دهد.
مثلاً درخت پوشای گراف روبرو:

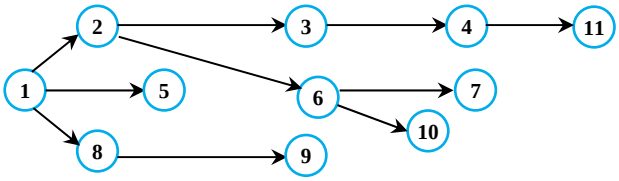



**۲ـ گزینه «۲»** پیمایش اول عمق برای هر یک از گزینه‌ها در زیر مشخص شده است. اما در گزینه ۲ بعد از c، باید e ملاقات شود که اشتباهاً f آمده است.




گزینه ۴ گزینه ۳ گزینه ۲ گزینه ۱

**۳ـ گزینه «۱»** درخت DFS گراف مورد نظر به صورت روبرو خواهد بود:




ترتیب پیمایش صحیح در گزینه اول آمده است.

**۴ـ گزینه «۱»** در گراف‌های مسطح رابطه زیر نشان‌دهنده تعداد ناحیه‌های موجود در گراف می‌باشد:

$$\text{تعداد ناحیه‌ها} = |E| - |V| + 2$$

تعداد دورهای ساده با توجه به تعداد ناحیه‌ها قابل محاسبه می‌باشد که زمان این محاسبه $\theta(1)$ خواهد بود.

**۵ـ گزینه «۴»** ابتدا رأس شماره 1 ملاقات می‌گردد سپس، هریک از رئوس 5، 2 یا 6 می‌توانند ملاقات گردند (6 حالت مختلف امکان‌پذیر است) بنابراین، هریک از رأس‌های 7 یا 3 را می‌توان ملاقات نمود. در نتیجه بیش از 12 ترتیب مرتب‌سازی توپولوژیک وجود خواهد داشت که فقط گزینه چهارم می‌تواند صحیح باشد.

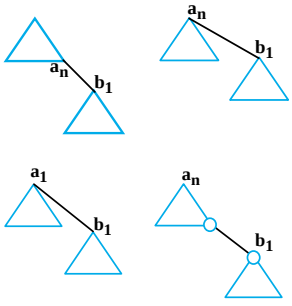
**۶ـ گزینه «۱»** بین تمام جفت رأس‌های گراف مسیر رفت و برگشت وجود دارد.

**۷ـ گزینه «۲»** با استفاده از الگوریتم DFS می‌توان این‌کار را انجام داد (وجود یال پشتی در این الگوریتم نشان‌دهنده وجود دور در گراف می‌باشد). با توجه به این که زمان الگوریتم DFS برابر $O(|E|+|V|)$ می‌باشد، بنابراین زمان یافتن دور در گراف‌های جهت‌دار برابر $O(|E|+|V|)$ و در گراف‌های بدون جهت برابر $O(|V|)$ می‌باشد.

**۸ـ گزینه «۴»** با توجه به شکل درخت DFS ادعای سه مؤلفه همبندی برای G رد می‌شود، چون با حذف a دو مؤلفه همبندی خواهیم داشت. از طرفی با توجه به درجه a که 4 است و فرزندان آن درخت DFS مطمئناً دو تا از فرزندان آن یک یال مستقیم با a دارند و این یعنی دور وجود دارد. ادعای ۳ نیز غلط است چون با حذف a دو مؤلفه همبندی مجزا خواهیم داشت.

**۹ـ گزینه «۲»**

ادعای گزینه ۲ نادرست می‌باشد چرا که در این صورت $b_1$ خودش باید در بین فرزندان $a_1$ در پیمایش BFS باشد نه بعد از آخرین عنصر درخت $T_1$. (به شکل‌های مقابل دقت کنید)




**۱۰ـ گزینه «۲»** یال‌های پشتی در جستجوی اول عمق برای این گراف سه تاست. (گراف دارای 3 دور می‌باشد)

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



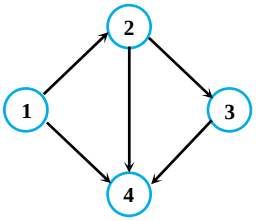
**۱۱ـ گزینه «۳»** جستجوی اول عمق تمام گره‌هایی که می‌توانند از یک گره در دسترس باشند را در زیر درختی که آن گره ریشه آن است قرار می‌دهد. بنابراین ادعای گزینه ۱ نادرست می‌باشد، گزینه ۲ نیز به همین ترتیب؛ اما ادعای گزینه ۳ می‌تواند صحیح باشد (اگر دو مسیر از 1 به 4 وجود داشته باشد یکی مستقیم و یکی از طریق 2 این ادعا تأیید می‌شود). گزینه ۴ نیز نادرست است.

**۱۲ـ گزینه «۱»** در درخت فراگیر به دست آمده از جستجوی اول سطح می‌تواند بین گره‌های یک سطح و هریک از گره‌های سطح بعدی یال وجود داشته باشد.

**۱۳ـ گزینه «۳»** با توجه به درخت پوشای به دست آمده از پیمایش Dfs گراف G، بعد از پیمایش گرهٔ 3 به گرهٔ 2 برگشته و بعد از آن گرهٔ 4 ملاقات می‌شود و بعد از ملاقات گرهٔ 4 به گرهٔ 2 برگشته و اگر بین گرهٔ 2 و 5 یال باشد باید گرهٔ 5 ملاقات شود. با توجه به درخت پوشای به دست آمده نمی‌تواند بین گره 2 و 5 یالی باشد و همین طور نمی‌تواند بین گرهٔ 2 و6 یا 7 و 6 یالی وجود داشته باشد؛ ولی اگر یالی بین دو گرهٔ 1 و 4 باشد روش پیمایش تغییری نمی‌کند.

**۱۴ـ گزینه «۱»** با توجه به درخت پوشای به دست آمده از پیمایش BFS گراف G، بعد از پیمایش گرهٔ ۱ فقط گره‌های 2 و 3 پیمایش می‌شوند، پس نمی‌تواند بین دو گرهٔ 1 و 5 یالی وجود داشته باشد و همین طور نمی‌تواند بین دو گرهٔ 4 و 1 و 7 و 1 یالی باشد و نیز بعد از پیمایش گرهٔ 3 فقط گرهٔ 6 پیمایش می‌شود، پس نمی‌تواند بین دو گرهٔ 7 و 3 یالی وجود داشته باشد؛ ولی اگر بین دو گرهٔ 5 و 4 یالی باشد روش پیمایش تغییر نمی‌کند.

**۱۵ـ گزینه «۴»** به ازای $n = 4$ شکل گراف به صورت زیر خواهد بود:




حال اگر گراف را بدون جهت در نظر بگیریم 3 دور ساده وجود خواهد داشت که تنها در رابطه گزینه چهارم صادق است. (اگر گراف را بدون جهت در نظر نگیریم با توجه به این که از بزرگترین رأس هیچ یالی خارج نمی‌شود، بنابراین هیچ دوری نیز وجود نخواهد داشت).

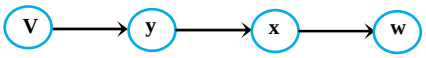
**۱۶ـ گزینه «۴»** وجود یا عدم وجود دور اویلری به درجه رئوس گراف بستگی دارد. وجود یال Cross در مورد دور اویلری (و یا مسیر هامیلتونی) اطلاعاتی ایجاد نمی‌کند.

**۱۷ـ گزینه «۲»** با توجه به این‌که گراف کامل است بین تمام رأس‌ها مسیر وجود دارد بنابراین، با DFS یک مسیر بدون بازگشت با $n-1$ یال طی می‌شود و الگوریتم متوقف می‌گردد در نتیجه مابقی یال‌ها ملاقات نمی‌شوند.

$$\frac{n(n-1)}{2} - (n-1) = \frac{n(n-1) - 2(n-1)}{2} = \frac{(n-2)(n-1)}{2}$$

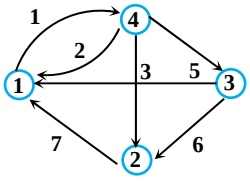
**۱۸ـ گزینه «۱»** BFS دقیقاً همین کار را انجام می‌دهد.

**۱۹ـ گزینه «۴»** اگر دور وجود داشته باشد آنگاه از W به V نیز مسیر وجود خواهد داشت. به عنوان مثال گراف زیر را در نظر بگیرید:




**۲۰ـ گزینه «۱»** با توجه به الگوریتم‌های پیمایش BFS و DFS به راحتی قابل بررسی است که دو پیمایش داده شده در هر دو گراف صحیح است.

**۲۱ـ گزینه «۴»** ماتریس بیان شده در این تست در حقیقت ماتریس برخورد یک گراف جهت‌دار می‌باشد. به عنوان مثال گراف زیر را در نظر بگیرید:




در این صورت ماتریس B عبارتست از:

| | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| 1 | −1 | 1 | 0 | 0 | 1 | 0 | 1 |
| 2 | 0 | 0 | 0 | 1 | 0 | 1 | −1 |
| 3 | 0 | 0 | 1 | 0 | −1 | −1 | 0 |
| 4 | 1 | −1 | −1 | −1 | 0 | 0 | 0 |

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



ماتریس $B.B^t$ عبارتست از:

$$C:\begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \end{array}\begin{array}{c} \begin{array}{cccc} 1 & 2 & 3 & 4 \end{array} \\ \begin{bmatrix} 4 & -1 & -1 & -2 \\ -1 & 3 & -1 & -1 \\ -1 & -1 & 3 & -1 \\ -2 & -1 & -1 & 4 \end{bmatrix} \end{array}$$

همان‌گونه که مشخص است عناصر قطر اصلی نشان‌دهنده تعداد یال‌های برخوردکننده با رأس مورد نظر است (حاصل‌جمع درجه ورودی و خروجی) و قدرمطلق سایر عناصر تعداد یال‌های بین دو رأس دلخواه را نشان می‌دهد.

---

**۲۲ـ گزینه «۴»** زمان اجرای الگوریتم BFS در یک گراف که بوسیله لیست پیوندی پیاده‌سازی شده برابر $O(|V|+|E|)$ می‌باشد. دقت نمایید که اگر گراف به وسیله ماتریس مجاورتی پیاده‌سازی شده باشد آنگاه زمان $O(|V|^2)$ خواهد بود.

---

**۲۳ـ گزینه «۳»** در حالت کلی مورد سوم صحیح نمی‌باشد. به نکات زیر در مورد گزینه‌های مطرح شده دقت کنید:

الگوریتم DFS از پشته استفاده می‌کند.

الگوریتم BFS از صف استفاده می‌کند.

الگوریتم DFS برای پیدا کردن Topological sort استفاده می‌شود.

پیچیدگی زمان الگوریتم DFS در یک گراف سریع‌تر از BFS نیست (بیشتر یا مساوی است).

در الگوریتم DFS یک گراف اگر Backedge وجود نداشته باشد گراف بدون دور است.

---

**۲۴ـ گزینه «۱»** در گزینه ۱، ابتدا A ملاقات شده و بعد B ملاقات می‌شود، سپس باید یکی از دو گره E یا D ملاقات شود که برخلاف این امر C ملاقات شده است و این موضوع در جستجوی Dfs گراف درست نیست.

---

**۲۵ـ گزینه «۴»** کافیست از DFS استفاده کنیم. با استفاده از الگوریتم DFS می‌توان این‌کار را انجام داد (وجود یال پشتی در این الگوریتم نشان‌دهنده وجود دور در گراف می‌باشد). با توجه به این که زمان الگوریتم DFS برابر $O(|E|+|V|)$ می‌باشد، بنابراین زمان یافتن دور در گراف‌های جهت‌دار برابر $O(|E|+|V|)$ و در گراف‌های بدون جهت برابر $O(|V|)$ می‌باشد.

---

**۲۶ـ گزینه «۲»** قطر یک گراف همان بزرگترین مسیر موجود در گراف است که دارای زمان $O(|V|^2+|V||E|)$ خواهد بود. (اگر به جای گراف درخت ذکر شده باشد جواب $O(|V|)$ خواهد بود.)

---

**۲۷ـ گزینه «۲»** یال $(u,v)$ یک یال ضربدری است اگر بازه مربوط به زمان‌های discovering time و finishing time رأس u با بازه v متداخل نباشد و بزرگتر از آن باشد یعنی:

dv fv du fu
[ ] [ ]

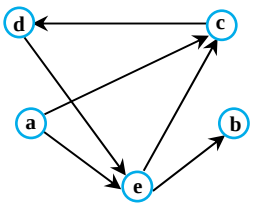
---

**۲۸ـ گزینه «۳»** اگر گراف n رأس داشته باشد و همبند باشد، آنگاه می‌تواند $n-1$ یال درختی وجود داشته باشد.

---

**۲۹ـ گزینه «۴»**

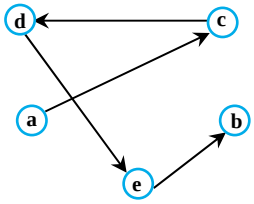
با توجه به مجموعه یال‌های داده شده گراف موردنظر به صورت مقابل می‌باشد:




www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir






درخت DFS مربوط به این گراف به صورت مقابل می‌باشد:

بنابراین یال‌های $(a , c)$، $(c , d)$، $(d , e)$ و $(a , b)$ یال‌های درختی می‌باشند و یال $(a , e)$ یک یال پیش‌رو است و یال $(e , c)$ یک یال پشتی است و هیچ یال ضربدری وجود ندارد.

---

**۳۰ـ گزینه «۳»** در اینجا وجود حلقه در گراف خواسته شده است نه وجود دور، بنابراین کافیست عناصر روی قطر اصلی ماتریس مجاورتی را بررسی کنیم. اگر مقدار عنصری روی قطر اصلی ۱ بود یعنی، آن رأس دارای حلقه است که این کار از مرتبه $O(|v|)$ قابل انجام است.

---

**۳۱ـ گزینه «۳»** در گزینه سوم پس از a ابتدا b و سپس c ملاقات شده است بنابراین، رأس d باید زودتر از e ملاقات شود. (زیرا d از طریق b قابل ملاقات است اما رأس e از طریق b قابل ملاقات نمی‌باشد.)

---

**۳۲ـ گزینه «۲»** با توجه به نوع پیمایش داده شده، هیچ‌یک از رأس‌های 2 یا 5 یکی از اجداد دیگری در جنگل پوشای DFS نیست بنابراین، یال موردنظر یک یال تقاطعی است.

---

**۳۳ـ گزینه «۴»** در پیمایش BFS یال forward ایجاد نمی‌شود.

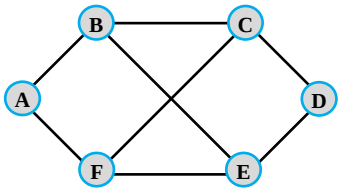
---

**۳۴ـ گزینه «۳»** برای پیدا کردن مؤلفه‌های همبند قوی در یک گراف، الگوریتم DFS یک بار بر روی گراف اصلی و یک بار بر روی ترانهاده آن اجرا می‌گردد.

---

**۳۵ـ گزینه «۴»** با توجه به این که finishing time رأس c برابر 11 می‌باشد و discovering time رأس h برابر 12 می‌باشد بنابراین، یالی از c به h نمی‌تواند وجود داشته باشد (اگر یال ch وجود داشته باشد، آنگاه finishing time رأس c باید بیشتر از 12 باشد).

---




**۳۶ـ گزینه «۱»** اگر گراف گزینه ۱ را به شکل مقابل در نظر بگیریم:

آنگاه اگر مجموعه‌های $x, y$ را به صورت زیر انتخاب کنیم گراف دو بخشی خواهد بود.

$$X = \{A, C, E\} \ , \ y = \{B, F, D\} .$$

---

**۳۷ـ گزینه «۱»** درایه‌های ماتریس مورد نظر نشان‌دهنده تعداد یال‌های موجود بین هر دو رأس دلخواه گراف می‌باشد.

---

**۳۸ـ گزینه «۲»** اگر گراف جهت‌دار باشد، آنگاه تعداد node ها برابر $|E|$ و تعداد لیست‌ها برابر $|V|$ خواهد بود. (اما اگر گراف بدون جهت باشد تعداد node ها برابر $2|E|$ خواهد بود).

---

**۳۹ـ گزینه «۱»** با توجه به درخت داده شده یال‌های $(3,4)$ و $(9,10)$ نمی‌توانند وجود داشته باشند. و همچنین یال $(6,2)$ در صورت وجود یال forward و یا back خواهد بود و نمی‌تواند Cross باشد.

---

**۴۰ـ گزینه «۲»** رابطه $|E| \le |V|^2$ در هر گراف ساده برقرار است. رابطه $|E| \ge |V| - 1$ در هر گراف همبند برقرار است.

از ترکیب دو عبارت بالا می‌توان گفت:

$$(|V|-1)^2 \le |E|^2 \Rightarrow (|V|-1)^6 + 1 \le |E|^2 + 1 \Rightarrow |V| \le (|V|-1)^2 + 1 \le |E|^2 + 1 \Rightarrow |V| \le |E|^2 + 1$$

اما گزینه دوم لزوماً برقرار نیست.

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



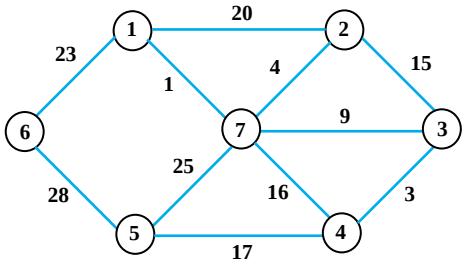
# فصل سیزدهم

## « الگوریتم‌های مبتنی بر فاصله در گراف »

### تست‌های طبقه‌بندی شده کنکوری فصل سیزدهم

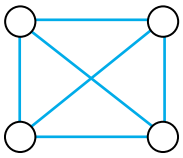
#### «یافتن MST»

۱ـ هزینهٔ درخت پوشای مینیمم (minimum spanning tree) گراف زیر چیست؟ (مهندسی کامپیوتر ـ آزاد ۷۷)

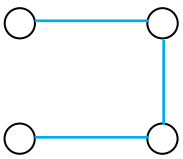
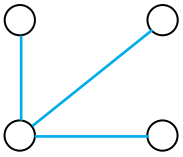
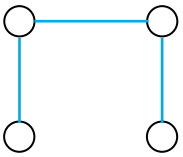



۱) 68
۲) 41
۳) 81
۴) 57

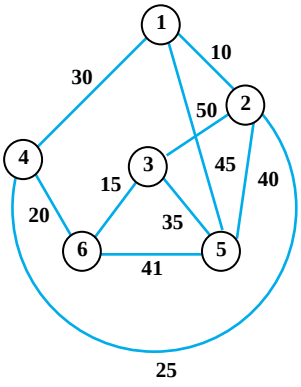
۲ـ گراف زیر داده شده است. (مهندسی کامپیوتر ـ آزاد ۷۷)

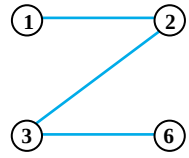
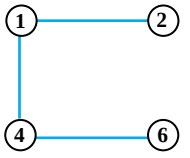
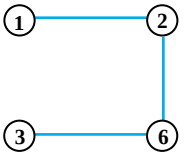
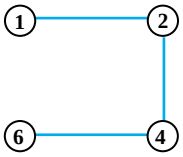


کدامیک از انتخاب‌های زیر Spanning Tree این گراف است؟

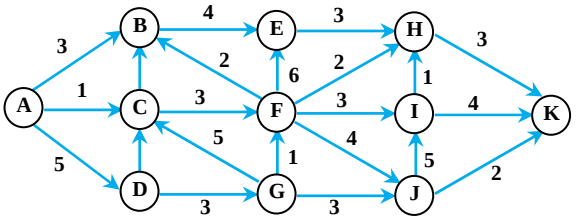
۱) 
۲) 
۳) 
۴) هر سه مورد

۳ـ اگر برای پیداکردن درخت پوشای مینیمم زیر از الگوریتم prim استفاده کنیم، کدامیک از گزینه‌های زیر درخت حاصله در انتهای مرحله سوم این الگوریتم را به ما می‌دهد؟ (مهندسی کامپیوتر ـ آزاد ۷۸)




۱) 
۲) 
۳) 
۴) 

۴ـ برای گراف زیر درخت پوشای مینیمم دارای چه وزنی است؟ (مهندسی کامپیوتر ـ آزاد ۷۸)



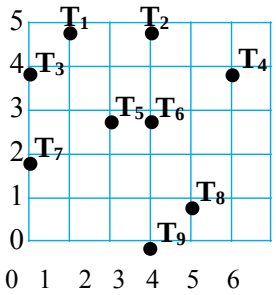

۱) 18
۲) 19
۳) 20
۴) 21

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۵ـ یک مهندس برق، مداری را طراحی نموده است دارای 9 ترمینال، که باید ولتاژی معادل 5 ولت به آنها متصل گردد. فرض کنید 5 ولت به یکی از ترمینال‌ها وصل است. برای اینکه کمترین سیم‌بندی در مدار به کار رود، بگوئید حداقل چقدر سیم لازم دارد؟ (فاصله هر سطر و ستون را یک سانتی‌متر در نظر بگیرید).** (مهندسی کامپیوتر ـ سراسری ۸۱)




۱) $8+2\sqrt{2}+2\sqrt{5}$

۲) $6+2\sqrt{2}+2\sqrt{5}$

۳) $5+2\sqrt{2}+3\sqrt{5}$

۴) $7+2\sqrt{2}+2\sqrt{5}$

**۶ـ کدام گزینه در مورد الگوریتم‌های کروسکال و پریم برای ایجاد درخت پوشای کمینه درست است؟** (مهندسی کامپیوتر ـ سراسری ۸۲)

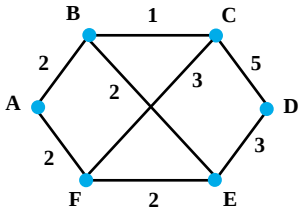
۱) زمان اجرای هر دو الگوریتم روی گراف‌های یکسان مساوی است.

۲) هر دو الگوریتم روی گراف‌های یکسان، درخت پوشای یکسان تولید می‌کنند.

۳) مجموع طول اضلاع درخت پوشا در هر دو الگوریتم یکسان است.

۴) هر دو الگوریتم با رشد و بهم پیوستن یک جنگل از درختها، درخت پوشا را تولید می‌کنند.

**۷ـ گراف بدون جهت و وزن‌دار زیر را در نظر می‌گیریم. اگر نقطه A را به عنوان مبدأ انتخاب کنیم و الگوریتم‌های کروسکال و پریم را جداگانه بر روی این گراف اجرا کنیم، لبه (یال)های انتخاب شده به ترتیب مطابق با کدام گزینه می‌باشد؟** (مهندسی کامپیوتر ـ سراسری ۸۲)




۱) کروسکال : BC , BE , FE , AF , ED
پریم : AF , FE , BE , BC , ED

۲) کروسکال : BC , BA , AF , FE , ED
پریم : AF , FE , EB , BC , ED

۳) کروسکال : BC , FE , AF , AB , ED
پریم : AB , AF , FE , BC , ED

۴) کروسکال : BC , BE , AB , ED , FE
پریم : AF , FE , ED , AB , BC

**۸ـ فرض کنیم گراف $G=(V,E)$ یک گراف بی‌جهت و M یک درخت فراگیر مینیمم برای G باشد، کدام عبارت درست است؟** (علوم کامپیوتر ـ سراسری ۸۲)

۱) مسیر M بین هر جفت رأس $V_1$ و $V_2$ کوتاهترین مسیر است.

۲) مسیر M بین هر جفت رأس $V_1$ و $V_2$ کوتاهترین مسیر نمی‌باشد.

۳) تمام مسیرهای موجود در M کوتاهترین مسیر می‌باشند.

۴) بین تمام رئوس در M مسیری وجود ندارد.

**۹ـ کدام الگوریتم از روش حریصانه (greedy) استفاده نمی‌کند؟** (علوم کامپیوتر ـ سراسری ۸۲)

۱) پیدا کردن کوتاهترین مسیر در یک گراف با روش دایکسترا

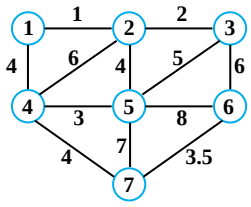
۲) پیدا کردن مینیمم درخت فراگیر Kruskal

۳) پیدا کردن مینیمم درخت فراگیر Prim

۴) پیدا کردن بزرگترین زیر رشته مشترک بین دو رشته

**۱۰ـ الگوریتم Kruskal در مرحله چهار خود، کدام کمان از گراف زیر را به عنوان کمان درخت پوشای با حداقل هزینه در نظر می‌گیرد؟**

(مهندسی IT ـ سراسری ۸۳)




۱) $(1,4)$

۲) $(4,5)$

۳) $(6,7)$

۴) هیچ‌کدام

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۱۱ـ یک گراف هم بند $G = (V, E)$ که وزن‌های یال‌های آن 1 است را در نظر بگیرید. می‌خواهیم درخت فراگیر کمینه (MST) این گراف را پیدا کنیم.**

(مهندسی کامپیوتر ـ سراسری ۸۴)

۱) این کار را می‌توان در $O(|E| \log |E|)$ انجام داد.
۲) این کار را می‌توان در $O(|V| |E|)$ انجام داد.
۳) این کار را می‌توان در $O(|V| \log |E|)$ انجام داد.
۴) این کار را می‌توان در $O(|E|)$ انجام داد.

**۱۲ـ در یک گراف همبند بدون جهت و وزن‌دار، کدامیک از گزینه‌های زیر در مورد درخت پوشای کمینه ایجاد شده توسط الگوریتم‌های prim و kruskal درست است؟**

(مهندسی کامپیوتر ـ سراسری ۸۴)

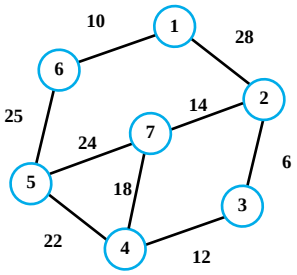
۱) با هر وزنی این دو الگوریتم درخت‌های یکسانی تولید می‌کند.
۲) درخت تولید شده توسط الگوریتم Prim مستقل از رأس شروع در آن الگوریتم است.
۳) اگر وزن یال‌ها مجزا باشند، درخت‌های تولیدی این دو الگوریتم یکسان هستند.
۴) اگر وزن همه یال‌ها برابر باشند، این دو الگوریتم درخت‌های یکسانی را تولید می‌کنند.

**۱۳ـ کدام یک از گزاره‌های ذیل، در مورد محاسبه درخت پوشای کمینه برای یک گراف همبند (Connected)، درست است؟**

(مهندسی IT ـ سراسری ۸۴)

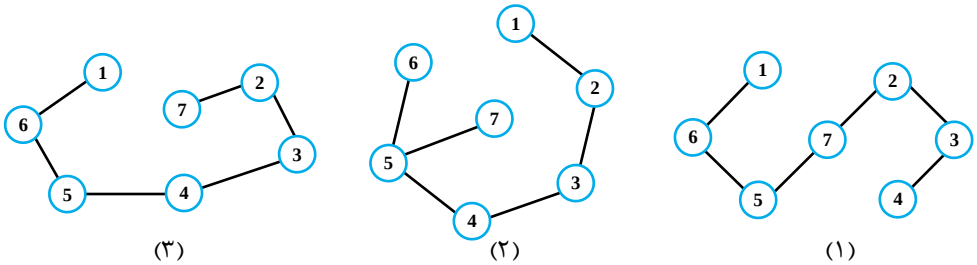
۱) در روش Prim در هر مرحله، همبندی درخت رعایت می‌شود و زمان مصرفی الگوریتم Prim و الگوریتم Kruskal، یکسان است.
۲) در روش Prim در هر مرحله، همبندی درخت، رعایت می‌شود، اما در روش Kruskal در مرحله پایانی، این امر اتفاق می‌افتد. اما زمان مصرفی روش Prim، بهتر است.
۳) در روش Kruskal، در هر مرحله، همبندی درخت رعایت می‌شود، اما در روش Prim در مرحله پایانی، این امر اتفاق می‌افتد، اما زمان مصرفی روش Prim، بهتر است.
۴) در روش Kruskal، در هر مرحله، همبندی درخت رعایت نمی‌شود، اما در پایان الگوریتم، درخت همبند است، اما زمان مصرفی آن از روش Prim بهتر است.

**۱۴ـ گراف ذیل را درنظر بگیرید:**




**درخت پوشا با حداقل هزینه برابر است با:**

(مهندسی کامپیوتر و مهندسی IT ـ آزاد ۸۴)




(۱) (۲) (۳) ۴) هیچ‌کدام

**۱۵ـ T درخت فراگیر کمینه برای یک گراف G=(V,E) است. وزن یک یال $e \in E$ را کاهش می‌دهیم. با چه مرتبه‌ای می‌توان درخت فراگیر کمینه گراف جدید را به دست آورد؟**

(مهندسی کامپیوتر ـ سراسری ۸۵)

۱) $O(|V|)$ ۲) $O(|V|+|E|)$ ۳) $O(|E|\log|V|)$ ۴) $O(|V|\log|E|)$

**۱۶ـ فرض کنیم که T یک درخت پوشای مینیمم برای گراف G باشد. اگر مقادیر برخی از یال‌های G کاهش پیدا کنند و گراف حاصل را $G'$ بنامیم در این صورت:**

(مهندسی کامپیوتر ـ سراسری ۸۵)

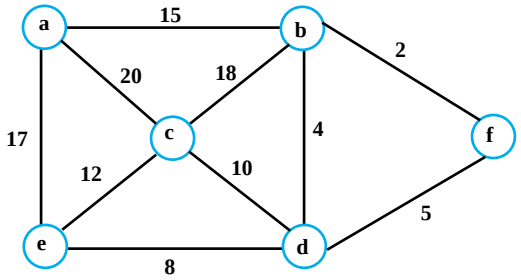
۱) T در هر حال یک درخت پوشای مینیمم از $G'$ خواهد بود.
۲) اگر یال‌هایی که مقدارشان کاهش پیدا کرده همگی متعلق به T باشند، آنگاه T یک درخت پوشای مینیمم برای $G'$ خواهد بود.
۳) حتی اگر فقط بخشی از یال‌هایی که مقدارشان کاهش یافته متعلق به T باشند، آنگاه T یک درخت پوشای مینیمم از $G'$ خواهد بود.
۴) T یک درخت پوشای مینیمم از $G'$ خواهد بود اگر یال‌هایی که مقدارشان کاهش پیدا کرده همگی متعلق به T باشند و کلیهٔ یالهای متعلق به T کاهش یافته باشند.

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۱۷ـ** ترتیب انتخاب یال‌ها در الگوریتم **Kruskal** بر روی گراف زیر کدام مورد است (از چپ به راست)؟ (مهندسی IT ـ سراسری ۸۵)



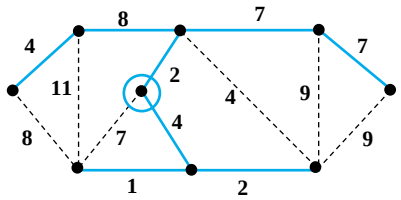

۱) $(b,f),(b,d),(e,d),(c,d),(a,b)$

۲) $(a,b),(b,f),(b,d),(f,d),(c,d)$

۳) $(b,f),(b,d),(e,d),(e,c),(a,e)$

۴) هیچ‌کدام

**۱۸ـ** در گراف زیر لبه‌های پررنگ، درخت پوشای حداقل را با استفاده از روش **prim** نمایش می‌دهد. گره‌ای که با دایره مشخص شده نقطه شروع است. در این حالت آخرین گره‌ای که به درخت پوشا متصل شده است دارای چه وزنی است؟ (مهندسی IT ـ سراسری ۸۶)




۱) 8

۲) 7

۳) 4

۴) 2

**۱۹ـ** یک گراف $G=(V,E)$ با یک یال $e=(u,v)$ را در نظر بگیرید. می‌خواهیم درخت فراگیر کمینه (MST) برای این گراف را پیدا کنید که حتماً شامل $e$ باشد. کدام یک از راه‌های زیر همیشه درست است؟ (مهندسی کامپیوتر ـ سراسری ۸۶)

۱) $e$ را حذف می‌کنیم (بدون حذف $v,u$)، MST گراف حاصل را به دست می‌آوریم و سپس $e$ را اضافه می‌کنیم.

۲) $v,u$ را در هم ادغام می‌کنیم و MST گراف حاصل را به دست می‌آوریم، سپس $e$ را اضافه می‌کنیم.

۳) همه یال‌های متصل به $v,u$ را حذف می‌کنیم. MST گراف حاصل را به دست می‌آوریم و سپس $e$ را اضافه می‌کنیم.

۴) این مسئله راه حل چند جمله‌ای ندارد.

**۲۰ـ** در گراف وزن‌دار $G(V,E)$ که $w_e$ وزن یال $e$ می‌باشد اگر درخت فراگیر مینیمم این گراف $\sum_{e\in T} w_e$ را مینیمم کند در آن صورت، الزاماً $\sum_{e\in T} w_e^2$ نیز مینیمم می‌گردد و بالعکس، گزاره بالا صحیح است یا نادرست؟ (علوم کامپیوتر ـ سراسری ۸۶)

۱) خیر

۲) بله

۳) فقط برای گراف‌ها بدون دور صحیح است.

۴) فقط در صورتی که $w_e > 0$ به ازای هر یال باشد صحیح است.

**۲۱ـ** یک گراف بسیار پراکنده شامل $n$ گره مفروض است. می‌خواهیم با استفاده از الگوریتم **kruskal** کوچک‌ترین درخت پوشای این گراف را به‌دست آوریم. در این صورت پیچیدگی زمانی این الگوریتم برابر است با: (مهندسی کامپیوتر ـ آزاد ۸۶)

۱) $O(n^2)$

۲) $O(n^2 \log n)$

۳) $O(n)$

۴) $O(n \log n)$

**۲۲ـ** الگوریتم زیر را بر روی یک گراف هم‌بند بدون جهت و وزن‌دار $G=(V,E)$ در نظر بگیرید: (مهندسی کامپیوتر ـ سراسری ۸۷)

تا وقتی که گراف $G$ دوری به نام $C$ دارد این کار را تکرار کن

یال $e$ با بیش‌ترین وزن در $C$ را به دست آور و آن را از $G$ حذف کن

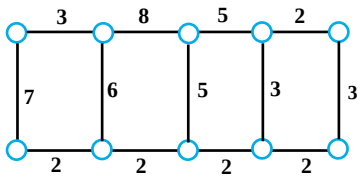
۱) این الگوریتم ممکن است ختم نشود.

۲) گراف حاصل ممکن است هم‌بند نباشد.

۳) در انتها گراف حاصل یک درخت فراگیر کمینه برای گراف اولیه است.

۴) در انتها گراف حاصل درخت فراگیر برای $G$ اولیه است ولی لزوماً کمینه نیست.

**۲۳ـ** یک شرکت در نظر دارد بین **10** اتاق شرکت یک شبکه **LAN** ایجاد نماید. گراف زیر اتاق‌هایی که امکان سیم‌کشی مستقیم بین آن‌ها وجود دارد را نمایش می‌دهد. وزن هر لبه هزینه سیم‌کشی بین آن دو اتاق است. اگر هدف انجام سیم‌کشی با حداقل هزینه باشد، کدام الگوریتم زیر برای حل این مسئله مناسب است؟ (مهندسی IT ـ سراسری ۸۷)




۱) Prim

۲) Floyd

۳) Kruskal

۴) Dijkstra

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



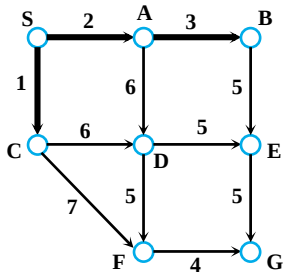
**۲۴ـ درخت پوشای مینیمم برای گراف وزن‌دار G، در چه صورت یکتا می‌باشد؟** (علوم کامپیوتر ـ سراسری ۸۷)

۱) در صورتی که گراف G یک درخت باشد.
۲) در صورتی که در گراف G هیچ دو یال دارای وزن یکسان نباشند.
۳) در صورتی که گراف G غیر جهت‌دار و مرتبط باشد.
۴) موارد ۱ و ۲

**۲۵ـ گراف بدون جهت G با n رأس و e یال را «تک‌ـ دوره» می‌گوییم، اگر هم‌بند باشد و فقط یک عدد دور داشته باشد. با چه سرعتی می‌توان درخت فراگیر کمینه‌ی یگ گراف تک‌ـ دوره با وزن‌های مثبت را به دست آورد؟** (مهندسی کامپیوتر ـ سراسری ۸۸)

۱) $O(n^2)$ ۲) $O(e+n)$ ۳) $O(e+n\lg n)$ ۴) $O(e\times n)$

**۲۶ـ فرض کنید که الگوریتم Prim روی گراف شکل زیر اجرا شده است. یال‌های پررنگ یال‌هایی است که تا به حال پردازش شده‌اند. یال بعدی کدام یال است؟** (مهندسی IT ـ سراسری ۸۸)




۱) یال FG
۲) یال CF
۳) یال CD
۴) یال BE

**۲۷ـ فرض می‌کنیم در گراف وزن‌دار $G=(V,E)$ یال $e\in E$ سنگین‌ترین یال در یک دور موجود در گراف باشد، در این صورت آیا درخت فراگیر کمینه شامل این یال می‌باشد یا خیر؟** (علوم کامپیوتر ـ سراسری ۸۸)

۱) خیر ـ هیچ‌وقت ۲) بله ـ همیشه ۳) در موارد خاص بله ۴) مشخص نمی‌باشد.

**۲۸ـ پیچیدگی زمان $O(|V|+|E|)$ برای یک گراف $G(V,E)$ به ازای کدام الگوریتم صادق نمی‌باشد؟** (علوم کامپیوتر ـ سراسری ۸۸)

۱) پیدا کردن مؤلفه‌های قویاً مرتبط
۲) پیدا کردن درخت پوشا
۳) Topological sort
۴) پیدا کردن درخت پوشای کمینه

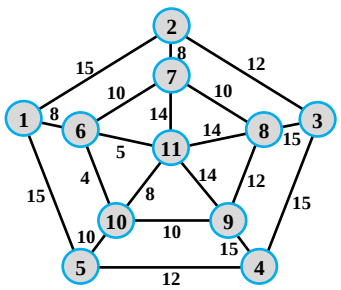
**۲۹ـ «درخت فراگیر گلوگاه» (د. ف. گلوگاه) T در یک گراف بدون جهت و وزن‌دار G یک درخت فراگیر (د. ف) است که وزن سنگین‌ترین یال آن نسبت به هر درخت فراگیر (نه لزوماً کمینه‌ی) دیگر کمینه باشد. می‌گوییم که «مقدار» یک درخت فراگیر گلوگاه، وزن سنگین‌ترین یال آن درخت است. می‌دانیم که درخت فراگیر کمینه (د.ف.کمینه) و درخت فراگیر بیشینه (د.ف.بیشینه) به ترتیب درخت فراگیر با کم‌ترین و بیش‌ترین وزن (مجموع وزن یالهای آن) ممکن هستند. کدام یک ازگزینه‌های زیر درست است؟** (مهندسی کامپیوتر ـ سراسری ۸۹)

۱) هر د.ف. گلوگاه یک د.ف کمینه است.
۲) هر د.ف. کمینه یک د.ف گلوگاه هم هست.
۳) هر د.ف. بیشینه یک د.ف گلوگاه است.
۴) هر د.ف. گلوگاه یک د.ف بیشینه است.

**۳۰ـ فرض کنید G یک گراف بدون جهت باشد که هیچ دو یال آن دارای وزن یکسان نیستند. کدام گزینه در مورد گراف G صحیح است؟ (دومین زیردرخت فراگیر مینیمم G، یکی از زیردرخت‌های فراگیر G است که فقط وزن زیردرخت فراگیر مینیمم از وزن آن کمتر باشد.)** (مهندسی کامپیوتر ـ سراسری ۸۹)

۱) زیر درخت فراگیر مینیمم و دومین زیردرخت فراگیر مینیمم، هیچ یک لزوماً یکتا نیستند.
۲) زیر درخت فراگیر مینیمم و دومین زیردرخت فراگیر مینیمم، هر دو یکتا هستند.
۳) زیر درخت فراگیر مینیمم لزوماً یکتا نیست ، اما دومین زیردرخت فراگیر مینیمم یکتا است.
۴) زیر درخت فراگیر مینیمم یکتا است ، اما دومین زیردرخت فراگیر مینیمم لزوماً یکتا نیست.

**۳۱ـ مجموع وزن یال‌های درخت پوشای کمینه (MST) گراف مقابل چیست؟** (مهندسی IT ـ سراسری ۸۹)




۱) 97
۲) 91
۳) 89
۴) 85

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۲۳ـ چند تا از گزاره‌های زیر درست هستند؟**

I. اگر در یک گراف وزن‌دار بدون جهت، یال $e=(u,v)$ در درخت فراگیر کمینه باشد، حتماً دو رأس وجود دارد که کوتاه‌ترین مسیر بین آن دو از $e$ می‌گذرد.

II. مسئله‌ی یافتن همه‌ی کوتاه‌ترین مسیرها از یک رأس به بقیه‌ی رأس‌ها را در یک گراف وزن‌دار بدون جهت و هم بند با مجموعه‌ی یال‌های $E$ را می‌توان در $O(|E|)$ و نه در $O(|E|+|V|)$ یافت.

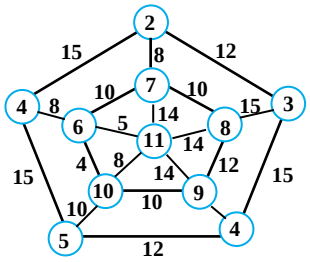
III. در یک گراف وزن دار و ساده بدون جهت با وزن‌های نامساوی، یال با سبک‌ترین وزن و یال دومین سبک‌ترین وزن هر دو در درخت فراگیر کمینه هستند.

(مهندسی کامپیوتر ـ سراسری ۹۰)

۱) 0 ۲) 1 ۳) 2 ۴) 3

**۳۳ـ اگر از الگوریتم Kruskal برای یافتن درخت پوشای کمینه (MST) گراف زیر استفاده کنیم، یال (6,11) در چه مرحله‌ای انتخاب می‌شود؟**

(مهندسی IT ـ سراسری ۹۰)




۱) مرحله ۱

۲) مرحله ۲

۳) مرحله ۳

۴) مرحله ۴

**۳۴ـ در مسئله پیدا کردن درخت پوشای مینیمم، اگر وزن یک یال از مجموع وزن یال‌های دیگر گراف بیشتر باشد، آنگاه در چه صورت این یال به عنوان یکی از یال‌های درخت پوشای مینیمم انتخاب نمی‌شود؟**

(علوم کامپیوتر ـ سراسری ۹۰)

۱) اگر آن یال، یک یال برشی باشد.

۲) در هر حالتی آن یال انتخاب نمی‌شود.

۳) اگر درجهٔ یکی از رئوسی که بر آن یال واقع شده‌اند برابر یک باشد.

۴) اگر یک دور در گراف شامل آن یال وجود داشته باشد.

**۳۵ـ در الگوریتم کروسکال برای یافتن درخت پوشای مینیمم، اگر گراف ورودی دارای یال‌هایی با وزن تکراری باشد و از الگوریتم‌های مرتب‌سازی متفاوت استفاده نماید، آنگاه کدام یک از جملات زیر صحیح است؟**

(علوم کامپیوتر ـ سراسری ۹۰)

۱) این الگوریتم می‌تواند درخت‌های پوشای مینیمم متفاوت تولید کند.

۲) این الگوریتم نمی‌تواند درخت‌های پوشای مینیمم متفاوت تولید کند.

۳) این الگوریتم همیشه درخت‌های پوشای مینیمم متفاوت تولید می‌کند و ارتباطی با الگوریتم مرتب‌سازی مورد استفاده در آن ندارد.

۴) این الگوریتم برای گراف‌های با وزن تکراری کار نمی‌کند.

**۳۶ـ یک گراف کامل بدون جهت وزن‌دار با $n$ رأس مفروض است. برای محاسبه‌ی درخت پوشای مینیمم در این گراف بیش‌ترین تعداد یال‌هایی که می‌تواند توسط الگوریتم کروسکال مورد بررسی قرار گیرد کدام مورد است؟**

(مهندسی کامپیوتر ـ آزاد ۹۰)

۱) $\frac{n(n-1)}{2}-(n-2)$ ۲) $\frac{n(n-1)}{2}$ ۳) $\frac{n(n-1)}{2}-(n-1)$ ۴) $\frac{n(n-1)}{2}-n$

**۳۷ـ کدام‌یک از الگوریتم های زیر حریصانه نیست؟**

(مهندسی نرم‌افزار ـ دکتری ۹۱)

۱) دایکسترا ۲) کدگذاری هافمن ۳) کروسکال ۴) فلوید

**«یافتن کوتاه‌ترین مسیر»**

**۳۸ـ کدام عبارت در مورد الگوریتم دایکسترا بر روی گراف وزن‌دار $G=(V,E)$ نادرست است؟ (گراف $G=(V,E)$ وزن‌دار است.)**

(مهندسی کامپیوتر ـ سراسری ۸۱)

۱) پیچیدگی الگوریتم می‌تواند $O(|E|\log|V|)$ باشد.

۲) اگر وزن هیچ یالی منفی نباشد، الگوریتم درست کار می‌کند.

۳) تنها اگر گراف دارای دور با وزن منفی نباشد الگوریتم درست کار می‌کند.

۴) هم برای گراف‌های بدون جهت درست کار می‌کند و هم برای گراف‌های جهت‌دار.

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۳۹ـ کدامیک از گزینه‌های زیر در یک گراف $G = (V, E)$ درست است؟** (مهندسی کامپیوتر ـ سراسری ۸۲)

۱) درخت‌های عمق اول و سطح اول $G$ با هم برابرند.

۲) وجود دور با وزن منفی در $G$ را می‌توان در $O(|V||E|)$ تشخیص داد.

۳) اگر وزن هر یال را به اندازه 1 واحد اضافه کنیم، کوتاه‌ترین مسیر بین هر دو رأس دلخواه تغییر نمی‌کند.

۴) اگر $M$ درخت پوشای کمینه باشد، مسیری که در $M$ بین هر دو رأس $u$ و $v$ وجود دارد کوتاهترین مسیر بین آن دو رأس در $G$ است.

**۴۰ـ در الگوریتم فلوید برای پیداکردن طول کوتاهترین مسیرها بین هر جفت گره از یک گراف جهت‌دار n گره‌ای، از روش پویا استفاده می‌شود و در هر خانه P[i, j] از ماتریسی به نام D طول کوتاهترین مسیرها بین گره‌های i تا j ثبت می‌شود. همچنین ماتریسی به نام P تشکیل می‌شود. در این ماتریس مقدار P[i, j] برابر گره‌ای است که در مسیر بهینه از i به j باید از آن عبور کرد. فرض کنیم که برای گرافی با 4 گره ماتریس P به شکل زیر باشد. کدام‌یک از گزینه‌های زیر مسیر بهینه برای رفتن از گره 1 به 3 را مشخص می‌کند؟** (مهندسی کامپیوتر ـ سراسری ۸۳)

$$P = \begin{bmatrix} 0 & 0 & 4 & 2 \\ 4 & 0 & 4 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}$$

۱) 1 4 3

۲) 1 2 3

۳) 1 4 2 3

۴) 1 2 4 3

**۴۱ـ در یک گراف جهت‌دار $G(V, E)$ وزن همه یال‌ها برابر است. می‌خواهیم طول کوتاه‌ترین مسیرها از یک رأس به نام S را تا بقیه رأس‌های G به دست آوریم. کدام‌یک از گزینه‌های زیر مرتبه یک الگوریتم کارا برای حل این مسأله است؟** (مهندسی کامپیوتر ـ سراسری ۸۳)

۱) $O(|V|^2)$ ۲) $O(|V||E|)$ ۳) $O(|V|\log|E|)$ ۴) $O(|V|+|E|)$

**۴۲ـ در مورد الگوریتم «دایکسترا» بر روی گرافی با E یال و V رأس کدام یک از گزینه‌های زیر نادرست است؟** (مهندسی کامپیوتر ـ سراسری ۸۳)

۱) پیچیدگی آن $O(E \lg V)$ است.

۲) الگوریتم اگر وزن هیچ یالی منفی نباشد درست کار می‌کند.

۳) الگوریتم فقط اگر گراف دور با وزن منفی نداشته باشد درست کار نمی‌کند.

۴) هم برای گراف‌های جهت‌دار و هم برای گرافهای بدون جهت درست کار می‌کند.

**۴۳ـ فرض کنید وزن تمام یال‌ها در یک گراف بدون جهت برابر با یک است. با اجرای یک الگوریتم کوتاهترین مسیر در این گراف، فاصله هر رأس تا رأس مبدأ (a) در جدول زیر به دست آمده است، درجه رأس f چند است؟** (علوم کامپیوتر ـ سراسری ۸۳)

| a | b | c | d | e | f | g | h | i | j |
|---|---|---|---|---|---|---|---|---|---|
| 0 | 2 | 1 | 5 | 2 | 4 | 1 | 2 | 3 | 5 |

۱) 2 ۲) 3 ۳) 4 ۴) 5

**۴۴ـ الگوریتم فلوید کوتاهترین مسیر بین همه‌ی زوج نقطه‌ها را در گراف جهت‌دار و وزن‌دار G محاسبه می‌کند. اگر یال‌های با وزن منفی هم داشته باشیم، کدام‌یک از گزینه‌های زیر بهترین توصیف این الگوریتم است؟** (مهندسی کامپیوتر ـ سراسری ۸۴)

۱) با این الگوریتم می‌توان وجود یا عدم وجود دور منفی را تشخیص داد.

۲) الگوریتم برای یال‌های منفی، ولی بدون دور منفی، ممکن است در دور بیفتد.

۳) الگوریتم برای یال‌های منفی، ولی بدون دور منفی، متوقف می‌شود. ولی درست کار نمی‌کند.

۴) الگوریتم با یال‌های منفی ولی بدون دور منفی، متوقف می‌شود. ولی جواب آن همیشه غلط است.

**۴۵ـ در مورد گراف‌های جهت‌دار و وزن‌دار با وزن مثبت و منفی ولی بدون دور منفی کدام یک از گزینه‌های زیر درست است؟** (مهندسی کامپیوتر ـ سراسری ۸۴)

۱) الگوریتم بلمن فورد همیشه درست کار می‌کند.

۲) الگوریتم دایکسترا همیشه درست کار می‌کند.

۳) الگوریتم فلوید ممکن است در دور بیفتد.

۴) الگوریتم کروسکال ممکن است درست کار نکند.

**۴۶ـ در یک گراف جهت دار بدون دور و وزن‌دار می‌خواهیم طولانی‌ترین مسیر از یک رأس مشخص شده به بقیهٔ رأس‌ها را بدست آوریم. این کار را در چه مرتبه‌ای می‌توان انجام داد؟** (مهندسی کامپیوتر ـ سراسری ۸۴)

۱) $O(|E||V|)$ ۲) $O(|E|+|V|)$

۳) $O(|V|^3)$ ۴) این کار را در زمان چند جمله‌ای نمی‌توان انجام داد.

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۴۷ـ یک مسیر ساده در گراف مسیری است که از هیچ رأسی بیش از یک بار عبور نکند. مسئله پیدا کردن طولانی‌ترین مسیر ساده برای گراف بدون وزن و جهت‌دار G را در نظر می‌گیریم. برای حل این مسئله:** (مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۴)

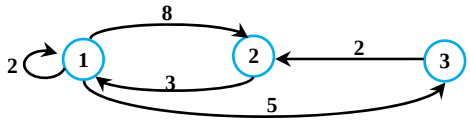
۱) می‌توان از روش برنامه‌ریزی پویا استفاده کرد چون اصل بهینگی برقرار است.

۲) با وجودی که اصل بهینگی برقرار است نمی‌توان از روش برنامه‌ریزی پویا استفاده کرد.

۳) نمی‌توان از روش برنامه‌ریزی پویا استفاده کرد چون اصل بهینگی برقرار نیست.

۴) با وجودی که اصل بهینگی برقرار نیست می‌توان از روش برنامه‌نویسی پویا استفاده کرد.

**۴۸ـ با استفاده از الگوریتم فلوید کوتاه‌ترین مسیر بین گره‌های ذیل عبارتند از:** (مهندسی کامپیوتر ـ آزاد ۸۴)




۱)

| | 1 | 2 | 3 |
|---|---|---|---|
| 1 | 0 | 5 | 7 |
| 2 | 3 | 0 | 8 |
| 3 | 5 | 10 | 0 |

۲)

| | 1 | 2 | 3 |
|---|---|---|---|
| 1 | 0 | 7 | 5 |
| 2 | 3 | 0 | 8 |
| 3 | 5 | 2 | 0 |

۳)

| | 1 | 2 | 3 |
|---|---|---|---|
| 1 | 2 | 7 | 5 |
| 2 | 3 | 0 | 8 |
| 3 | 5 | 2 | 0 |

۴)

| | 1 | 2 | 3 |
|---|---|---|---|
| 1 | 0 | 8 | 5 |
| 2 | 3 | 0 | 8 |
| 3 | 5 | 2 | 0 |

**۴۹ـ در یک گراف N گره با شماره‌های 1 تا N موجود است. از هر گره i به تمامی گره‌هایی که شماره آن از i بزرگتر است یک یال وجود دارد. تعداد مسیرهای ممکن از گره 1 به N را محاسبه کنید.** (مهندسی کامپیوتر ـ سراسری ۸۶)

۱) $N-1$ ۲) $N$ ۳) $2^{n-2}$ ۴) $\sum_{i=1}^{N} i$

**۵۰ـ شما باید الگوریتمی طراحی کنید که «مسیر ساده با کمترین تعداد یال» را بین دو رأس مفروض در یک گراف بیابد. شما الگوریتم خود را براساس کدام یک از الگوریتم‌های زیر ارائه می‌کنید؟** (مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۶)

۱) الگوریتم Kruskal ۲) الگوریتم Dijkstra ۳) جستجوی عمق اول (DFS) ۴) جستجوی سطح اول (BFS)

**۵۱ـ فرض کنید G یک گراف وزندار، همبند و غیرجهتدار باشد که وزن یال‌های آن اعداد صحیح است. در مورد اعمال کردن الگوریتم Floyd – Warshall روی گراف G چه می‌توان گفت؟** (مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۶)

۱) چون G جهتدار نیست نمی‌توان این الگوریتم را استفاده کرد.

۲) چون گراف G غیرجهتدار است الگوریتم قادر نیست که حلقه‌های منفی را تشخیص دهد.

۳) چون گراف G غیرجهتدار است تابع زمان اجرای این الگوریتم روی G رشد کمتری نسبت به گراف‌های جهتدار دارد.

۴) هیچ‌کدام

**۵۲ـ در گراف G = (V,E) اگر وزن بعضی از یال‌ها منفی باشد برای پیدا کردن کوتاه‌ترین مسیر بین دو رأس s و t به تمام وزن یال‌ها مقدار ثابت c اضافه کنیم به طوری که وزن تمام یال‌های منفی، مثبت شود در آن صورت الگوریتم Dijkstra .....** (علوم کامپیوتر ـ سراسری ۸۶)

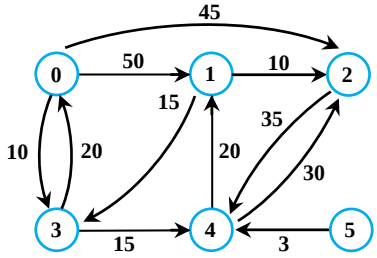
۱) کوتاه‌ترین مسیر بین t,s را محاسبه می‌کند.

۲) کوتاه‌ترین مسیر بین t,s را محاسبه نمی‌کند.

۳) کوتاه‌ترین مسیر بین t,s را محاسبه می‌کند فقط در صورتی که گراف مذکور G دارای دور منفی باشد.

۴) فقط در صورتی که گراف مذکور G دور منفی نداشته باشد کوتاه‌ترین مسیر بین t,s حاصل می‌شود.

**۵۳ـ در گراف زیر کوتاه‌ترین مسیر از گره 0 به سایر گره‌ها با استفاده از الگوریتم کوتاه‌ترین مسیر چیست؟** (مهندسی کامپیوتر ـ آزاد ۸۶)




۱) $[45,45,10,25,\infty]$

۲) $[50,45,10,25,\infty]$

۳) $[50,60,10,25,\infty]$

۴) $[45,60,10,25,\infty]$

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۵۴ـ فرض کنید گراف G=(E,V) وزن‌دار و جهت‌دار است به طوری که وزن یال‌ها $w : E \to R$ است. همان گراف را با وزن‌های جدید که با تابع $w' : e \to E$ نشان می‌دهیم عوض می‌کنیم و آن را G'' می‌نامیم به طوری که $w'(u,v) = w(u,v) - out_degree(u) + in_degree(u)$ در آن صورت:** (مهندسی کامپیوتر ـ سراسری ۸۷)

۱) وزن یک دور در G با وزن همان دور G'' برابر است.

۲) وزن یک درخت فراگیر کمینه در G با وزن همان درخت در G'' برابر است.

۳) گراف G'' دور با وزن منفی دارد اگر و فقط اگر گراف G دوری داشته باشد که تعداد یالِ آن از وزنِ آن کم‌تر باشد.

۴) گراف G دور با وزن منفی دارد اگر و فقط اگر گراف G'' دوری داشته باشد که تعداد یال آن از وزن آن کم‌تر باشد.

**۵۵ـ یک گراف جهت‌دار و وزن‌دار G(V,E) داده شده است. وزن هر یال یک عدد صحیح صفر یا مثبت است. دو عمل زیر را بر روی یک رأس $v \in V$ تعریف می‌کنیم.**

**عمل A: اگر وزن تک تک یال‌هایی که به v ختم می‌شود مثبت باشد، از وزن هر یک 1 واحد کم می‌کنیم و به وزن هر یک از یال‌های خروجی آن، 1 واحد اضافه می‌کنیم.**

**عمل B: اگر وزن تک‌تک یال‌های خروجی از v مثبت باشد، از وزن هر یک از آنها 1 واحد کم می‌کنیم و به وزن هر یک از یال‌های ورودی به آن، 1 واحد اضافه می‌کنیم.**

**گراف G را می‌گیریم و یکی از دو عمل فوق را روی یک رأس آن در صورت امکان انجام می‌دهیم تا گراف $G_1$ به دست آید. همین کار را بر روی گراف $G_1$ انجام می‌دهیم و بر روی گراف حاصل از هر مرحله هم این کار را چند بار تکرار می‌کنیم تا گراف G'' به دست آید.**

**کدام‌یک از گزینه‌های زیر در مورد گراف‌های G و G'' درست است؟** (مهندسی کامپیوتر ـ سراسری ۸۷)

۱) تعداد یال‌های با وزن مثبت تغییر نمی‌کند.

۲) مجموع وزن یال‌های هر دور تغییر نمی‌کند.

۳) مجموع کل وزن یال‌های گراف تغییر نمی‌کند.

۴) اگر گراف دور نداشته باشد، تعداد یال‌های با وزن صفر تغییر نمی‌کند.

**۵۶ـ چند تا از گزاره‌های زیر در مورد یک گراف G = (V,E) صحیح است؟** (مهندسی کامپیوتر ـ سراسری ۸۷)

**ـ BFS (Breadth-First-Search) از پشته استفاده می‌کند.**

**ـ درخت فراگیر کمینه (MST) را می‌توان در $O(|E|\lg|V|)$ ساخت.**

**ـ یک درخت فراگیر کمینه ممکن است شامل یال با بیشترین وزن باشد.**

۱) 0 ۲) 1 ۳) 2 ۴) 3

**۵۷ـ مسئله‌ی پیدا کردن همه‌ی طولانی‌ترین مسیرها از یک رأس داده شده در یک گراف DAG (جهت‌دار و بدون دور) را در چه زمانی می‌توان حل کرد؟** (مهندسی کامپیوتر ـ سراسری ۸۸)

۱) $O(|V||E|)$ ۲) $O(|V|+|E|)$ ۳) $O(|V|\log E)$ ۴) راه‌حل چندجمله‌ای ندارد.

**۵۸ـ برای ۴ الگوریتم زیر برای گراف با n رأس و O(n) یال، ۶ زمان اجرا به ترتیب از A تا F داده شده است:**

**۱ـ دایکسترا، ۲ـ پریم، ۳ـ بلمن فورد، ۴ـ همه‌ی کوتاه‌ترین مسیرها از یک رأس در یک DAG (گراف جهت‌دار و بدون دور)**

**A)$O(\log n)$, B)$O(n)$ C)$O(n\log n)$, D)$O(n^2)$ E)$O(n^2\log n)$, F)$O(n^3)$**

**بهترین گزینه برای زمان اجرای هر الگوریتم کدام یک می‌باشد؟** (مهندسی کامپیوتر ـ سراسری ۸۸)

۱) 1:D,2:C,3:F,4:C ۲) 1:C,2:D,3:F,4:F ۳) 1:D,2:D,3:D,4:B ۴) 1:C,2:C,3:D,4:B

**۵۹ـ چندتا از عبارت‌های زیر صحیح هستند؟** (مهندسی کامپیوتر ـ سراسری ۸۸)

**ـ BFS از پشته استفاده می‌کند.**

**ـ یک درخت پوشای کمینه ممکن است شامل یال با بیشترین وزن باشد.**

**ـ درخت پوشای بیشینه را می‌توان در $O(E\log V)$ ساخت.**

**ـ درخت عمق اول و سطح اول یگ گراف مانند هم‌اند.**

۱) صفر ۲) 1 ۳) 2 ۴) 3

**۶۰ـ فرض کنید که A ماتریس مجاورت گراف G باشد، کدام ماتریس تمامی زوج‌های مرتبی از گره‌ها را نشان می‌دهند که به طول دقیقاً k به هم متصل شده‌اند؟ (توجه: V نشان‌دهنده جمع دودویی دو ماتریس است.)** (مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۸)

۱) $A^{k+1}$ ۲) $A^k$ ۳) $A^{k-1}$ ۴) $\bigvee_{j=0}^{k} A^k$

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۶۱ـ برای پیدا کردن کوتاهترین مسیر از یک رأس به تمام رئوس در یک گراف، اگر وزن همه یال‌ها برابر باشد کدام الگوریتم مناسب‌تر است؟**

(مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۹)

۱) BFS ۲) DFS ۳) Dijkstra ۴) Floyd Warshall

**۶۲ـ در یک گراف با وزن‌های صحیح بزرگ‌تر از 1، فرض کنید که وزن هر یال را یک واحد زیاد کنیم. در آن صورت چندتا از گزاره‌های زیر درست‌اند؟**

(مهندسی کامپیوتر ـ سراسری ۹۰)

**I.** برش کمینه‌ی **(S,T)** در هر دو گراف یکی است. **II.** درخت فراگیر کمینه‌ی هر دو گراف یکی است.

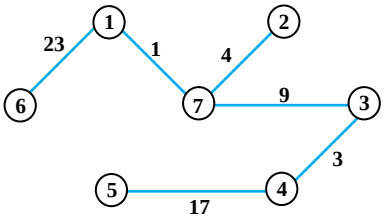
**III.** کوتاهترین مسیر بین دو رأس مشخص در دو گراف شامل یال‌های یکسان هستند.

۱) 0 ۲) 1 ۳) 2 ۴) 3

## پاسخنامه تست‌های طبقه‌بندی شده کنکوری فصل سیزدهم

**«یافتن MST»**

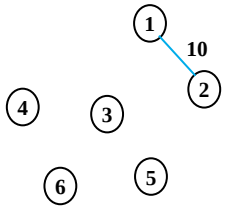
**۱ـ گزینه «۴»** با استفاده از الگوریتم راشال درخت پوشای مینیمم را به دست آورده و سپس مجموع یال‌های این درخت را محاسبه می‌کنیم.



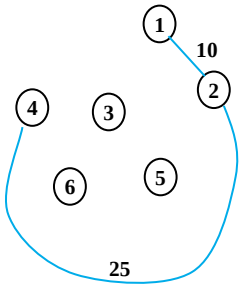

$$1+3+4+9+17+23=57$$

**۲ـ گزینه «۴»** منظور از Spanning Tree همان درخت پوشا است.

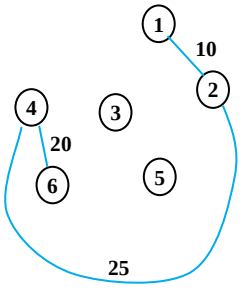
**۳ـ گزینه «۴»** با توجه به الگوریتم پریم (prim)، درخت پوشای مینیمم تا انتهای مرحله سوم به صورت زیر است:




(الف)



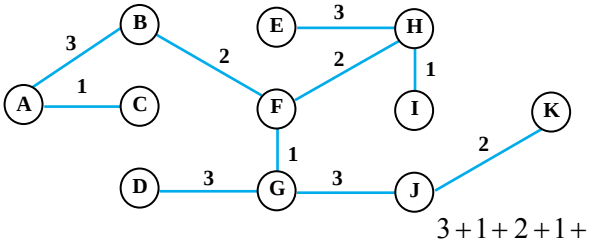

(ب)




(ج)

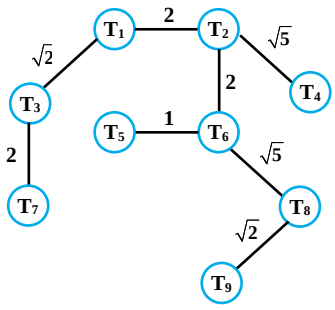
**۴ـ گزینه «۴»**

با استفاده از الگوریتم راشال یا پریم درخت پوشای مینیمم به صورت مقابل می‌باشد.




حاصل‌جمع وزن یال‌های درخت پوشا عبارت است از: $3+1+2+1+3+3+2+2+3+1=21$

**۵ـ گزینه «۴»** این مسئله در حقیقت یافتن درخت پوشای کمینه است و اگر برای این کار از روش کروسکال استفاده کنیم درخت حاصل به شکل زیر خواهد بود.




$$\text{وزن درخت} = 7+2\sqrt{2}+2\sqrt{5}$$

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۶ـ گزینه «۳»** زمان اجرای کروسکال $O(e\log e)$ است، اما زمان پریم $O(n^2)$ می‌باشد. بنابراین گزینه اول درست نمی‌باشد. گزینهٔ دوم به این دلیل غلط است که در الگوریتم پریم در حالتی که چند یال با وزن یکسان داشته باشیم ممکن است درخت‌های متفاوتی ایجاد گردد و گزینهٔ چهارم نیز در مورد پریم غلط است.

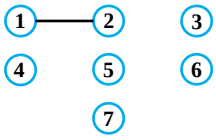
**۷ـ گزینه «۲»** در الگوریتم کروسکال $n-1$ یال با کمترین وزن که ایجاد دور نمی‌کنند انتخاب می‌گردند (البته گزینه ۱ نیز می‌تواند درست باشد).

**۸ـ گزینه «۲»** دقت کنید که مسیر بین هر جفت رأس در درخت فراگیر مینیمم برای یک گراف بی‌جهت لزوما کوتاهترین مسیر نمی‌باشد. (سایر گزینه‌ها نادرست می‌باشند.)

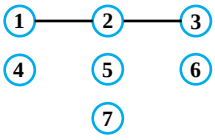
**۹ـ گزینه «۴»** این گزینه با برنامه‌سازی پویا به طور مؤثری پیاده‌سازی می‌شود.

**۱۰ـ گزینه «۳»** در الگوریتم کروسکال در هر مرحله یال با کمترین وزن با شروع از سبک‌ترین یال انتخاب می‌گردد، بنابراین ترتیب انتخاب یال‌ها به صورت مقابل خواهد بود: $(1,2),(2,3)(4,5)(6,7)$

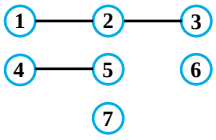
مراحل محاسبه درخت MST به صورت زیر خواهد بود:



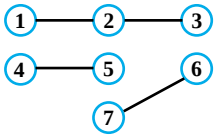

مرحله اول



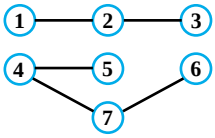

مرحله دوم



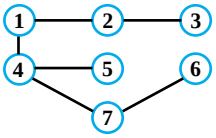

مرحله سوم




مرحله چهارم




مرحله پنجم




مرحله ششم

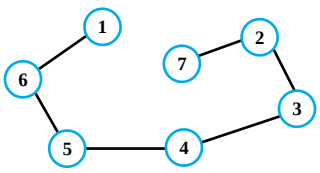
**۱۱ـ گزینه «۴»** اگر وزن تمام یال‌ها برابر باشد، آنگاه درخت پوشای BFS همان درخت MST خواهد بود که زمان محاسبه درخت BFS از مرتبه $O(|V|+|E|)$ می‌باشد و با توجه به این که گراف همبند است بنابراین، $|E|\geq|V|-1$ و مرتبه الگوریتم $O(|E|)$ خواهد بود.

**۱۲ـ گزینه «۳»** اگر وزن‌های موجود در یک دور تکراری باشند، آنگاه این دو الگوریتم ممکن است درخت‌های متفاوتی ایجاد نمایند، همچنین درخت ایجاد شده توسط روش prim می‌تواند به رأس شروع بستگی داشته باشد. اما در صورتی که وزن یال‌های گراف متمایز باشد، آنگاه درخت‌های ایجاد شده یکسان خواهند بود.

**۱۳ـ گزینه «۲»** زیر گراف ایجاد شده توسط الگوریتم prim در هر مرحله همبند است، اما زیر گراف ایجاد شده توسط الگوریتم kruskal در مراحل میانی لزوماً یک درخت نیست و ممکن است یک جنگل را تشکیل دهد.

همچنین سرعت الگوریتم prim اصولاً بهتر از الگوریتم kruskal است (به جز در گراف‌های اسپارس).

**۱۴ـ گزینه «۳»** با هریک از الگوریتم‌های تعیین MST می‌توان به‌سادگی به گزینه سوم رسید. به عنوان مثال اگر از الگوریتم کروسکال استفاده کنیم، ابتدا یال 2 به 3 و سپس یال 1 به 6 انتخاب می‌شود و با ادامه همین روند شکل مقابل حاصل می‌گردد.




**۱۵ـ گزینه «۱»** در زمان $O(V)$ می‌توان بررسی کرد که یال e متعلق به درخت پوشا می‌باشد یا نه، اگر متعلق به گراف پوشا نباشد در زمان $O(V)$ در صورت نیاز می‌توان آن را به درخت MST اضافه نمود.

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۱۶ـ گزینه «۲»** دقت کنیم که اگر تمام یالهایی که کاهش می‌یابند متعلق به T باشند، آنگاه درخت پوشای کمینه در گراف جدید همان T خواهد بود؛ ولی اگر حتی یکی از یالهایی که کاهش می‌یابند متعلق به T نباشد، آنگاه ممکن است T برای گراف جدید، دیگر یک MST نباشد.

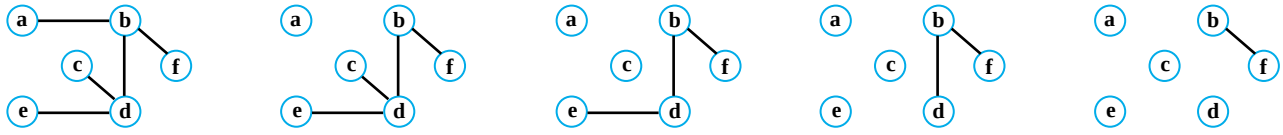
**۱۷ـ گزینه «۱»** در الگوریتم کروسکال، ابتدا یالها بر اساس ترتیب وزن مرتب می‌شوند و سپس با شروع از ابتدای لیست مورد نظر $n-1$ یال (که ایجاد دور نمی‌کنند) انتخاب خواهند شد. بنابراین ترتیب انتخاب یالها به صورت زیر می‌باشد:

(با توجه به این‌که گراف بدون جهت است، یال $(d,e)=(e,d)$ بنابراین گزینه ۱ درست است.)

$$\overset{1}{(b,f)},\overset{2}{(b,d)},(f,d),\overset{3}{(d,e)},\overset{4}{(c,d)},(c,e),\overset{5}{(a,b)},(a,e),(b,c),(a,c)$$

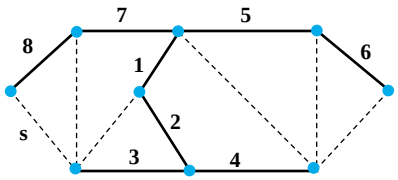
پنج یالی که انتخاب می‌گردند با اعداد 1 تا 5 مشخص شده‌اند.

بنابراین نحوه ساختن MST به صورت زیر خواهد بود:




مرحله اول　مرحله دوم　مرحله سوم　مرحله چهارم　مرحله پنجم

**۱۸ـ گزینه «۳»** ترتیب انتخاب یالها و روش prim با اعداد 1 تا 8 در گراف روبرو نشان داده شده است:



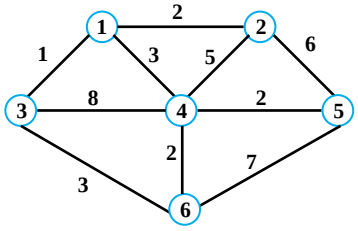

**۱۹ـ گزینه «۲»** اگر e را حذف کنیم در ادامه، MST را به دست آورده و سپس e را اضافه نماییم، آنگاه در صورتی که با حذف e گراف همبند باقی مانده باشد حتماً با اضافه نمودن آن به MST دور ایجاد خواهد شد. بنابراین گزینه اول نادرست است.

همچنین اگر تمام یال‌های متصل به u,v را حذف نماییم، آنگاه با اضافه نمودن e باز هم گراف ناهمبند باقی می‌ماند. بنابراین گزینه سوم نیز نادرست است.

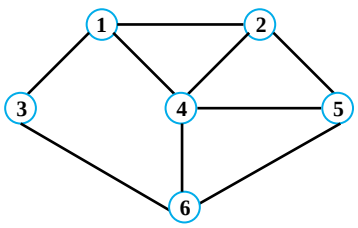
**۲۰ـ گزینه «۴»** اگر وزن برخی از یال‌ها منفی باشد، گزاره مورد نظر نادرست خواهد بود. اما در صورتی که به ازای هر یال $w_e > 0$ باشد گزاره مورد نظر صحیح است.

**۲۱ـ گزینه «۴»** در حالت کلی زمان الگوریتم kruskal برابر $O(E\log E)$ می‌باشد و با توجه به این‌که گراف پراکنده است، بنابراین $E \approx |V| = n$ و در نتیجه مرتبه زمانی الگوریتم $O(n\log n)$ خواهد بود.

**۲۲ـ گزینه «۳»** یکی از الگوریتم‌های محاسبه MST روش بیان شده می‌باشد. به عنوان مثال گراف زیر را در نظر بگیرید:




دور دلخواه $V_1,V_3,V_4$ را در نظر بگیرید. یال با بیشترین وزن آن یعنی، یال $(V_3,V_4)$ را حذف می‌کنیم داریم:

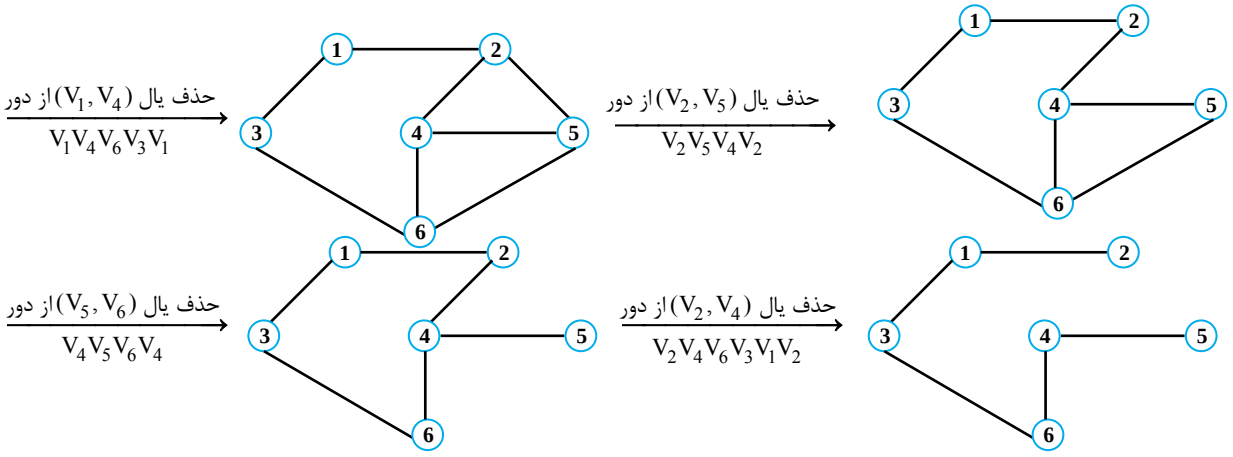



www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



حال می‌توان دورهای دیگر موجود در گراف را به همین ترتیب از بین برد:




---

**۲۳ـ گزینه «۳»** هدف یافتن MST در گراف مورد نظر می‌باشد و در مورد گراف داده شده که یک گراف اسپارس است بهترین الگوریتم kruskal خواهد بود. دقت نمایید که تعداد یال‌های گراف چگال و گراف اسپارس در حالت کلی در روابط زیر صادق می‌باشند:

گراف چگال : $|E| \approx |V|^2$ گراف اسپارس : $|E| \approx \alpha |V|$

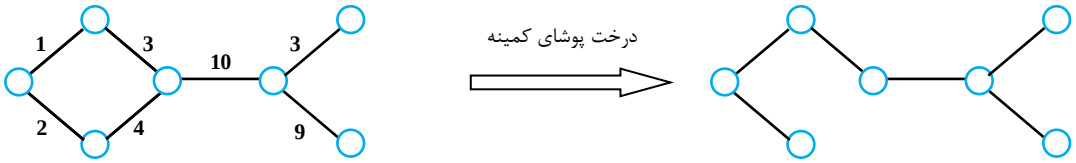
---

**۲۴ـ گزینه «۲»** در صورتی که وزن یال‌های گراف متمایز باشند و یا گراف دارای دور نباشد، درخت پوشای کمینه منحصر به فرد است.

---

**۲۵ـ گزینه «۲»** برای یافتن درخت پوشای یک گراف تک دوره کافیست در لیست مجاورتی آن یال با سنگین‌ترین وزن موجود در دور را حذف نمود.

به عنوان مثال گراف زیر را در نظر بگیرید.




همان‌گونه که مشخص می‌باشد برای به دست آوردن درخت MST کافیست یال با وزن 4 که بزرگترین یال موجود در دور می‌باشد حذف شود (دقت نمایید که حذف بزرگترین یال، (در اینجا یال با وزن 10) در حالت کلی درست نیست.)

---

**۲۶ـ گزینه «۴»** مجموعه یال‌های انتخاب شده توسط الگوریتم prim برابر $E = \{(S,C),(S,A),(A,B)\}$ می‌باشد، بنابراین یال با کمترین وزنی که از این مجموعه عبور می‌کند یال $(B,E)$ با وزن 5 می‌باشد.

---

**۲۷ـ گزینه «۳»** اگر بتوان دوری شامل یال e یافت که یال e از سایر یال‌های آن دور سنگین‌تر باشد، این یال در درخت پوشای کمینه قرار نخواهد داشت. ولی اگر چند یال دیگر در هر دور نیز هم‌وزن با e باشند، ممکن است این یال در درخت پوشای کمینه قرار گیرد.

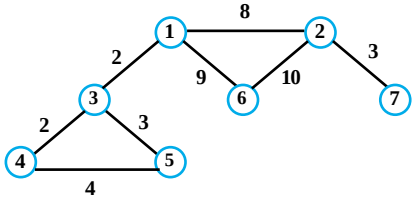
---

**۲۸ـ گزینه «۴»** هر سه گزینه اول کاربردهای DFS می‌باشند اما گزینه چهارم در حالت کلی دارای زمان $O(E \log V)$ می‌باشد.

---

**۲۹ـ گزینه «۲»** زیرا اگر یک درخت MST یک درخت فراگیر گلوگاه نباشد، آنگاه می‌توان با تعویض بزرگترین یال آن با یک یال دیگر به یک درخت فراگیر با وزن کمتر رسید که با MST بودن درخت اولیه در تناقض است. در مورد گزینه‌های سوم و چهارم دقت نمایید که وزن سنگین‌ترین یال در درخت فراگیر بیشینه بیشترین وزن ممکن در بین تمام یال‌های گراف می‌باشد که با تعریف درخت فراگیر گلوگاه در تناقض است.

به گراف زیر دقت نمایید:

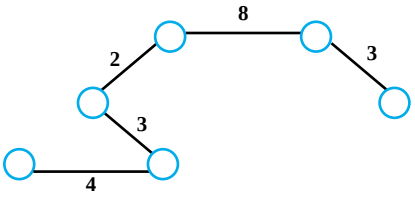


www.modaresanesharif.ac.ir

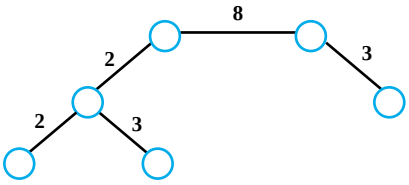
www.modaresanesharif.ac.ir



در این حالت درخت زیر یک درخت فراگیر گلوگاه است.

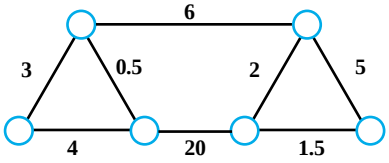



اما درخت فراگیر کمینه به صورت زیر می‌باشد.




---

**۳۰ـ گزینه «۴»** اگر وزن یال‌های گراف متمایز باشند، آنگاه درخت پوشای کمینه منحصر به فرد خواهد بود؛ اما برای دومین درخت پوشای کمینه لزوماً این رابطه برقرار نیست. به عنوان مثال گراف روبرو را در نظر بگیرید:

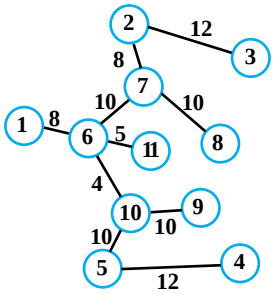



هر دو گراف زیر، دومین درخت‌های فراگیر می‌باشند:

---

**۳۱ـ گزینه «۳»** با استفاده از الگوریتم کروسکال، ابتدا یالهای گراف را براساس وزن به صورت صعودی مرتب می‌کنیم، سپس از کوچکترین وزن شروع کرده و به تعداد $11-1=10$ یال را انتخاب می‌کنیم.

البته به شرطی که انتخاب یال مورد نظر ایجاد دور نکند.




$(6,10)=4$
$(6,11)=5$
$(1,6)=8$
$(10,11)=8$
$(2,7)=8$

$(6,7)=10$
$(7,8)=10$
$(10,9)=10$
$(10,5)=10$
$(2,3)=12$

$(5,4)=12$
$(9,8)=12$
$\vdots$

با توجه به درخت پوشای بدست آمده مجموع وزن یال‌های درخت عبارتند از:

$$4+5+8+8+10+10+10+10+12+12=89$$

---

**۳۲ـ گزینه «۳»** الف) اگر e یال برشی باشد بنابراین حداقل کوتاه‌ترین مسیر بین دو رأس از آن می‌گذرد. اگر یال برشی نباشد آنگاه حتماً در یک دور حضور دارد. حال اگر کوتاه‌ترین مسیر بین هیچ دو رأس آن دور از e عبور نکند یعنی e از بخش سایر یال‌های دور سنگین‌تر است و بنابراین سنگین‌ترین یال دور است پس نمی‌تواند در MST باشد.

رابطهٔ II نادرست است، زیرا با فرض داشتن یالهای با وزن منفی الگوریتم دارای مرتبه زمانی بیشتر از $O(/E/)$ خواهد بود.

رابطه I و III درست می‌باشند.

برای درستی رابطهٔ III می‌توان الگوریتم kruskal را در نظر گرفت. در این الگوریتم دو یال با کمترین وزن به عنوان اولین یالها انتخاب می‌گردند و با توجه به این که گراف ساده است، بنابراین دو یال موردنظر ایجاد دور نمی‌کنند و حتماً انتخاب می‌شوند.

---

www.modaresanesharif.ac.ir

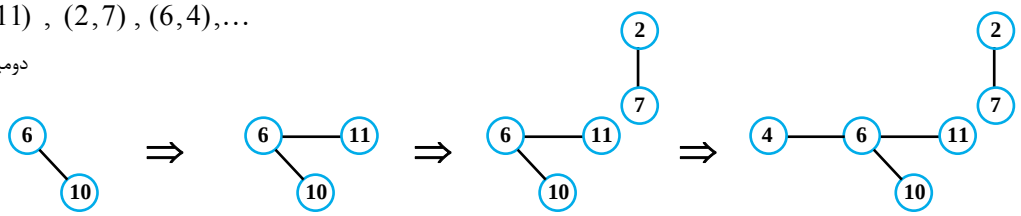
www.modaresanesharif.ac.ir



**۳۳ـ گزینه «۲»** در الگوریتم kruskal برای تعیین MST کافیست یال‌ها براساس وزن آنها و به ترتیب صعودی انتخاب شوند، به طوری‌که مجموعه انتخاب شده ایجاد دور نکند. بنابراین ترتیب انتخاب یال‌ها به صورت زیر خواهد بود.

$(6,10)\ ,\ (6,11)\ ,\ (2,7)\ ,\ (6,4),\ldots$

دومین یال




**۳۴ـ گزینه «۴»** در صورتی که یال مذکور یک یال موجود در یک دور باشد، قطعاً هنگام ساخت MST حذف می‌شود. (به متن درس مراجعه کنید)

**۳۵ـ گزینه «۱»** اگر الگوریتم‌های مرتب‌سازی متفاوت ترتیب‌های با وزن مساوی را تغییر دهند درخت‌های MST نیز تغییر می‌کند.

**۳۶ـ گزینه «۱»** به عنوان مثال در گراف $k_3$ حداکثر 2 یال و در گراف $k_4$ حداکثر 4 یال کنترل می‌شوند.

**۳۷ـ گزینه «۴»** دایکسترا، الگوریتمی است برای ایجاد درخت پوشای کمینه که حریصانه عمل می‌کند. کدگذاری هافمن کدگذاری حروفی است که طول متن ساخته شده کمینه و حریصانه باشد. کروسکال، الگوریتم یافتن کوتاه‌ترین مسیر بین دو رأس در گراف است که حریصانه می‌باشد. اما الگوریتم فلوید با استفاده از روش برنامه‌ریزی پویا عمل می‌کند.

**«یافتن کوتاه‌ترین مسیر»**

**۳۸ـ گزینه «۳»** گزینه سوم در حالت کلی در الگوریتم دایکسترا درست نمی‌باشد و در مورد الگوریتم بلمن فورد برقرار است. به نکات زیر در مورد الگوریتم دایکسترا دقت کنید:

پیچیدگی الگوریتم دایکسترا $O(|E|\log|V|)$ است.

اگر وزن هیچ یالی منفی نباشد، الگوریتم درست کار می‌کند.

اگر گراف دارای دور با وزن منفی نباشد، اما یال با وزن منفی داشته باشد باز هم ممکن است الگوریتم درست کار نکند.

هم برای گراف‌های بدون جهت درست کار می‌کند و هم برای گراف‌های جهت‌دار.

**۳۹ـ گزینه «۲»** وجود دور با وزن منفی را می‌توان در زمان $O(VE)$ توسط الگوریتم بلمن فورد تشخیص داد.

**۴۰ـ گزینه «۴»** با توجه به ماتریس مسیر داده شده می‌توان مسیر بهینه را به صورت زیر محاسبه نمود:

$P(1,3)=4 \Rightarrow$ رأس 4 به عنوان یک رأس میانی در مسیر بهینه از 1 به 3 قرار دارد. $\Rightarrow$ مسیر بهینه $V_1\ldots V_4\ldots V_3$

$P(1,4)=2 \Rightarrow$ رأس 2 به عنوان یک رأس میانی در مسیر بهینه از رأس 1 به 4 وجود دارد. $\Rightarrow$ مسیر بهینه $V_1\ldots V_2\ldots V_4\ldots V_3$

$P(1,2)=0 \Rightarrow$ مسیر بهینه از رأس 1 به رأس 2 شامل هیچ رأس میانی نیست. $\Rightarrow$ مسیر بهینه $V_1V_2\ldots V_4\ldots V_3$

$P(2,4)=0 \Rightarrow$ مسیر بهینه از $V_2$ به $V_4$ شامل هیچ رأس میانی نیست. $\Rightarrow$ مسیر بهینه $V_1V_2V_4\ldots V_3$

$P(4,3)=0 \Rightarrow$ مسیر بهینه از $V_4$ به $V_3$ شامل هیچ رأس میانی نیست. $\Rightarrow$ مسیر بهینه $V_1V_2V_4V_3$

**۴۱ـ گزینه «۴»** اگر وزن تمام یال‌ها برابر باشد با استفاده از الگوریتم BFS می‌توان کوتاهترین مسیر را از رأس S به سایر رئوس در زمان $O(|V|+|E|)$ محاسبه نمود.

**۴۲ـ گزینه «۳»** گزینهٔ اول و دوم و چهارم درست می‌باشند و شکل صحیح گزینه سوم به این صورت است که اگر گراف یالی با طول منفی نداشته باشد الگوریتم و دایکسترا درست کار می‌کند.

در مورد گزینهٔ اول نیز دقت کنید که با توجه به نوع ساختمان دادهٔ مورد استفاده زمان می‌تواند $O(e\log e)$ یا $O(V^2)$ یا $O(e\log V)$ باشد.

**۴۳ـ گزینه «۲»** برای رفتن به d و j از f عبور کرده‌ایم، از طرفی i هم به آن وارد شده است. (با توجه به این‌که فاصله f برابر 4 می‌باشد بنابراین، رأس‌هایی که فاصله آن‌ها برابر 5 است از طریق f مورد دسترسی قرار گرفته‌اند و f از طریق رأسی که فاصله آن برابر 3 می‌باشد مورد دسترسی قرار گرفته است).

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۴۴ـ گزینه «۱»** الگوریتم فلوید ـ وارشال وجود دور با وزن منفی را تشخیص می‌دهد.

---

**۴۵ـ گزینه «۱»** به موارد زیر در مورد الگوریتم‌های بلمن فورد و دایکسترا دقت نمایید:

الگوریتم دایکسترا بر روی گراف‌های جهت‌دار و بدون جهت (با وجود دور یا عدم وجود دور) در حالتی که وزن تمام یال‌ها مثبت باشد به درستی کار می‌کند.

الگوریتم دایکسترا بر روی گراف‌های با وزن منفی (حتی بدون دور منفی) همواره به درستی عمل نمی‌کند.

الگوریتم بلمن فورد در گراف‌هایی که دور با طول منفی نداشته باشد (حتی در صورت وجود یال با وزن به درستی عمل می‌کند.)

در صورتی که دور با وزن منفی وجود داشته باشد هیچ الگوریتمی نمی‌تواند بر روی گراف اجرا شود.

الگوریتم فلوید نیز مانند الگوریتم بلمن فورد برای گراف‌هایی با یال منفی (ولی بدون دور منفی) درست کار می‌کند. این الگوریتم حتی می‌تواند دور با وزن منفی را نیز تشخیص دهند.

---

**۴۶ـ گزینه «۲»** این کار را می‌توان با اندکی تغییر در الگوریتم DFS انجام داد که مرتبه زمانی آن از $O(|V|+|E|)$ می‌باشد.

---

**۴۷ـ گزینه «۳»** اصل بهینگی در مورد کوتاهترین مسیر بین دو رأس صادق است ولی، در مورد بزرگترین مسیر بین دو رأس صادق نیست و می‌دانیم که اگر اصل بهینگی در مورد یک مسئله برقرار باشد، آنگاه می‌توان از برنامه‌نویسی پویا در مورد آن استفاده کرد. اصل بهینگی در مسئله بزرگ‌ترین مسیر بین رئوس گراف برقرار نیست و بنابراین برنامه‌نویسی پویا روش مناسبی برای چنین مسئله‌ای نمی‌باشد. دقت نمایید که مسئله مطرح شده یک مسئله Np-hard است.

---

**۴۸ـ گزینه «۲»** به عنوان مثال، وزن کوتاه‌ترین مسیرها از 1 به سایر رئوس به‌صورت روبه‌رو می‌باشد:

| | 1 | 2 | 3 |
|---|---|---|---|
| 1 | 0 | 7 | 5 |

که فقط در گزینه دوم وجود دارد.

---

**۴۹ـ گزینه «۳»** اگر تعداد مسیرهای ممکن از گره 1 به N را با $T(N)$ نمایش دهیم، آنگاه واضح است که $T(2)=1$، حال با داشتن $T(N-1)$ برای محاسبه $T(N)$ یک گره با شماره N به گراف با $N-1$ رأس اضافه می‌شود و از تمام رئوس گراف قبلی یک یال به رأس N متصل می‌گردد، در این صورت به ازای هر گره همان مسیر قبلی که از 1 به گره $N-1$ منتهی می‌گردید وجود دارد. حال با استفاده از یال بین $N-1$ و N باز هم یک مسیر معین است، اما یک مسیر جدید نیز که از $N-1$ عبور نمی‌کند نیز به وجود آمده است، بنابراین:

$$T(N)=2T(N-1), T(2)=1 \Rightarrow T(N)=2^{N-2}$$

---

**۵۰ـ گزینه «۴»** الگوریتم جست‌وجوی BFS کوتاهترین مسیر را بین تمام رئوس با اولین رأس ملاقات شده بدست می‌آورد. بنابراین برای طراحی الگوریتم مورد نظر کافیست الگوریتم BFS را بر روی گراف انجام دهیم و یکی از دو رأس مورد نظر را به عنوان اولین رأس انتخاب نماییم.

---

**۵۱ـ گزینه «۴»** در حالت کلی، هیچ‌کدام از سه گزینه اول در مورد الگوریتم فلوید ـ وارشال درست نمی‌باشد.

---

**۵۲ـ گزینه «۲»** الگوریتم دایکسترا در شرایط کلی‌تر هم این کار را نمی‌کند.

---

**۵۳ـ گزینه «۱»** کوتاه‌ترین مسیرها از رأس 0 به سایر رئوس به‌صورت زیر می‌باشد:

کوتاه‌ترین مسیر از 0 به 1 (طول = 45)　　0,3,4,1

کوتاه‌ترین مسیر از 0 به 2 (طول = 45)　　0,2

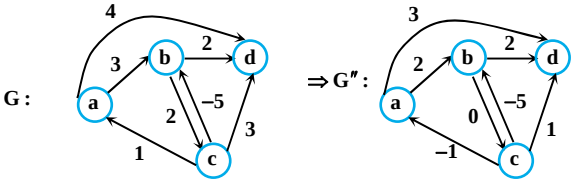
کوتاه‌ترین مسیر از 0 به 3 (طول = 10)　　0,3

کوتاه‌ترین مسیر از 0 به 4 (طول = 25)　　0,3,4

و با توجه به این‌که هیچ مسیری از 0 به 5 وجود ندارد، طول مسیر برابر بی‌نهایت درنظر گرفته می‌شود.

---

**۵۴ـ هیچ‌کدام از گزینه‌ها صحیح نیست.** گراف‌های زیر را در نظر بگیرید:

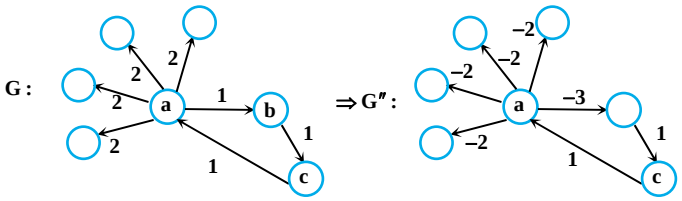



www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



وزن دور abca در گراف G برابر 1- و در گراف $G''$ برابر 4- است. وزن درخت MST در این دو گراف نیز متفاوت است. دو گراف زیر را در نظر بگیرید:

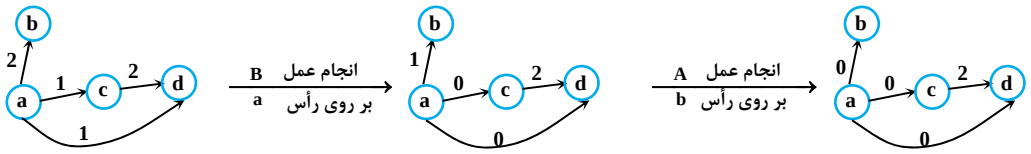



در این حالت گراف $G''$ دور با وزن منفی دارد اما G دوری ندارد که تعداد یال‌های آن کمتر از وزن آن دور باشد.

در همان گراف قبل اگر وزن یال $a \to b$ برابر 3- باشد، آنگاه G دارای یک دور با وزن منفی خواهد بود در صورتی که گراف $G''$ دوری ندارد که تعداد یال‌های آن کمتر از وزن آن دور باشد.

---

**۵۵ـ گزینه «۱»** گراف زیر را در نظر بگیرید:




بنابراین گزینه‌های ۲ و ۳ و ۴ نادرست می‌باشد.

---

**۵۶ـ گزینه «۳»** مورد اول نادرست است (BFS از صف استفاده می‌کند.) اما موارد دوم و سوم درست می‌باشند. به نکات زیر در مورد گزینه‌های مطرح شده دقت کنید:

الگوریتم DFS از پشته استفاده می‌کند.

الگوریتم BFS از صف استفاده می‌کند.

درخت فراگیر کمینه (MST) را می‌توان در ساخت.

یک درخت فراگیر کمینه همواره شامل یال با کمترین وزن است.

یک درخت فراگیر کمینه ممکن است شامل یال با بیشترین وزن باشد.

---

**۵۷ـ گزینه «۲»** با استفاده از الگوریتم DFS می‌توان این مسئله را حل نمود.

---

**۵۸ـ گزینه «۴»** با توجه به این که تعداد یال‌های گراف $O(n)$ می‌باشد بنابراین، گراف موردنظر یک گراف اسپارس است. در نتیجه در الگوریتم‌های پریم و دایکسترا زمان اصلی الگوریتم صرف مرتب‌سازی یال‌ها می‌گردد و بنابراین زمان $O(n \log_2^n)$ خواهد بود.

زمان اجرای الگوریتم بلمن ـ فورد برابر $O(|V||E|)$ می‌باشد و با توجه به این که $|V| = n$ و $|E| = O(n)$ بنابراین، مرتبه الگوریتم $O(n^2)$ خواهد بود.

کوتاهترین مسیر در یک گراف DAG بدون وزن را می‌توان با استفاده از BFS تعیین نمود که زمان آن $O(|V| + |E|)$ می‌باشد و با توجه به این که $|V| = n$ و $|E| = O(n)$ بنابراین، مرتبه کلی الگوریتم $O(n)$ خواهد بود.

---

**۵۹ـ گزینه «۳»** موارد اول و چهارم نادرست است اما موارد دوم و سوم گزاره‌های درستی را بیان می‌کنند.

---

**۶۰ـ گزینه «۲»** اگر ماتریس $A^{(k)}$ محاسبه گردد، آنگاه $A_{ij}^{(k)}$ نشان‌دهنده تعداد مسیرهای به طول k بین رئوس i و j را نشان می‌دهد.

---

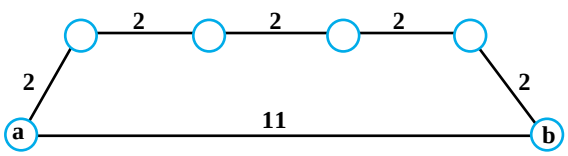
**۶۱ـ گزینه «۱»** در الگوریتم BFS از یال‌های مستقیم بین همسایه‌ها استفاده می‌شود که با توجه به صورت تست کم هزینه‌ترین راه برای رفتن از یک رأس به رأس دیگر، مسیری است که در آن کمترین یال ممکن طی شود.

---

**۶۲ـ گزینه «۲»** اگر به تمام یالهای یک گراف یک واحد اضافه گردد، آنگاه ممکن است برش کمینه آن تغییر کند.

مورد II : درخت فراگیر کمینه (MST) گراف حاصل بدون تغییر باقی می‌ماند (بازهم از لحاظ ساختار).

مورد III: نادرست است. به عنوان مثال گراف مقابل را در نظر بگیرید:




در این گراف کوتاه‌ترین مسیر بین a و b دارای وزن 10 می‌باشد اما اگر به هر یک از یال‌ها یک واحد اضافه شود، آنگاه کوتاه‌ترین مسیر از a به b کاملاً تغییر می‌کند.

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



# بخش پنجم: مرتب‌سازی و مرتبه‌های آماری

## فصل چهاردهم

## «مرتب‌سازی‌های مقایسه‌ای»

### تست‌های طبقه‌بندی شده کنکوری فصل چهاردهم

۱ـ پیچیدگی کدام یک از الگوریتم‌های مرتب‌کننده زیر (برحسب تابعی از اندازه ورودی) در حالت متوسط **(Average Case)** و در بدترین حالت **(Worst Case)** با هم متفاوت است؟ (مهندسی کامپیوتر ـ سراسری۷۵)

۱) Ouick Sort ۲) Binary Insertion Sort ۳) Heap Sort ۴) Merge Sort

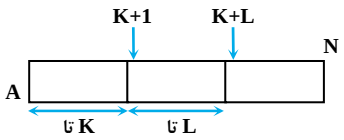
۲ـ کدام یک از روش‌های مرتب‌سازی زیر ترتیب اولیه رکوردهای هم کلید (منظور برابری کلید مرتب‌سازی در این رکوردها است) را حفظ می‌کند؟ (مهندسی کامپیوتر ـ سراسری۷۶)

۱) Selection Sort ۲) Quick Sort ۳) Insertion Sort ۴) Heap Sort

۳ـ الگوریتم زیر برای مرتب‌سازی صعودی آرایه $A$ با $N$ عنصر (با اندیس‌های از 1 تا $N$) پیشنهاد شده است. در این الگوریتم، فرض کنید $sort(A,i,j)$ عناصر $i$ تا $j$ از آرایه $A$ را با یکی از الگوریتم‌های شناخته شده به صورت صعودی مرتب می‌کند.

**sort (A, K+1 , N)** ; **sort (A, 1, K+L)** ; **sort (A, K+1, N)**

کدام‌یک از گزینه‌های زیر برای هر $N$ درست است؟ (مهندسی کامپیوتر ـ سراسری۷۸)




۱) تنها اگر $K \le L$ باشد این الگوریتم درست است.

۲) تنها اگر $K > L$ باشد این الگوریتم درست است.

۳) تنها اگر $N = 3K = 3L$ باشد الگوریتم درست است.

۴) تنها اگر $K = L$ باشد الگوریتم درست است.

۴ـ یک ماتریس $N \times M$ از اعداد صحیح دلخواه داده شده است. الگوریتم زیر را بر روی این ماتریس انجام می‌دهیم:

الف ـ هر کدام از سطرهای ماتریس را مستقلاً از چپ به راست به صورت صعودی مرتب می‌کنیم.ب ـ هر کدام از ستون‌های ماتریس را مستقلاً از بالا به پایین به صورت صعودی مرتب می‌کنیم.کدام یک از گزینه‌های زیر درست است؟ (مهندسی کامپیوتر ـ سراسری۷۹)

۱) سطرها لزوماً مرتب نیستند، ولی اگر یک بار دیگر سطرها را مرتب کنیم، هم سطرها و هم ستون‌ها مرتب می‌شوند.

۲) سطرها لزوماً مرتب نیستند، ولی اگر یک بار دیگر هم سطرها را مرتب کنیم، ستون‌ها لزوماً مرتب نمی‌شوند.

۳) سطرها و ستون‌ها هر کدام به صورت صعودی مرتب می‌شوند.

۴) تکرار این الگوریتم لزوماً هم سطرها و هم ستون‌ها را مرتب نمی‌کند.

۵ـ آرایه $A$ شامل $n$ عنصر مرتب (از اندیس 1 تا $n$) و $k$ عنصر نامرتب (از اندیس $n+1$ تا $n+k$) است. کدام یک از الگوریتم‌های زیر برای مرتب‌سازی $A$ کمترین تعداد مقایسه را دارد؟ (فرض کنید $k$ مستقل از $n$ و بسیار کمتر از آن است). (مهندسی کامپیوتر ـ سراسری۷۹)

۱) Insertion Sort ۲) Quick Sort ۳) Heap Sort ۴) Merge Sort

۶ـ کدام یک از گزینه‌های زیر در مورد الگوریتم **Quicksort** درست است؟ فرض کنید که عنصر اول همواره به عنوان محور **(Pivot)** انتخاب می‌شود. ( $n$ تعداد عناصر آرایهٔ ورودی است). اگر آرایهٔ ورودی از قبل مرتب باشد، . . . . . . (مهندسی کامپیوتر ـ سراسری ۸۰)

۱) زمان اجرای الگوریتم $O(n)$ است.

۲) زمان اجرای الگوریتم $O(n^2)$ است.

۳) زمان اجرای الگوریتم $O(n \log n)$ است.

۴) متوسط زمان اجرا $O(n \log n)$ است.

۷ـ لیست زیر را درنظر بگیرید. اگر عنصر اول لیست یعنی عدد ۹ را به عنوان لولا **(pivot)** اختیار کنیم کدام یک از گزینه‌های زیر می‌توانند خروجی مرحله اول الگوریتم مرتب‌سازی سریع **(Quick Sort)** باشد؟ (علوم کامپیوتر ـ سراسری ۸۰)

۱) (7,8,9,10,3,6,15) ۲) (7,8,9,3,6,10,15) ۳) (6,3,8,7,9,15,10) ۴) (6,7,8,9,3,10,15)

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



۸ـ اگر برای تمام مقادیر $i$، $1 \le i \le n$ و $A[i] \ge 1$ و $A[0] = 0$ باشد، روال زیر آرایه $A[0 \ldots n]$ را به چه روشی مرتب می‌کند؟ (مهندسی کامپیوتر ـ سراسری ۷۶ و آزاد ۸۱)

```
for i: = 2 to n do
begin
  j: = i;
  while A[j] < A [j – 1] do
    begin
    T: = A[j – 1];
    A[j – 1]: = A [j];
    A[j]:= T;
    j: = j – 1;
   end;
end;
```

۱) Shell Sort
۲) Bubble Sort
۳) Selection Sort
۴) Insertion Sort

۹ـ آرایه $n$ خانه‌ای $A$ را در نظر می‌گیریم که برای ذخیره‌سازی عناصر یک درخت دودوئی کامل مورد استفاده قرار گرفته است. فرض کنیم الگوریتمی کارا برای بررسی این که این درخت دودویی یک heap است یا خیر در اختیار داریم. پیچیدگی این الگوریتم در بدترین حالت چقدر است؟ (مهندسی کامپیوتر ـ سراسری ۸۱)

۱) $O(n)$ ۲) $O(n^2)$ ۳) $O(\log n)$ ۴) $O(n \log n)$

۱۰ـ رویه Partition در الگوریتم Quick Sort به صورت زیر است:

```
function  partition (p , r : integer) : integer;
      Var x , i , j : integer;
begin
      x:=A[p] ;       i : = p-1 ; j:=r+1
      while  true  do  begin
      repeat     j : = j-1        until A[j] < = x ;
      repeat     i : = i+1        until A[i] > = x ;
      if    i < j      then  swap (A [i] , A[j]) else  return (j)
   end
end
```

اگر تمام درایه‌های آرایه $A[p..r]$ دارای مقدار یکسانی باشند، مقداری که رویهٔ فوق بر می‌گرداند چقدر است؟ (مهندسی کامپیوتر ـ سراسری ۸۱)

۱) $p$ ۲) $r$ ۳) $\left\lceil \frac{p+r}{2} \right\rceil$ ۴) $\left\lfloor \frac{p+r}{2} \right\rfloor$

۱۱ـ سه آرایه $A$، $B$ و $C$ هر کدام با $n$ عنصر را در نظر بگیرید به قسمی که $A$ و $C$ مرتب شده صعودی و $B$ مرتب شده نزولی است. همچنین $A$ و $B$ عنصر تکراری ندارند ولی $C$ دارد. به منظور مرتب‌سازی صعودی، الگوریتم‌های Selection Sort، Straight Insertion Sort، Heap Sort، Quick Sort را بر روی هر کدام از این آرایه‌ها اجرا می‌کنیم. در این صورت می‌توان گفت: (مهندسی کامپیوتر ـ سراسری ۸۲)

۱) برای B پیچیدگی زمانی Heap Sort و Quick Sort یکسان است.
۲) برای C برنامه Quick Sort سریعتر از Heap Sort به پایان می‌رسد.
۳) برای C پیچیدگی زمانی Selection Sort و Straight Insertion Sort یکسان است.
۴) برای A اجرای Straight Insertion Sort سریعتر از Selection Sort و Quick Sort به پایان می‌رسد.

۱۲ـ دو مجموعه $A$ و $B$ به ترتیب دارای $m$ و $n$ عنصر هستند با $m \le n$ و در این صورت: (مهندسی کامپیوتر ـ سراسری ۸۲)

۱) اگر آرایه‌های $A$ و $B$ مرتب نباشند نمی‌توان اجتماع و اشتراک آنها را در زمان $\theta(n)$ به دست آورد.
۲) اگر آرایه‌های $A$ و $B$ مرتب باشند نمی‌توان در زمانی کمتر از $\theta(n^2)$ اجتماع و اشتراک آنها را به دست آورد.
۳) این دو آرایه چه مرتب باشند و چه نامرتب، نمی‌توان در زمانی کمتر از $\theta(n^2)$ اجتماع و اشتراک آنها را به دست آورد.
۴) اگر آرایه‌های $A$ و $B$ مرتب باشند می‌توان اجتماع و اشتراک آنها را در زمان $\theta(n)$ به دست آورد.

۱۳ـ پنج فایل مرتب شده به اندازه‌های $30,25,20,10,5$ مفروضند. می‌خواهیم از ادغام دوبه‌دوی آنها یک فایل مرتب شده واحد شامل همهٔ رکوردها به دست آوریم. در هر ادغام رکوردهای فایل‌های ورودی ممکن است چند بار از یک فایل خوانده و در یک فایل دیگر نوشته شوند. به هر کدام از این نوشتن‌ها و خواندن‌ها یک جابجایی می‌گوئیم. حداقل تعداد کل این جابجایی‌ها برای ادغام همهٔ فایل‌ها چقدر است؟ (مهندسی کامپیوتر ـ سراسری ۸۲)

۱) 195 ۲) 200 ۳) 185 ۴) 215

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



۱۴ـ آرایه $n$ عنصری $A$ را در نظر بگیرید، فرض کنید $n = 2^k$. الگوریتم Mergesort بر روی $A$، (مهندسی کامپیوتر ـ سراسری ۸۲)

۱) در بدترین حالت $n.\log_2 n - n + 1$ مقایسه میان عناصر آرایه انجام می‌دهد.

۲) در بدترین حالت $n.\log_2 n + n - 1$ مقایسه میان عناصر آرایه انجام می‌دهد.

۳) در بهترین حالت $n.\log_2 n - 2n - 1$ مقایسه میان عناصر آرایه انجام می‌دهد.

۴) در بهترین حالت $n.\log_2 n - n - 1$ مقایسه میان عناصر آرایه انجام می‌دهد.

۱۵ـ حداقل تعداد مقایسه‌های هر الگوریتم مرتب‌کننده که با مقایسه 8 عنصر را مرتب می‌کند، کدام است؟ (علوم کامپیوتر ـ سراسری ۸۲)

۱) 13 ۲) 14 ۳) 15 ۴) 16

۱۶ـ برای ادغام لیست‌های مرتب شده $L_1, L_2, \ldots, L_k$، کدام یک از ساختار داده‌های زیر (علاوه بر لیست‌های ورودی) بهترین است؟ (مهندسی کامپیوتر ـ سراسری ۸۳)

۱) درخت Min-heap ۲) لیست دو طرفه خطی ۳) آرایه خطی ۴) درخت دودویی جست و جو

۱۷ـ در یک آرایهٔ $A$ به اندازهٔ $n$، اگر $i < j$ و $A[i] > A[j]$ می‌گوییم که زوج $(i, j)$ یک «زوج ـ معکوس» (inversion) در $A$ است. بیشترین تعداد زوج ـ معکوس‌ها در یک آرایه $n$ عضوی چند تاست؟ (مهندسی کامپیوتر ـ سراسری ۸۳)

۱) $\frac{n(n-1)}{2}$ ۲) $n^2$ ۳) $n^2 - n$ ۴) $\frac{n^2}{2}$

۱۸ـ می‌خواهیم یک آرایه $A$ با $n$ عنصر مجزا را به $k = 2^r$ زیر آرایه $A_1, \ldots, A_k$ هر یک با تعداد عناصر $\left\lceil \frac{n}{k} \right\rceil$ یا $\left\lfloor \frac{n}{k} \right\rfloor$ تقسیم کنیم به طوری که برای همهٔ $i < j$ و هر $a_i \in A_i$ ، $a_j \in A_j$ داشته باشیم $a_i < a_j$. بهترین الگوریتم برای این کار چه ویژگی دارد؟ (مهندسی کامپیوتر ـ سراسری ۸۳)

۱) همیشه از $O(n \lg k)$ است. ۲) همیشه از $O(n \lg n)$ است.

۳) در حالت متوسط از $O(n \log n)$ است. ۴) در حالت متوسط از $O(n)$ است.

۱۹ـ در گونه جدیدی از مرتب‌سازی سریع (quick sort)، برای انتخاب محور از میان $n$ عنصر، $2\sqrt{n}+1$ عنصر اول آرایه را انتخاب می‌کنیم و با یک الگوریتم ساده مانند مرتب‌سازی درجی، آن‌ها را مرتب می‌کنیم. عنصر میانه این تعداد عنصر مرتب را به عنوان محور الگوریتم انتخاب می‌کنیم. بقیه الگوریتم مانند قبل عمل می‌کند. بدترین زمان اجرای این الگوریتم با کدام رابطه بازگشتی نشان داده می‌شود؟ (مهندسی کامپیوتر ـ سراسری ۸۳)

۱) $T(n) = 2T(\sqrt{n}) + n$ ۲) $T(n) = T(n-1) + n$

۳) $T(n) = 2T(n - \sqrt{n}) + n$ ۴) $T(n) = T(n - \sqrt{n}) + T(\sqrt{n}) + n$

۲۰ـ عمق یک گره در درایه‌ی $i$ در یک هیپ با $n$ گره چه قدر است؟ (مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۳)

۱) $[\lg i]$ ۲) $i - 1$ ۳) $\lfloor i/2 \rfloor$ ۴) $\lceil i/2 \rceil$

۲۱ـ آرایه $n$ عنصری $A[1:n]$ را در نظر بگیرید، کدام یک از گزاره‌های ذیل، درست یا درست‌تر است؟ (مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۳)

۱) الگوریتم INSERTION SORT، در بدترین حالت و به طور متوسط، زمان مصرفی مرتبه $n^2$ دارد.

۲) الگوریتم QUICK SORT، در بدترین حالت و به طور متوسط زمانی در گروه $n.\log n$ دارد.

۳) الگوریتم QUICK SORT، همواره از الگوریتم INSERTION SORT، از نظر زمان مصرفی، بهتر است.

۴) الگوریتم MERGE SORT، همواره از INSERTION SORT بهتر است، اما SELECTION SORT از هر دو، از نظر زمان مصرفی بهتر است.

۲۲ـ یک الگوریتم مرتب‌سازی برای مرتب کردن $n$ عدد حداقل نیاز به چه تعداد مقایسه دارد؟ (علوم کامپیوتر ـ سراسری ۸۳)

۱) $\log n$ ۲) $\log n!$ ۳) $n$ ۴) $n^2$

۲۳ـ می‌خواهیم $k$ تا کوچکترین عناصر یک آرایه را از کوچک به بزرگ به صورت مرتب به دست آوریم. کدام یک از الگوریتم‌های زیر درست و سریعتر از بقیه است؟ (مهندسی کامپیوتر ـ سراسری ۸۴)

۱) عناصر را مرتب می‌کنیم و سپس $k$تای اول آن را به ترتیب برمی‌داریم.

۲) یک Heap بر روی این عناصر می‌سازیم و سپس $k$ بار کوچکترین عنصر را بدست می‌آوریم.

۳) $k$امین عنصر را انتخاب می‌کنیم این عنصر را محور قرار داده و عمل بخش‌پذیری را انجام می‌دهیم و سپس کار را دنبال می‌کنیم.

۴) عناصر میانه را به دست آورده و براساس این محور عمل بخش‌بندی را انجام می‌دهیم سپس در یکی از بخش‌ها کار را دنبال می‌کنیم.

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۲۴ـ آرایه n تایی A حاوی جایگشتی از اعداد 1 تا n است. برنامه‌ی زیر این اعداد را مرتب می‌کند. بیشترین تعداد تعویض‌های برنامه فوق چند تا است؟**

(مهندسی کامپیوتر ـ سراسری ۸۴)

```
For i = 1 to n – 1 do
    while A[i] <> i do
        Swap (A[i], A[A[i])
```

۱) $n$

۲) $n-1$

۳) $n \log n$

۴) $\frac{n(n-1)}{2}$

**۲۵ـ به منظور sort صعودی یک آرایه از heapsort استفاده نموده و max-heap می‌سازیم. در مورد زمان اجرای این الگوریتم کدام گزینه صحیح است؟**

(مهندسی کامپیوتر ـ سراسری ۸۴)

۱) اگر آرایه از ابتدا به صورت صعودی مرتب شده باشد زمان مورد نیاز $\theta(n)$ است.

۲) اگر آرایه از ابتدا به صورت نزولی مرتب شده باشد زمان مورد نیاز $\theta(n)$ است.

۳) اگر آرایه از ابتدا به صورت نزولی مرتب شده باشد زمان مورد نیاز $\theta(n^2 \log n)$ است.

۴) اگر آرایه از ابتدا به صورت صعودی مرتب شده باشد زمان مورد نیاز $\theta(n \log n)$ است.

**۲۶ـ آرایه‌ی نامرتب $n-1$ عضوی A شامل $n-1$ عدد صحیح مجزای بین 1 تا n است، به جز یک عدد x و می‌خواهیم با یک الگوریتم کارا x را پیدا کنیم. کدام یک از گزینه‌های زیر الگوریتم درست و از بقیه سریع‌تر برای این کار است؟**

(مهندسی کامپیوتر ـ سراسری۸۴)

۱) A را مرتب می‌کنیم و سپس با پیمایش آرایه‌ی مرتب، x را پیدا می‌کنیم.

۲) یک heap بر روی A می‌سازیم. از ریشه حرکت می‌کنیم و به یکی از زیر درخت‌های آن می‌رویم و کار جستجو را دنبال می‌کنیم.

۳) میانه‌ی A را پیدا می‌کنیم و براساس آن به عنوان محور عمل «بخش‌بندی» (partition) را انجام می‌دهیم و بر این اساس کار جستجو را ادامه می‌دهیم.

۴) یک درخت دودویی جست و جوی متوازن بر روی اعداد می‌سازیم و کار جستجو را از ریشه و سپس در یکی از زیر درخت‌های آن دنبال می‌کنیم.

**۲۷ـ زیر برنامه New quick sort را برای مرتب سازی آرایه T به ترتیب نزولی در نظر می‌گیریم. pivot (T,p,k) عناصر T را حول محور p طوری می‌چرخاند که پس از اتمام کار، کلیه عناصر بزرگتر از p قبل از خانه k و عناصر کوچکتر و مساوی بعد از آن در T جای می‌گیرد (k پارامتر خروجی است) Insertion sort به روش درجی آرایه را به ترتیب نزولی مرتب می‌کند. پیچیدگی زمانی این زیر برنامه برای کدام یک از آرایه‌های زیر مطمئناً کمتر از پیچیدگی زمان الگوریتم معمولی Quick Sort است؟**

(مهندسی کامپیوتر ـ سراسری ۸۴)

```
New Quick sort (T[1...n])
    p ← T[1]
    pivot (T, p, k)
Insertion sort(T[1...k – 1])
Insertion sort(T(k + 1...n))
```

۱) برای همهٔ حالات ورودی

۲) آرایه ابتدایی مرتب‌شده نزولی باشد.

۳) آرایه ابتدایی مرتب‌شده صعودی باشد.

۴) p دقیقاً میانه آرایه باشد و آرایه ابتدایی نامرتب باشد.

**۲۸ـ در گونه‌ی جدیدی از مرتب‌سازی سریع (quick sort)، برای انتخاب محور از میان n عنصر، $2\sqrt{n}+1$ عنصر اول آرایه را انتخاب می‌کنیم و با یک الگوریتم ساده (مانند Insertion Sort) آن‌ها را مرتب می‌کنیم. محور، عنصر میانه‌ی این تعداد عنصر مرتب است. بقیه‌ی الگوریتم مانند quick sort عمل می‌کند. کدام‌یک از رابطه‌های بازگشتی زیر، زمان اجرای این الگوریتم در بدترین حالت را نشان می‌دهد؟**

(مهندسی کامپیوتر ـ سراسری ۸۴)

۱) $T(n) \le T(n-\sqrt{n}) + O(\sqrt{n})$

۲) $T(n) \le T(2\sqrt{n}) + T(n-2\sqrt{n}) + O(n)$

۳) $T(n) \le T(\sqrt{n}) + T(n-\sqrt{n}) + O(n)$

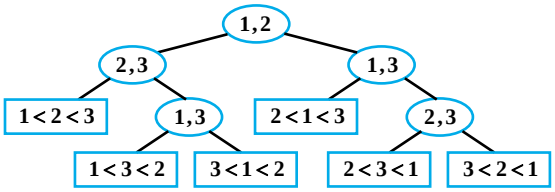
۴) $T(n) \le T(2\sqrt{n}) + T(n-2\sqrt{n}) + O(n)$

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۲۹ـ یک درخت دودویی تصمیم‌گیری را در نظر بگیرید که از آن برای مرتب کردن عناصر یک مجموعه، براساس عمل مقایسه، استفاده می‌شود. برای مثال درخت زیر، برای مرتب کردن یک فایل با 3 عضو، استفاده می‌شود. کدام یک از گزاره‌های ذیل، در مورد این نوع درخت‌ها، درست است؟** (مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۴)




۱) عمق این نوع درخت‌ها، حداقل $\log_2(n!)$ است و در گروه $O(n.\lg n)$ قرار می‌گیرد.
۲) عمق این نوع درخت‌ها، حداکثر $\log_2(n!)$ است و در گروه $O(n.\lg n)$ قرار می‌گیرد.
۳) عمق این نوع درخت‌ها، حداکثر $\log_2(n!)$ است و حداقل $O(n.\lg n)$ مقایسه در آنها انجام می‌شود.
۴) عمق این نوع درخت‌ها، حداقل $\log_2(n!)$ است و حداکثر $O(n.\lg n)$ مقایسه در آنها انجام می‌شود.

**۳۰ـ دومین عدد بزرگ یک آرایه n عضوی را با چند مقایسه می‌توان پیدا نمود؟** (مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۴)

۱) $n+\lfloor \log_2 n \rfloor$ ۲) $(n-1)+(n-2)$ ۳) $n+\lceil \log_2 n \rceil-2$ ۴) $n+\lfloor \log_2 n \rfloor-2$

**۳۱ـ یک الگوریتم مرتب‌سازی صعودی در زمان اجرا در چهارمین تکرار خود دارای ارزش‌های زیر می‌باشد. این الگوریتم چه نوع مرتب‌سازی می‌باشد؟ 15,18,30,34,41,0,91,32,49** (علوم کامپیوتر ـ سراسری ۸۴)

۱) Bubble ۲) Selection ۳) Insertion ۴) Quicksort

**۳۲ـ ترتیب پیچیدگی الگوریتم‌های Quick Sort و Heap sort در بدترین حالت به‌ترتیب از راست به چپ برابر است با:** (مهندسی کامپیوتر ـ آزاد ۸۴)

۱) $O(n\log n), O(n\log n)$ ۲) $O(n^2), O(n^2)$ ۳) $O(n\log n), O(n^2)$ ۴) هیچ‌کدام

**۳۳ـ پیچیدگی کدام یک از الگوریتم‌های مرتب کننده زیر (بر حسب تابعی از اندازه ورودی، n) در حالت متوسط (Average Case) و در بدترین حالت (Worst Case) با هم متفاوت است؟** (مهندسی کامپیوتر و مهندسی فناوری اطلاعات (IT) ـ آزاد ۸۴)

۱) Quick Sort ۲) Binary Insertion Sort ۳) Heap Sort ۴) Merge Sort

**۳۴ـ آرایه A تقریباً مرتب شده است. یعنی برای $i=1,2,\ldots,n-k$ داریم $A[i]\le A[i+k]$ برای مرتب نمودن تمام n عضو حداقل چه مقدار زمان لازم است؟** (مهندسی کامپیوتر ـ سراسری ۸۵)

۱) $\theta(n)$ ۲) $\theta(n.k)$ ۳) $O(n\log k)$ ۴) $O(n\log n)$

**۳۵ـ آرایه نامرتب A شامل $n-1$ عدد صحیح و مجزا بین یک تا n است. به جز یک عدد x، می‌خواهیم با یک الگوریتم کارا x را پیدا کنیم. کدام یک از الگوریتم‌های زیر درست و نسبت به بقیه سریع‌تر است؟** (مهندسی کامپیوتر ـ سراسری ۸۵)

۱) A را مرتب می‌کنیم و با یک پیمایش، x، A را به دست می‌آوریم.
۲) یک min−heap بر روی A می‌سازیم، از ریشه شروع می‌کنیم و برای پیدا کردن x به یکی از زیردرخت‌های آن می‌رویم و کار را به صورت بازگشتی دنبال می‌کنیم.
۳) یک درخت دودوئی جست‌وجوی متوازن بر روی عناصر A می‌سازیم و کار جست‌وجو برای پیدا کردن x را از ریشه و سپس یکی از زیردرخت‌های آن دنبال می‌کنیم.
۴) میانه A را پیدا می‌کنیم و آن را برای عمل بخش‌بندی (Partition) محور قرار می‌دهیم. بر این اساس، کار جست‌وجو را دنبال می‌کنیم تا x به دست آید.

**۳۶ـ براساس قطعه کد زیر، کدام یک از گزینه‌ها قطعاً در محل *** درست است؟** (مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۵)

```
for (item=n;item>1;item--)
{
 int large =a[1];
 int index=1;
 for(i=2;i<=item;i++)
  if(a[i]>large)
  {
   large=a[i];
   index=i;
  }
a[index]=a[item];
a[item]=large;
//***
}
```

۱) برای تمام j‌هایی که $1<j<=item$، $a[j]<a[j-1]$
۲) برای تمام j‌هایی که $item<j<=n$، $a[j]<a[j+1]$
۳) برای تمام j‌هایی که $item<=j<n$، $a[j]<=a[j+1]$
۴) برای تمام j‌هایی که $item<=j<n$، $a[j]<a[j+1]$

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



۳۷ـ اگر در الگوریتم **Merge Sort**، برای مرتب سازی لیست‌های دارای کمتر از 20 عنصر از الگوریتم **Insertion Sort** استفاده شود، پیچیدگی زمانی الگوریتم چه خواهد شد؟ (مهندسی IT ـ سراسری ۸۵)

۱) $\theta(n^2)$ ۲) $\theta(n)$ ۳) $\theta(n^2 Logn)$ ۴) $\theta(n \log n)$

۳۸ـ جواب تابع بازگشتی زیر پیچیدگی زمانی کدام یک از الگوریتم‌های زیر را نشان می‌دهد؟ (علوم کامپیوتر ـ سراسری ۸۵)

$$t(1) = 0$$

$$t(n) = t(n-1) + \frac{2}{n}$$

۱) Binary search ۲) Insertion sort ۳) Quick sort ۴) Sequential search

۳۹ـ اگر بخواهیم دو **min-heap** به تعداد **m** و **n** را با هم **merge** کنیم پیچیدگی زمانی آن برابر است با: (علوم کامپیوتر ـ سراسری ۸۵)

۱) $m \log mn$ ۲) $m \log n + n \log m$ ۳) $n \log n . \lg m$ ۴) $m . \log n . \log m$

۴۰ـ الگوریتم مرتب‌سازی **(Heap sort)** با چه سرعتی یک آرایه شامل **n** عنصر را مرتب می‌نماید؟ (مهندسی کامپیوتر ـ آزاد ۸۵)

۱) $O(n^2)$ ۲) $O(n^2 \log n)$ ۳) $O(n^3)$ ۴) $O(n \log n)$

۴۱ـ کدام‌یک از الگوریتم‌های مرتب‌سازی زیر، جهت مرتب نمودن یک آرایه **n** عنصری نامرتب دارای پیچیدگی زمانی $O(n \log n)$ نمی‌باشد؟ (مهندسی کامپیوتر ـ آزاد ۸۵)

۱) Quick Sort ۲) Insertion Sort ۳) Merge Sort ۴) Heap Sort

۴۲ـ آرایه‌ای شامل **n** عنصر می‌باشد. می‌خواهیم با استفاده از روش مرتب‌سازی درختی **(Tree Sort)** عناصر آرایه را به‌ترتیب در درخت **T**، درج و پیمایش و در خروجی ثبت نماییم. در بدترین حالت هزینه درج و پیمایش برای درخت فوق چیست؟ (مهندسی کامپیوتر ـ آزاد ۸۵)

۱) هزینه درج $O(\log n)$ و هزینه پیمایش $O(n \log n)$ می‌باشد.
۲) هزینه درج $O(n)$ و هزینه پیمایش $O(n^2)$ می‌باشد.
۳) هزینه درج $O(n)$ و هزینه پیمایش $O(n \log n)$ می‌باشد.
۴) هزینه درج $O(\log n)$ و هزینه پیمایش $O(n^2)$ می‌باشد.

۴۳ـ الگوریتم **Quick sort** یک رشته **n** تایی مرتب شده را با چه سرعتی مرتب می‌نماید؟ (مهندسی کامپیوتر ـ آزاد ۸۵)

۱) $O(n \log n)$ ۲) $O(\log n)$ ۳) $O(n^2)$ ۴) $O(\log_2^n)$

۴۴ـ الگوریتم مرتب‌سازی ادغامی **(merge sort)** را بر یک آرایه **n** عنصری مرتب اعمال می‌کنیم. زمان موردنیاز برای الگوریتم فوق چیست؟ (مهندسی کامپیوتر ـ آزاد ۸۵)

۱) $O(n \log n)$ ۲) $O(n)$ ۳) $O(n^2)$ ۴) $O(\frac{n}{2})$

۴۵ـ از الگوریتم‌های مرتب‌سازی ۱) حبابی ۲) درجی ۳) انتخابی ۴) سریع، کدام یک در بدترین حالت از مرتبه زمانی $O(n^2)$ می‌باشند. (**n** تعداد عناصر آرایه می‌باشد) (مهندسی فناوری اطلاعات (IT) و مهندسی کامپیوتر ـ آزاد ۸۵)

۱) 2,1 ۲) 4,3,2,1 ۳) 3,2,1 ۴) 3,1

۴۶ـ کدام یک از الگوریتم‌های مرتب‌سازی زیر در بهترین حالت از مرتبه زمانی $O(n)$ می‌باشند؟ (**n** تعداد عناصر آرایه می‌باشند) (مهندسی فناوری اطلاعات (IT) و مهندسی کامپیوتر ـ آزاد ۸۵)

۱) مرتب‌سازی درجی (Insertion sort)
۲) مرتب‌سازی حبابی (Bubble sort)
۳) مرتب‌سازی انتخابی (Selection sort)
۴) مرتب‌سازی سریع (Quick sort)

۴۷ـ به منظور مرتب‌سازی یک آرایه نامرتب شامل **n** عنصر از الگوریتم **Quick sort** استفاده می‌کنیم. در این صورت سرعت اجرای الگوریتم چه خواهد بود؟ (مهندسی کامپیوتر و مهندسی فناوری اطلاعات (IT) ـ آزاد ۸۵)

۱) $O(n^2)$ ۲) $O(\log_2^n)$ ۳) $O(\log 2n)$ ۴) $O(n \log n)$

۴۸ـ الگوریتم مرتب‌سازی ادغامی **(Merge sort)** یک آرایه **n** عنصری را با چه زمانی مرتب می‌نماید؟ (مهندسی فناوری اطلاعات (IT) ـ آزاد ۸۵)

۱) $O(n^2)$ ۲) $O(2^n)$ ۳) $O(n \log n)$ ۴) $O(n^3)$

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



۴۹ـ اگر الگوریتم مرتب‌سازی درجی (Insertion sort) را بر یک آرایه N عنصری مرتب اعمال کنیم. زمان مورد نیاز برای الگوریتم فوق چیست؟
(مهندسی فناوری اطلاعات (IT) و مهندسی کامپیوتر ـ آزاد ۸۵)

۱) $O(n\log n)$ ۲) $O(n)$ ۳) $O(n^2)$ ۴) $O(\frac{n}{2})$

۵۰ـ الگوریتم pancake sort براساس رفتار آشپزها با پن کیک بنا شده است. فرض کنید n عدد پن کیک بر روی هم قرار گرفته‌اند و بر روی هر کدام عددی نوشته شده است که قابل رؤیت می‌باشد . تنها کار ممکن، فرو کردن کفگیر قبل از یک پن کیک خاص و وارونه کردن همه پن کیک‌های بالای کفگیر است. این کار را عمل «وارون » می‌گوییم که می‌تواند از هر جا انجام شود. هدف از این الگوریتم مرتب کردن صعودی اعداد روی پن کیک‌ها با استفاده از کم‌ترین تعداد عمل «وارون » است. اگر تعداد $n \geq 6$ باشد با چه تعداد عمل وارون می‌توان یقیناً اعداد را مرتب کرد؟ (کمترین گزینه را علامت بزنید)
(مهندسی کامپیوتر ـ سراسری ۸۶)

۱) n ۲) $n-1$ ۳) $2n-1$ ۴) $2n-2$

۵۱ـ یک آرایه $A=[1..n]$ را $-k$ مرتب می‌گوییم اگر برای هر i که $k < i \leq n-k$ داشته باشیم: $A[i-k] \leq A[i] \leq A[i+k]$
مثلاً عناصر 8 5 7 3 6 2 4 1 (از چپ به راست ) یک آرایه‌ی 2- مرتب است. در یک آرایه 2- مرتب با 2N عنصر، حداکثر اختلاف بین اندیس یک عنصر در این آرایه و اندیس همان عنصر اگر آرایه 1- مرتب می‌بود چند است؟ (مهندسی کامپیوتر ـ سراسری ۸۶)

۱) N ۲) 2 ۳) $\frac{N}{2}$ ۴) $2N-1$

۵۲ـ رویه partition در quick sort به صورت زیر است. اگر A با n عنصر نامساوی در ابتدا برعکس مرتب باشد، تعداد تعویض‌ها دقیقاً چند تاست؟
(مهندسی کامپیوتر ـ سراسری ۸۶)

```
Partition (A,p,r)
 x:=A[p]; i:=p-1; j:=r+1;
 while (true)
      repeat j:=j-1 until A[j] ≤ x
      repeat i:=i+1 until A[i] ≥ x
     if (i<j) then swap (A[i] , A[j]) else return j
   end
end
```

و خود quick sort به صورت زیر است:

```
Quick sort (A,p,r)
   q:= if p < r than begin
                partition (A,p,r)
                Quick sort (A,p,r)
                Quick sort (A,q+1,r)
      end
```

۱) $[\frac{n}{2}]$ ۲) n ۳) $n-1$ ۴) $2n-2$

۵۳ـ مرتبه زمانی مرتب‌سازی یک آرایه با n عنصر با روش Merge sort در بدترین حالت (Worst case) برابر است با:
(مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۶)

۱) n ۲) $n^2$ ۳) $n\log n$ ۴) $n^2\log n$

۵۴ـ می‌خواهیم آرایه $A[0\cdots n]$ را با فرض این که $A[0]=0$ و $\forall i, 1 \leq i \leq n, A[i] \geq 1$ باشد. با روش زیر مرتب کنیم. نام این روش مرتب‌سازی چیست؟ (مهندسی فناوری اطلاعات (IT) ـ آزاد ۸۶)

```
For i := 2 to n do
begin
j :=i
 while A[j] < A[j−1]do
begin
 Tmp := A[j];
 A[j] := A[j−1];
 j := j−1;
end
end
```

۱) Selection Sort
۲) Quick Sort
۳) Insertion Sort
۴) Shell Sort

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



۵۵ـ فرض کنید می‌خواهیم مرتب‌سازی **Heap Sort** را روی آرایه‌ای که شامل **14** عنصر زیر است، انجام دهید.

$A[1]...A[14] = 16, 50, 23, 20, 21, 40, 41, 30, 3, 32, 33, 34, 35, 36$

برای تبدیل آرایه یک **Max Heap** به صورت درجا ( ) چند جابجایی بین گره با فرزند انجام می‌شود؟ (مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۷)

۱) 7 ۲) 6 ۳) 5 ۴) 4

۵۶ـ الگوریتم زیر نسخه بازگشتی کدام یک از الگوریتم‌های مرتب‌سازی است؟ (مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۷)

```
function  Sort (A[0-n-1],n)
if n>1  then
   for k=1    to n-1do
       if       A[k-1]>A[k]then
                  Swap (A,K,K-1)
     end if
     Sort (A,n-1)
  end for
end if
end sort
```

۱) Insertion sort
۲) Bubble Sort
۳) Selection Sort
۴) Shell sort

۵۷ـ اگر ارتفاع یک گره را فاصله طولانی‌ترین مسیر آن تا برگ در نظر بگیریم، حداکثر چند گره با ارتفاع **h** در یک **heap** تشکیل شده از **n** عنصر وجود دارد؟ (مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۷)

۱) $\left\lceil \frac{n+1}{2^h} \right\rceil$ ۲) $\frac{n+1}{2^h}$ ۳) $\left\lceil \frac{n}{2^{h+1}} \right\rceil$ ۴) $\left\lceil \frac{n}{2^h} \right\rceil$

۵۸ـ برای مرتب‌سازی آرایه‌ای با **2000** (دو هزار) عضو از الگوریتم **Randomized quick sort** استفاده شده است. اگر **Call stack** برنامه در شروع الگوریتم خالی باشد و برای هر فراخوانی تابع تنها **4** بایت آدرس برگشت در **call stack** قرار گیرد، در طی این فراخوانی، حداکثر طول اشغال شده **call-stack** به طور متوسط چقدر خواهد بود؟ (علوم کامپیوتر ـ سراسری ۸۷)

۱) 4 بایت ۲) 8 بایت ۳) 44 بایت ۴) 88 بایت

۵۹ـ در الگوریتم مرتب‌سازی حبابی (یا تبادلی) فرض می‌کنیم **T(n)** تعداد دستورالعمل مقایسه باشد. در این صورت **T(n)** کدام است؟ (مهندسی کامپیوتر ـ آزاد ۸۷)

۱) $T(n) = \frac{n(n+1)}{2}$ ۲) $T(n) = 2n - 1$ ۳) $T(n) = \frac{n(n-1)}{2}$ ۴) $T(n) = \frac{n^2}{2} - 1$

۶۰ـ در کدام یک از الگوریتم‌های مرتب نمودن زیر، پیچیدگی حالت میانگین با پیچیدگی بدترین حالت برابر نیست؟ (مهندسی کامپیوتر ـ آزاد ۸۷)

۱) Heap Sort ۲) Selection Sort ۳) Quick Sort ۴) Insertion Sort

۶۱ـ الگوریتم ادغام و مرتب کردن **(Merge Sort)** که بر روی آرایه **n** عنصری اعمال می‌شود، دارای کدام یک از مشخصه‌های زیر می‌باشد؟ (مهندسی فناوری اطلاعات (IT) و مهندسی کامپیوتر ـ آزاد ۸۷)

۱) پیچیدگی حالت میانگین $O(n \log n)$ می‌باشد. پیچیدگی بدترین حالت $O(n^2)$ می‌باشد. نیازمند یک آرایه کمکی با n عنصر می‌باشد.
۲) پیچیدگی حالت میانگین $O(n \log n)$ می‌باشد. پیچیدگی بدترین حالت $O(n \log n)$ می‌باشد. نیازمند یک آرایه کمکی با n عنصر می‌باشد.
۳) پیچیدگی حالت میانگین $O(n \log n)$ می‌باشد. پیچیدگی بدترین حالت $O(n)$ می‌باشد. نیازمند یک آرایه کمکی با n عنصر می‌باشد.
۴) پیچیدگی حالت میانگین و بدترین حالت با هم برابر بوده و تنها به تعداد محدودی از خانه اضافی حافظه احتیاج دارد که مستقل از n می‌باشد.

۶۲ـ الگوریتم زیر چه نام دارد و چه عملی انجام می‌دهد؟ (آرایه **M[1..n]** از نوع صحیح می‌باشد.) (مهندسی فناوری اطلاعات (IT) و مهندسی کامپیوتر ـ آزاد ۸۷)

```
Void Sort(int n,int M[ ])
{
int K,L;
for(K = n;K >= 2;K --)
for (L = K -1;L >= 1;L --)
if (M[K] > M[L])
Exchange (M[K],M[L]); }
```

۱) مرتب‌سازی حبابی ـ عمل مرتب‌سازی غیر نزولی انجام می‌دهد.
۲) مرتب‌سازی انتخابی ـ عمل مرتب‌سازی نزولی انجام می‌دهد.
۳) مرتب‌سازی انتخابی ـ عمل مرتب‌سازی غیر نزولی انجام می‌دهد.
۴) مرتب‌سازی حبابی ـ عمل مرتب‌سازی نزولی انجام می‌دهد.

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



۶۳ـ یک الگوریتم مرتب‌سازی را «خسته‌کننده» می‌گوییم اگر جایگشتی از $n$ عنصر مجزا از هم به عنوان ورودی وجود داشته باشد که یک عنصر خاص از آن ورودی توسط الگوریتم مورد نظر در مجموع $\Omega(n)$ بار با بقیه‌ی عناصر مقایسه کند. کدام یک از الگوریتم‌های زیر خسته‌کننده نیست؟
(مهندسی کامپیوتر ـ سراسری ۸۸)

۱) Insertion sort ۲) Merge sort ۳) Bubble sort ۴) Heap sort

۶۴ـ یک مجموعه $A$ از $n$ عدد مجزا از هم داده شده است. می‌خواهیم بدانیم آیا برای همه‌ی $x,y,z \in A$ رابطه‌ی زیر برقرار است؟
$x+y>z, x+z>y, and y+z>x$ بهترین راه حل این مسئله از چه مرتبه‌ای است؟
(مهندسی کامپیوتر ـ سراسری ۸۸)

۱) $O(n)$ ۲) $O(n^2)$ ۳) $O(n\log n)$ ۴) $O(n^2\log n)$

۶۵ـ $A$ یک آرایه $n$ عضوی با اعضای غیرتکراری و $B$ یک آرایه $m$ عضوی با اعضای غیرتکراری هستند و $m<n$ است می‌خواهیم آرایه $C$ را طوری بیابیم که اعضاء $A$ و $B$ در آن وارد شده باشد و عضو تکراری هم نداشته باشیم. بهترین الگوریتم برای این کار چه زمانی نیاز دارد؟
(مهندسی کامپیوتر ـ سراسری ۸۸)

۱) زمان $m\log m + n\log n + m + n$ ۲) زمان $(m+n)\log(m+n)$
۳) زمان $(m+n)\log n$ ۴) زمان $(m+n)\log m$

۶۶ـ می‌خواهیم $k$ فایل مرتب $f_1$ تا $f_k$ را در هم ادغام کنیم و یک فایل مرتب بسازیم. فایل $f_i$ به اندازه‌ی $n_i$ رکورد دارد و فایل خروجی هم به اندازه‌ی $n=\sum_{i=1}^{k} n_i$ رکورد خواهد داشت. با هر بار خواندن از یک فایل و نوشتن در انتهای یک فایل می‌توانیم بلوکی به اندازه‌ی $r$ رکورد از یک فایل را بخوانیم یا در فایل خروجی بنویسیم. فایل‌ها همه ترتیبی هستند، یعنی هر جا که باشیم فقط بلوک بعدی را می‌توانیم بخوانیم یا در انتهای فایل خروجی بنویسیم. تعداد کل خواندن بلوک‌ها و تعداد کل نوشتن بلوک‌ها چند تاست؟ در صورتی که حافظه اصلی برای $k+1$ بلوک ظرفیت داشته باشد.
(مهندسی کامپیوتر ـ سراسری ۸۸)

۱) خواندن و نوشتن هر دو برابر $\left\lceil \frac{n}{r} \right\rceil$
۲) خواندن $\left\lceil \frac{n}{r} \right\rceil$ و نوشتن $\sum_{i=1}^{k}\left\lceil \frac{n_i}{r} \right\rceil$
۳) خواندن و نوشتن هر دو برابر $\sum_{i=1}^{k}\left\lceil \frac{n_i}{r} \right\rceil$
۴) خواندن و نوشتن هر دو برابر $\sum_{i=1}^{k}\left(1+\left\lceil \frac{n_i}{r} \right\rceil\right)$

۶۷ـ احتمال این‌که Randomized –Quick sort به زمان $\Omega(n^2)$ نیاز داشته باشد تا $n$ عنصر مجزا از هم را مرتب کند چقدر است؟ (در Randomized –Quick sort، محور به صورت تصادفی و با احتمال یکسان یکی از عناصر انتخاب می‌شود و بقیه‌ی الگوریتم مانند Quick sort عادی است.) بهترین جواب را انتخاب کنید.
(مهندسی کامپیوتر ـ سراسری ۸۸)

۱) حداقل $\Pi_{i=1}^{n-1}\frac{i}{n-i}$ ۲) حداقل $\frac{1}{(n^n)}$ ۳) حداقل $\frac{1}{(n!)}$ ۴) حداقل $\frac{1}{(n^2)}$

۶۸ـ فرض کنید تعدادی داده داریم که به صورت نزولی مرتب شده‌اند. با اجرای یک الگوریتم مرتب‌سازی صعودی روی آن، بعد از دو مرحله اجراء داده‌ها به صورت زیر در آمده است:
(علوم کامپیوتر ـ سراسری ۸۸)

**48,59,61,77,11,15,19,26,1,5**

الگوریتم اجرا شده چه نوع sort‌ای می‌باشد.

۱) Heap sort ۲) Merge sort ۳) Quick sort ۴) Insertion sort

۶۹ـ اگر $n$ عدد صحیح داشته باشیم که یکی از آن‌ها $x$ باشد، الگوریتمی که تشخیص دهد دو عدد در این اعداد وجود دارد که مجموع این دو عدد دقیقاً $x$ می‌باشد، دارای کدام پیچیدگی است؟
(علوم کامپیوتر ـ سراسری ۸۸)

۱) $O(\lg n)$ ۲) $O(n)$ ۳) $O(n\lg n)$ ۴) $O(n^2)$

۷۰ـ Heap sort، Quick sort و Merge sort را به عنوان روش‌های مرتب‌سازی در نظر بگیرید. کدام‌یک از روش‌های فوق پایدار (stable) می‌باشند؟
(مهندسی کامپیوتر ـ آزاد ۸۸)

۱) Merge sort و Heap sort ۲) فقط Merge sort
۳) Merge sort و Quick sort ۴) Merge sort و Quick sort و Heap sort

۷۱ـ کدام یک از الگوریتم‌های مرتب‌سازی، یک آرایه‌ای که تقریباً مرتب شده است را سریع‌تر مرتب می‌نماید؟
(مهندسی فناوری اطلاعات (IT) و مهندسی کامپیوتر ـ آزاد ۸۸)

۱) Insertion sort ۲) Quick sort ۳) Bubble sort ۴) Selection sort

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



۷۲ـ می‌دانیم که هزینه‌ی الگوریتم مرتب‌سازی درجی (Insertion Sort) برای مرتب‌سازی یک آرایه‌ی A با n عنصر متناسب با تعداد «وارونگی» (inversion) های عناصر آن آرایه است. زوج (i, j) را یک عدد وارونگی می‌گوییم اگر $i < j$ و $A[i] > A[j]$ با فرض احتمال این که یک زوج اندیس دلخواه از A یک وارونگی باشد برابر $\frac{1}{2}$ است، میانگین تعداد وارونگی‌های یک آرایه‌ی A با عناصر متمایز چه قدر است؟ (مهندسی کامپیوتر ـ سراسری ۸۹)

۱) $\frac{n^2-n}{2}$ ۲) $\frac{n^2}{2}$ ۳) $\frac{n^2}{4}$ ۴) $\frac{n^2-n}{4}$

۷۳ـ در الگوریتم مرتب‌سازی آرایه‌ی A با n عنصر فرض کنید $b > 1$ یک عدد ثابت است. هم چنین فرض کنید که هزینه‌ی مقایسه‌ی دو عنصر A[i] و A[j]، یا تعویض آن‌ها، اگر $|i-j| \le b$ برابر صفر (خیلی کم) و در غیر این صورت برابر 1 (خیلی زیاد) است. توجه کنید که با این فرض، هزینه‌ی مرتب‌سازی درجی، حبابی (Bubble sort) برابر O(1) می‌شود. چون فقط عناصر مجاور را مقایسه و تعویض می‌کنند. با این فرض هزینه‌ی مرتب‌سازی ادغامی (Merge sort) در بدترین حالت چه قدر است؟ (بهترین جواب را انتخاب کنید) (بدیهی است که اگر T(n) زمان اجرا باشد داریم: $n < b$ و $T(n) = 1$.) (مهندسی کامپیوتر ـ سراسری ۸۹)

۱) $O(n \lg(n/b))$ ۲) $O(n/b \lg(n/b))$ ۳) $O(n \lg n)$ ۴) $O(n \lg n)$

۷۴ـ فرض کنید می‌خواهیم یک لیست حاوی عناصر (از چپ به راست) 2 و 4 و 5 و 3 و 1 را به لیست با عناصر (از چپ به راست) 1 و 2 و 3 و 4 و 5 تبدیل کنیم. تنها عملیات مجاز جابجا کردن درایه‌های متوالی (پشت سر هم) با یکدیگر است. حداقل چند جابجایی لازم است؟ (مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۹)

۱) 5 ۲) 4 ۳) 6 ۴) 7

۷۵ـ در کدام‌یک از الگوریتم‌های مرتب‌سازی پیچیدگی الگوریتم آن به‌صورت زیر است؟ (n توانی از 2 است) (مهندسی کامپیوتر ـ آزاد ۸۹)

$$\begin{cases} T(n) = 2T(\frac{n}{2}) + n - 1 \\ T(1) = 0 \end{cases}$$

۱) Bubble sort
۲) Merge sort
۳) Selection sort
۴) Quick sort

۷۶ـ چنانچه در الگوریتم quick sort، الگوریتم بخش‌بندی (partition) زمان ثابت C نیاز داشته باشد، زمان اجرای مرتب‌سازی سریع در حالت تصادفی (داده‌های تصادفی) چیست؟ (مهندسی کامپیوتر ـ سراسری ۹۰)

۱) $\Theta(n)$ ۲) $\Theta(n^2)$ ۳) $\Theta(\lg n)$ ۴) $\Theta(n \lg n)$

۷۷ـ دو لیست نامرتب A و B هر کدام با n عنصر داده شده‌اند. می‌خواهیم به صورت بهینه لیست $A \cap B$ را به دست آوریم. زمان اجرای بهینه‌ی این الگوریتم در دو حالت میانگین و بدترین حالت چقدر است؟ (مهندسی کامپیوتر ـ سراسری ۹۰)

۱) میانگین $\Theta(n \lg n)$ و بدترین حالت $\Theta(n^2)$
۲) میانگین $\Theta(n)$ و بدترین حالت $\Theta(n \lg n)$
۳) میانگین $\Theta(n)$ و بدترین حالت $\Theta(n)$
۴) میانگین $\Theta(n \lg n)$ و بدترین حالت $\Theta(n \lg n)$

۷۸ـ بر روی آرایه‌ی A با N عنصر A[1] تا A[N] که در ابتدا حاوی جایگشتی از اعداد 1 تا N است، رویهٔ زیر را اجرا می‌کنیم:

```
MAYBE SORT(A,N)
1       k ← 0
2       repeat
3           k ← k + 1
4           for i = 1 to N
5              do B[i] ← A[A[i]]
6           for i == 1 to N
7              do A[i] ← B[i]
8       until ∀1 ≤ i ≤ N A[i] = i
9       return k
```

کدام یک از گزینه‌های زیر درست است؟ (مهندسی کامپیوتر ـ سراسری ۹۰)

۱) مقدار خروجی دقیقاً مقدار N است.
۲) مقدار خروجی دقیقاً مقدار $N-1$ است.
۳) اجرای این برنامه ممکن است هیچ وقت تمام نشود.
۴) خروجی این برنامه هیچ وقت بیش‌تر از $\lceil \log N \rceil$ نیست.

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۷۹ـ دنباله‌ای از اعداد صحیح به تعداد $n$ وجود دارد. اگر برخی از این اعداد تکراری باشند و در کل $k$ عدد غیرتکراری وجود داشته باشد آنگاه با چه مرتبه زمانی مناسبی می‌توان این دنباله را مرتب کرد؟** (علوم کامپیوتر ـ سراسری ۹۰)

۱) $O(n\log n)$ ۲) $O(n\log k)$ ۳) $O(k\log n)$ ۴) $O(k\log k)$

**۸۰ـ اگر در الگوریتم مرتب‌سازی ادغامی، به جای تقسیم آرایه به دو بخش مساوی، آرایه را هر بار از یک مکان تصادفی تقسیم کنیم، میانگین زمان اجرای این الگوریتم از چه مرتبه‌ای است؟ دقیق‌ترین جواب را انتخاب کنید.** (مهندسی نرم‌افزار ـ دکتری ۹۱)

۱) $O(n)$ ۲) $O(n^2)$

۳) مانند حالت معمول خواهد بود. ۴) از حالت معمول سریع‌تر خواهد بود.

## پاسخنامه تست‌های طبقه‌بندی شده کنکوری فصل چهاردهم

**۱ـ گزینه «۱»** در Quick sort، مرتبهٔ زمانی حالت متوسط $O(n\log n)$ و بدترین حالت $O(n^2)$ است. در Heap sort و Merge sort، مرتبهٔ زمانی حالت متوسط و بدترین حالت $O(n\log n)$ است و در Insertion sort مرتبهٔ زمانی حالت متوسط و بدترین حالت $O(n^2)$ می‌باشد.

**۲ـ گزینه «۳»** الگوریتم‌های درجی و ادغامی متعادل هستند.

**۳ـ گزینه «۱»** فرض کنیم آرایه ورودی، به صورت نزولی مرتب شده است و می‌خواهیم با این الگوریتم آرایه به صورت صعودی مرتب شود. در دستور اول از عنصر $k+1$ تا $N$ و در دستور دوم از عنصر $1$ تا $k+L$ به صورت صعودی مرتب می‌شود. حال اگر $k\le L$ باشد $k$ عنصر اول همگی به اندیس‌های $k+1$ تا $L$ منتقل شده و با اجرای دستور سوم کاملاً آرایه صعودی خواهد شد.

**۴ـ گزینه «۳»** بعد از مرتب کردن ستونی، ترتیب سطری از بین نمی‌رود؛ زیرا اگر عنصر $A[i,j]$ $(i=1,\dots,N)$ $(j=1,\dots,m)$ از $A[i+1,j]$ بزرگ‌تر باشد جای این دو عنصر جا به جا می‌شود. چنانچه $A[i+1,j]$ از $A[i,j-1]$ کوچک‌تر باشد، چون قبلاً سطرها مرتب شده است و $A[i+1,j-1]$ از $A[i+1,j]$ کوچک‌تر است، در نتیجه جای $A[i,j-1]$ و $A[i+1,j-1]$ نیز جا به جا شده و هر دو سطر باز هم مرتب خواهند ماند.

**۵ـ گزینه «۱»** می‌دانیم Insertion Sort در بهترین حالت دارای مرتبهٔ اجرایی $O(n)$ است. بهترین حالت زمانی است که آرایه از قبل مرتب باشد، سایر مرتب‌سازی‌ها در این حالت حداقل مرتبهٔ اجرایی $O(n\log n)$ دارند و چون $n$ خیلی بزرگ‌تر از $k$ است، پس تقریباً آرایه مرتب می‌باشد.

**۶ـ گزینه «۲»** در Quick sort هر گاه آرایهٔ ورودی از قبل مرتب باشد، بدترین حالت اتفاق می‌افتد و می‌دانیم زمان اجرای الگوریتم Quick sort در بدترین حالت $O(n^2)$ است.

**۷ـ گزینه «۳»** با فرض عدد 9 به عنوان لولا، تمام اعداد کوچک‌تر از آن به قبل و تمام اعداد بزرگ‌تر از آن به بعد از 9 منتقل می‌شوند، بنابراین داریم:

$$\underset{\downarrow}{9},\underset{\uparrow}{10},\underset{\uparrow}{8},\underset{\uparrow}{7},\underset{\uparrow}{6},\underset{\downarrow}{15},\underset{\uparrow}{3}$$

اعداد $\downarrow$ به بعد و اعداد $\uparrow$ به قبل از 9 منتقل می‌شوند.

از میان گزینه‌ها گزینه سوم فقط اعداد 10 و 15 را بعد از 9 قرار داده است.

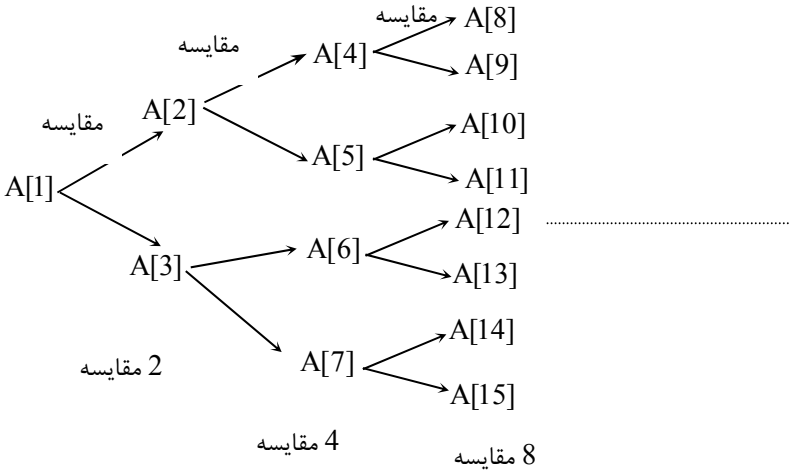
**۸ـ گزینه «۴»** در روال ذکر شده ابتدا مقدار i در j قرار گرفته و سپس در حلقه while مقدار A[j] با A[j-1] مقایسه شده و اگر A[j] از A[j-1] کوچک‌تر باشد، مقدار A[j] با A[j-1] عوض شده و j = j-1 می‌شود یا به عبارت دیگر مقدار A[i] در لیست مرتب A[0]…A[i-1] درج می‌شود.

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



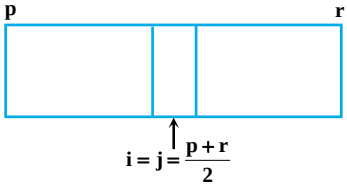
**۹ـ گزینه «۱»** برای بررسی heap بودن یک آرایه داده شده، می‌توان به صورت زیر مقایسه عناصر را انجام داد:



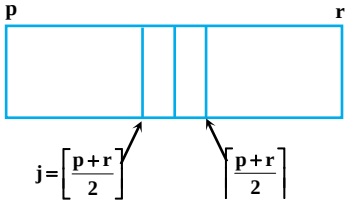

بنابراین تعداد مقایسه‌ها از مرتبه $O(n)$ می‌باشد.

**۱۰ـ گزینه «۴»** در این الگوریتم در هر بار حلقه‌های repeat until فقط یک بار اجرا می‌شوند و در هر بار اجرا، شمارنده j یک واحد کاهش و شمارنده i یک واحد افزایش می‌یابد. بنابراین در نهایت زمانی شرط if برقرار نخواهد بود که $i >= j$ باشد. در نتیجه مقدار نهایی j به صورت زیر خواهد بود:

اگر $p+r$ زوج باشد



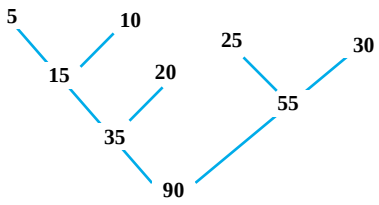

اگر $p+r$ فرد باشد




**۱۱ـ گزینه «۴»** با توجه به این که آرایه‌های A و C مرتب شده صعودی می‌باشند، بنابراین مرتب‌سازی Quick Sort دارای زمان $O(n^2)$ می‌باشد اما مرتب‌سازی insertion سریع‌تر از بقیه عمل می‌کند (در زمان $O(n)$).

**۱۲ـ گزینه «۴»** از روشی شبیه به روش ادغام دو لیست مرتب استفاده می‌کنیم؛ یعنی به دو اشاره‌گر نیاز داریم که در ابتدا به اول لیست‌ها اشاره می‌کنند سپس برای اشتراک، اشاره‌گری که به عنصر کوچکتر از اشاره‌گر دیگر اشاره می‌کند را به جلو می‌بریم تا به یک عنصر مساوی یا بزرگتر از دیگری برسیم حال در مورد اشتراک این عضو را در مجموعه جدید کپی می‌کنیم و در مورد اجتماع، تمام عناصر دیگر از هر دو اشاره‌گر را کپی می‌کنیم و این عنصر را فقط از یکی از اشاره‌گرها کپی می‌نماییم و سپس دوباره همین کار تکرار می‌گردد تا هر دو اشاره‌گر به انتهای لیست خود برسند بنابراین در زمان $\theta(m+n)$ این عمل قابل اجرا است و با توجه به این که $m < n$ بنابراین زمان $\theta(n)$ خواهد بود.

**۱۳ـ گزینه «۱»** برای این که کمترین تعداد جابجایی را داشته باشیم در هر زمان کوچکترین فایل‌ها را با هم ادغام می‌کنیم. بنابراین با توجه به شکل روبرو کمترین تعداد جابجایی‌ها عبارتست از:




$$15+35+55+90=195$$

**۱۴ـ گزینه «۱»** می‌دانیم در mergesort یک آرایهٔ n عضوی به n قسمت به طول یک تقسیم و سپس هر دو قسمت مجاور به صورت مرتب شده با هم ادغام می‌شوند و این روند تا مرتب شدن کل آرایه ادامه می‌یابد. در بدترین حالت برای ادغام دو لیست با $\frac{n}{2}$ عضو $(n-1)$ مقایسه نیاز است. بنابراین بدترین حالت برای مرتب‌سازی آرایه n عضوی از معادلهٔ $W(n)=2W(\frac{n}{2})+n-1$ به دست می‌آید، که جواب برابر: $W(n)=n\log_2^n-n+1$ است.

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۱۵ـ گزینه «۴»** حداقل تعداد مقایسه‌ها در بدترین حالت برابر $\lceil \log_2 8! \rceil$ می‌باشد:

$$\lceil \log_2 8! \rceil = \lceil \log_2 8 + \log_2 7! \rceil = 3 + \lceil \log_2 7! \rceil = 3 + 13 = 16$$

---

**۱۶ـ گزینه «۱»** اگر تعداد عناصری که در لیست‌های $L_1, L_2, \ldots, L_K$ هستند را با $n$ نمایش دهیم، برای ادغام این $K$ لیست با استفاده از آرایه و لیست دو طرفه و درخت دودویی جستجو، در بدترین حالت مرتبهٔ اجرایی $O(n^2)$ است، اما با استفاده از درخت heap، $n$ درج و $n$ حذف داریم و باتوجه به اینکه عمل درج و حذف در heap با مرتبه اجرایی $O(\log n)$ انجام می‌شود، بنابراین مرتبهٔ اجرایی ادغام این $k$ لیست، $O(n \log n)$ خواهد بود.

---

**۱۷ـ گزینه «۱»** لیست زیر را در نظر بگیرید:

$$n, n-1, n-2, n-3, \ldots, 2, 1$$

در این حالت عنصر $n$ با تمام $n-1$ عنصر سمت راست خود وارونگی می‌سازد.

عنصر $n-1$ با تمام $n-2$ عنصر سمت راست خود وارونگی می‌سازد.

عنصر $n-2$ با تمام $n-3$ عنصر سمت راست خود وارونگی می‌سازد.

$\vdots$

عنصر 2 با عنصر 1 یک وارونگی می‌سازد.

بنابراین تعداد کل وارونگی‌ها برابر است با:

$$1 + 2 + 3 + \cdots (n-1) = \frac{n(n-1)}{2}$$

با توجه به این که تعداد کل زوج عناصر از یک مجموعه $n$ عنصری برابر است با:

$$\binom{n}{2} = \frac{n(n-1)}{2}$$

بنابراین حداکثر تعداد ممکن وارونگی $\frac{n(n-1)}{2}$ می‌باشد.

به عنوان مثال لیست روبرو را در نظر بگیرید:

$$5, 4, 3, 2, 1$$

در این حالت، وارونگی‌ها عبارتند از:

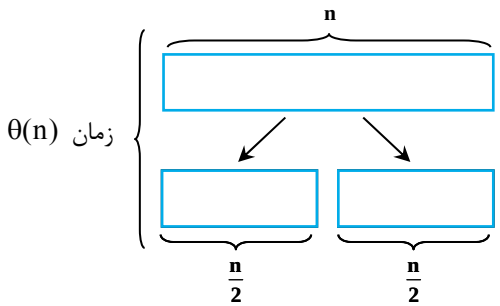
$(5,4), (5,3), (5,2), (5,1)$
$(4,3), (4,2), (4,1)$
$(3,2), (3,1)$
$(2,1)$

---

**۱۸ـ گزینه «۱»**

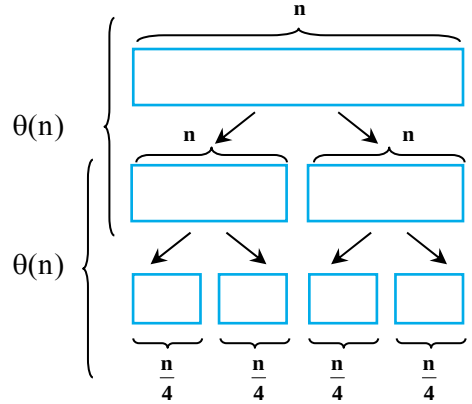
دقت کنید در حالتی که $k = n$ باشد در حقیقت همان مسئله مرتب‌سازی را خواهیم داشت و می‌دانیم بهترین الگوریتم برای مرتب‌سازی در همه حالات دارای زمان $O(n \log n)$ می‌باشد، به عبارت دقیق می‌توان از الگوریتم زیر استفاده نمود.

ابتدا در زمان $\theta(n)$ عنصر میانه آرایه تعیین می‌گردد سپس عناصر آرایه در زمان $\theta(n)$ با توجه به عنصر میانه به دو زیر آرایه تقسیم می‌شوند.




سپس در هر یک از دو زیر آرایه همین عمل انجام می‌شود و چهار زیر آرایه با اندازه $\frac{n}{4}$ ایجاد می‌گردد.

و همین عمل تکرار می‌گردد تا در نهایت آرایه‌هایی با اندازه $\frac{n}{k}$ ایجاد گردد بنابراین با توجه به این که تعداد مراحل برابر با $O(\log_2^k)$ می‌باشد و در هر مرحله به زمان $\theta(n)$ نیاز است بنابراین، در کل زمان عبارتست از: $O(n \log_2^k)$




www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۱۹ـ گزینه «۴»** در بدترین حالت عنصر محوری انتخاب شده از $\sqrt{n}$ عنصر بزرگتر و از سایر عناصر(یعنی $n-\sqrt{n}$ عنصر باقیمانده) کوچکتر یا مساوی است و بنابراین دو زیرمسئله با اندازه‌های $\sqrt{n}$ و $n-\sqrt{n}$ ایجاد می‌شوند: $T(n)=T(\sqrt{n})+T(n-\sqrt{n})+O(n)$

مکان عنصر محوری در بدترین حالت را می‌توان به صورت زیر در نظر گرفت:

همچنین برای افراز آرایه همواره به زمان $O(n)$ نیاز می‌باشد.

---

**۲۰ـ گزینه «۱»** در پیاده‌سازی یک هیپ به کمک آرایه‌هایی که اندیس آنها از یک شروع می‌شود، عمق گره‌ای که در اندیس $i$ام قرار دارد برابر $[\log i]$ می‌باشد. اگر اندیس آن از صفر شروع شود، عمق گره $i$ام برابر با $[\log i]+1$ می‌شود.

---

**۲۱ـ گزینه «۱»** گزینه‌ی دوم نادرست است، زیرا که Quicksort در بدترین حالت، مرتبه‌ی زمانی $n^2$ دارد. گزینه‌ی سوم نادرست است، زیرا اگر آرایه از قبل مرتب شده باشد (به صورت صعودی) مرتب‌سازی درجی دارای مرتبه $O(n)$ می‌شود که در نتیجه کارآیی آن از مرتب‌سازی سریع بهتر می‌شود. گزینه‌ی چهارم نیز نادرست می‌باشد، زیرا مرتب‌سازی ادغامی از نظر زمان بهتر از مرتب‌سازی انتخابی است.

---

**۲۲ـ گزینه «۲»** توجه کنید که در اینجا منظور، تعداد مقایسه‌ها در بهترین حالت نیست، چرا که در این صورت این تعداد برای مرتب‌سازی درجی برابر $n$ است بلکه کمترین پیچیدگی در مقایسه‌ها برای مرتب‌سازی مورد نظر است که $o(nlng)$ می‌باشد.

دقت کنید که: $\log n! \simeq n\log n$ (البته در اینجا منظور مرتب‌سازی‌های مبتنی بر مقایسه می‌باشد).

---

**۲۳ـ گزینه «۳»** ابتدا با استفاده از الگوریتم SELECTION در زمان $O(n)$، $k$ امین عنصر را بدست آورده و آرایه را با استفاده از Partition افراز می‌کنیم (در زمان $O(n)$). سپس $k$ تای اول را که $k$ تا کوچکترین عناصر می‌باشند با استفاده از یکی از روش‌های مرتب‌سازی مرتب می‌کنیم. در نتیجه کل زمان برابر با $O(n)+O(klogk)$ می‌باشد.

---

**۲۴ـ گزینه «۲»** دقت کنید که بدترین حالت زمانی اتفاق می‌افتد که آرایهٔ ورودی به شکل $[2,3,\cdots,n-1,1]$ باشد که در این حالت با $n-1$ جابجایی تمام عناصر در مکان صحیح خود قرار می‌گیرند. به عنوان مثال لیست زیر را در نظر بگیرید:

| A[1] | A[2] | A[3] | A[4] | A[5] |
|---|---|---|---|---|
| 2 | 3 | 4 | 5 | 1 |

نحوه اجرای الگوریتم به صورت زیر می‌باشد:

$i=1, A[1]<>1 \Rightarrow swap(A[1], A[A[1]])$

بنابراین $A[2]$ و $A[1]$ تعویض می‌گردند:

| | | | | |
|---|---|---|---|---|
| 3 | 2 | 4 | 5 | 1 |

$A[1]<>1 \Rightarrow swap(A[1], A[3])$

| | | | | |
|---|---|---|---|---|
| 4 | 2 | 3 | 5 | 1 |

$A[1]<>1 \Rightarrow swap(A[1], A[4])$

| | | | | |
|---|---|---|---|---|
| 5 | 2 | 3 | 4 | 1 |

$A[1]<>1 \Rightarrow swap(A[1], A[5])$

| | | | | |
|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 |

بنابراین می‌توان گفت در هر بار اجرای حلقه while و انجام یک تعویض، حداقل یک عنصر در مکان صحیح خود قرار می‌گیرد و در نهایت با آخرین تعویض دو عنصر در مکان صحیح خود قرار خواهند گرفت بنابراین، حداکثر تعداد تعویض‌ها برابر $n-1$ خواهد بود.

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۲۵ـ گزینه «۴»** زمان مرتب‌سازی heapsort با هر نوع ورودی $O(n\log_2^n)$ می‌باشد.

---

**۲۶ـ گزینه «۳»** برای جستجوی x، اگر A را مرتب کنیم در بدترین حالت، هزینه $O(n^2)$ است و پیمایش با هزینهٔ $O(n)$ انجام می‌شود. اگر از heap استفاده کنیم هزینهٔ درج در heap، $O(\log n)$ است. چون n درج داریم، هزینه $O(n\log n)$ می‌شود و جستجو با هزینه $O(n)$ خواهد بود و با استفاده از درخت جستجوی دودویی متوازن هزینه $O(n^3)$ است، امّا، اگر ابتدا میانهٔ A را پیدا کنیم چون در A از اعداد 1 تا n ذخیره شده میانه، $(1+n)\text{div }2$ می‌شود، با استفاده از عمل بخش‌بندی از سمت چپ شروع کرده و اولین بزرگ‌تر از میانه را می‌یابیم و از سمت راست اولین کوچک‌تر، سپس این دو را با هم تعویض می‌کنیم تا به اندیس $(1+n)\text{div }2$ برسیم. اگر این اندیس مقدارش با $(1+n)\text{div }2$ یکی باشد پس x در نیمهٔ بالایی است و اگر از این مقدار بزرگ‌تر باشد x در نیمهٔ پایانی است و همین عملیات را تا یافتن x تکرار می‌کنیم. حداکثر هزینه در این حالت $O(n\log n)$ خواهد بود.

---

**۲۷ـ گزینه «۲»** با توجه به این نکته که مرتب‌سازی مورد نظر به صورت نزولی انجام می‌شود، اگر لیست داده شده از ابتدا به صورت نزولی مرتب باشد، آنگاه پس از اجرای pivot لیست ورودی تغییر نمی‌کند. بنابراین دو زیر لیست ایجاد می‌گردد که یکی تهی است و دیگری شامل $n-1$ عنصر به صورت نزولی می‌باشد که با استفاده از insertion sort در زمان $\theta(n)$ مرتب خواهد شد.

---

**۲۸ـ گزینه «۳»** در بدترین حالت عنصر میانی تعیین شده در بین $2\sqrt{n}+1$ عنصر ابتدایی از $\sqrt{n}$ عنصر بزرگتر و از سایر عناصر یعنی $n-\sqrt{n}$ عنصر دیگر کوچکتر یا مساوی است و بنابراین زمان الگوریتم برابر است با:

$$T(n)=T(\sqrt{n})+T(n-\sqrt{n})+\underbrace{O((\sqrt{n})^2)}_{\downarrow}$$

زمان مرتب‌سازی insertion در بدترین حالت و انجام partition

---

**۲۹ـ گزینه «۱»** تعداد برگ‌های یک درخت تصمیم‌گیری برای مرتب‌سازی n کلید برابر $n!$ و بنابراین عمق آن حداقل برابر $\log_2 n!$ است.

---

**۳۰ـ گزینه «۳»** فرض کنید یک مجموعه عدد داریم و بین آنها یک سری مسابقات حذفی داریم. آن‌گاه بزرگ‌ترین عدد آرایه، عددی است که به قهرمانی رسیده است. دومین عدد بزرگ یکی از عضوهایی است که به نفر اول باخته است. پس n-1 مقایسه برای یافتن قهرمان و $\lceil\log n\rceil-1$ مقایسه، برای مقایسه بین بازنده‌هایی که به نفر اول باخته‌اند و پیدا کردن بزرگ‌ترین عضو در آنها نیاز داریم پس در مجموع $n+\left\lceil\log_2^n\right\rceil-2$ مقایسه نیاز داریم.

---

**۳۱ـ گزینه «۳»** با توجه به مقادیر ذکر شده، الگوریتم مرتب‌سازی مورد نظر نمی‌تواند حبابی bubble و انتخابی Selection باشد، زیرا در هر دوی این الگوریتم‌ها در اولین تکرار کوچک‌ترین مقدار یعنی 0 در مکان اول قرار می‌گیرد و نیز می‌تواند Quicksort باشد. زیرا در اولین تکرار باید به عنوان یک عنصر محوری تمامی مقادیر سمت راست از آن عنصر محوری بزرگ‌تر و تمامی مقادیر سمت چپ از آن عنصر محوری کوچک‌تر باشد که ترتیب ذکر شده این چنین نیست ولی می‌تواند در چهارمین تکرار مرتب‌سازی درجی Insertion باشد.

---

**۳۲ـ گزینه «۳»** مرتبه زمانی در بدترین حالت برای مرتب سازی سریع $O(n^2)$ و برای heapsort برابر $O(n\log n)$ می‌باشد.

---

**۳۳ـ گزینه «۱»** مرتب‌سازی‌های Heap و merge و Insertion در حالت متوسط و در بدترین حالت دارای یک پیچیدگی زمانی هستند ولی در Quick sort در بدترین حالت، پیچیدگی $O(n^2)$ و در حالت متوسط، پیچیدگی $O(n\log n)$ می‌باشد.

---

**۳۴ـ گزینه «۳»**

**روش اول:** اگر $k=1$ باشد لیست مرتب است و نیاز به هیچ عملیاتی نیست.

اگر $k=2$ باشد آنگاه عناصر زوج آرایه یک لیست مرتب و عناصر فرد آرایه نیز یک لیست مرتب را می‌سازند که با ادغام آن‌ها در زمان $O(n)$ یک لیست مرتب ایجاد می‌گردد. اگر $k=4$ باشد آنگاه می‌توان چهار آرایه مرتب زیر را از آرایه اصلی استخراج نمود:

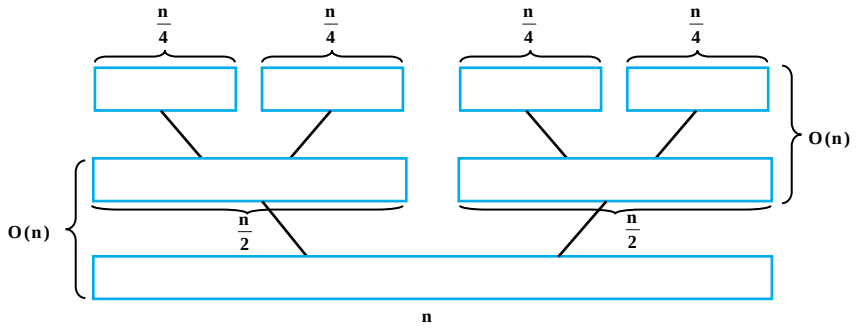
| $\frac{n}{4}$ | $\frac{n}{4}$ | $\frac{n}{4}$ | $\frac{n}{4}$ |
|---|---|---|---|
| A[1] A[5] A[9] … | A[2] A[6] A[10] … | A[3] A[7] A[11] … | A[4] A[8] A[12] … |

www.modaresanesharif.ac.ir

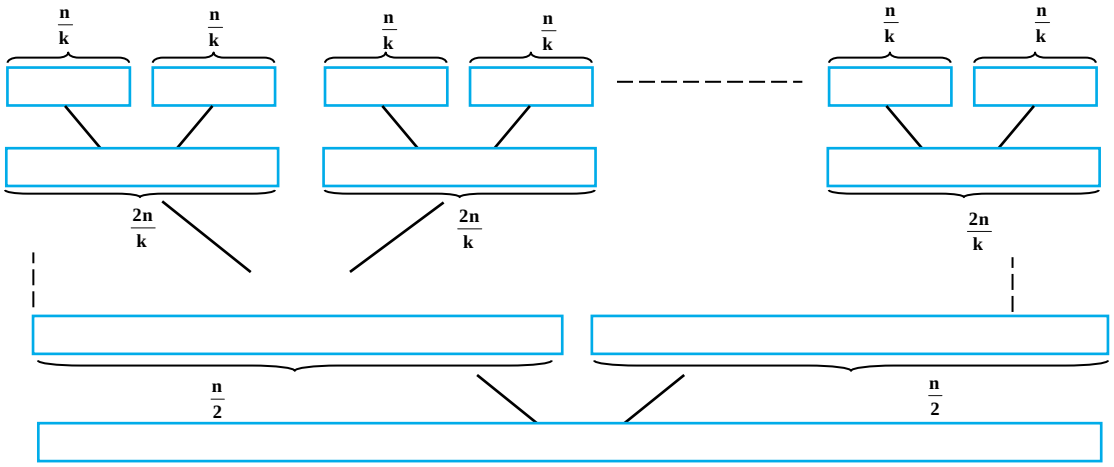
www.modaresanesharif.ac.ir



که با ادغام دوبه‌دوی آن‌ها به صورت زیر (مانند الگوریتم merge sort) یک آرایه مرتب ایجاد می‌گردد:




و در حالت کلی می‌توان آرایه‌های مرتب با اندازه $\frac{n}{k}$ را از آرایه اصلی استخراج نمود و در نهایت با $k$ عمل ادغام به یک آرایه مرتب $n$ عنصری رسید:




آرایه مرتب نهایی با **n** عنصر

**روش دوم:** زمان حل این مسئله باید به $k$ بستگی داشته باشد بنابراین، گزینه‌های ۱ و ۴ نادرست‌اند و باید دقت کنیم اگر $k$ برابر یک باشد در حقیقت آرایه مرتب است و بنابراین هیچ زمانی نیاز نداریم که این مطلب فقط در مورد گزینه سوم صادق است.

---

**۳۵ـ گزینه «۴»** در حالتی که از روش Partition استفاده نماییم مرتبه الگوریتم $O(nlogn)$ خواهد بود (هر بار یکی از لیست‌های ایجاد شده حذف می‌شوند) اما سایر گزینه‌ها علاوه بر زمان $O(n \log n)$ به یک جست وجو نیز نیاز دارند.

---

**۳۶ـ گزینه «۳»** در این قطعه کد داده شده آرایه به ترتیب صعودی مرتب می‌شود. در نتیجه $a[j] <= a[j+1]$ خواهد بود. $(item <= j < n)$

---

**۳۷ـ گزینه «۴»** زمان الگوریتم Merge Sort در حالت کلی برابر $\theta(n \log_2^n)$ می‌باشد. با انجام تغییر مورد نظر در مرتبهٔ زمانی الگوریتم تغییری ایجاد نمی‌شود و زمان همان $\theta(n \log_2^n)$ خواهد بود.

---

**۳۸ـ گزینه «۱»** با حل رابطه و به دست آوردن کلاس پیچیدگی آن جواب به دست می‌آید.

$$t(n) = (n-1) + \frac{2}{n} = t(n-2) + \frac{2}{n-1} + \frac{2}{n} = \ldots = 2\sum_{1}^{n} \frac{1}{i} = 2Lnn$$

در نتیجه $t(n) \in O(Log\, n)$ لذا تنها پیچیدگی جستجوی دودویی این‌گونه است.

---

**۳۹ـ گزینه «۱»** کافی است داده‌های یکی را حذف کنیم و در دیگری اضافه کنیم. $m(Logm + Logn) = m\, Log\, mn$

---

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۴۰ـ گزینه «۴»** زمان مرتب‌سازی heap sort همواره از مرتبه $n \log n$ می‌باشد.

**۴۱ـ گزینه «۲»** در بین مرتب‌سازی‌های بیان شده تنها مرتب‌سازی درجی در حالت میانگین از مرتبه $\theta(n^2)$ می‌باشد.

**۴۲ـ هیچ‌کدام از گزینه‌ها صحیح نیست.** حداکثر هزینه درج هر عنصر در یک درخت BST شامل n عنصر برابر $O(n)$ می‌باشد و هزینه پیمایش همواره $O(n)$ می‌باشد.

**۴۳ـ گزینه «۳»** در حالت میانگین مرتبه زمانی مرتب‌سازی سریع برابر $\theta(n \log n)$ می‌باشد. برای لیست‌های مرتب شده، این الگوریتم از مرتبه $(n^2)$ است.

**۴۴ـ گزینه «۱»** مرتبه زمانی مرتب‌سازی ادغامی در همه حالات $\theta(n \log n)$ می‌باشد.

**۴۵ـ گزینه «۲»** الگوریتم‌های مرتب‌سازی حبابی و انتخابی همواره دارای مرتبهٔ زمانی $O(n^2)$ است و در مرتب‌سازی درجی اگر آرایه معکوس مرتب باشد، (بدترین حالت) مرتبهٔ زمانی $O(n^2)$ و در مرتب‌سازی سریع اگر آرایه مرتب باشد (بدترین حالت)، مرتبهٔ زمانی $O(n^2)$ است.

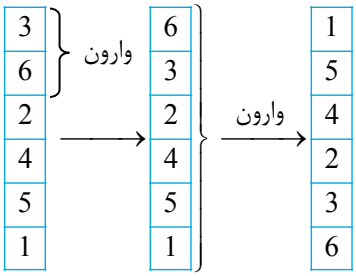
**۴۶ـ گزینه «۱»** در مرتب‌سازی درجی اگر آرایه از قبل مرتب باشد، در هر مرحله یک مقایسه و هیچ جابه‌جایی داریم، بنابراین مرتبهٔ اجرایی مرتب‌سازی درجی در بهترین حالت $O(n)$ می‌باشد.

**۴۷ـ گزینه «۴»** الگوریتم Quick sort در بهترین حالت و حالت متوسط (زمانی که آرایه نامرتب باشد) دارای مرتبهٔ زمانی $O(n \log n)$ است.

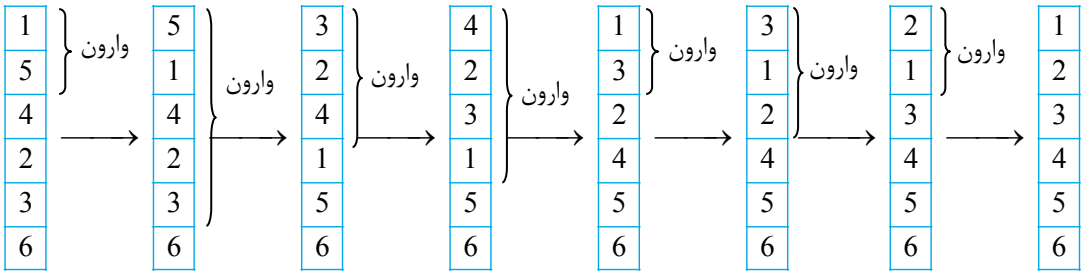
**۴۸ـ گزینه «۳»** الگوریتم مرتب‌سازی ادغامی در کلیهٔ حالات دارای مرتبهٔ زمانی $O(n \log n)$ می‌باشد.

**۴۹ـ گزینه «۲»** در الگوریتم مرتب‌سازی درجی وقتی عناصر آرایه از قبل مرتب باشد (بهترین حالت) مرتبهٔ اجرایی $O(n)$ است.

**۵۰ـ گزینه «۴»** در این الگوریتم در گام اول پس از یافتن بزرگترین عنصر با انجام یک عمل «وارون» آن را به ابتدای آرایه منتقل می‌نماییم و سپس با انجام یک عمل وارون دیگر کل آرایه معکوس می‌گردد و بزرگترین عنصر در جای صحیح خود قرار داده می‌شود. به عنوان مثال آرایه زیر را در نظر بگیرید.




سپس در هر مرحله عنصر بزرگ بعدی پیدا می‌شود و با انجام دو عمل وارون به مکان صحیح خود انتقال می‌یابد.




بنابراین همان‌گونه که مشخص است در بدترین حالت برای این که هر یک از $n-2$ عنصر ابتدایی در جای صحیح خود قرار گیرند. نیاز به 2 عمل وارون می‌باشد و برای 2 عنصر آخر حداکثر نیاز به یک عمل وارون می‌باشد بنابراین حداکثر تعداد اعمال وارون عبارتست از: $2\times(n-2)+1=2n-3$

نزدیکترین جواب به این مقدار 2n-2 است که در گزینه ۴ آمده است.

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۵۱ـ گزینه «۱»** به عنوان مثال آرایه زیر یک آرایه 8 عنصری 2- مرتب است.

| 8 | 1 | 9 | 2 | 10 | 3 | 11 | 4 |
|---|---|---|---|---|---|---|---|

که نسبت به آرایه 1- مرتب (یعنی آرایه مرتب) عنصر 8، چهار واحد اختلاف دارد.

---

**۵۲ـ گزینه «۱»** در این جا منظور تعداد تعویض‌ها در الگوریتم Partition است. که در هر فراخوانی یک عمل تعویض انجام می‌شود تا در نهایت i >= j شود، بنابراین $\left\lfloor \frac{n}{2} \right\rfloor$ عمل تعویض صورت می‌پذیرد. در اولین Partition با یک swap i≮j و بنابراین دو عنصر در جای درست خود قرار می‌گیرند. Partition بعدی هیچ swap اتفاق نمی‌افتد، در بعدی یکی رخ می‌دهد.

---

**۵۳ـ گزینه «۳»** به طور کلی روش Merge sort یک آرایه با n عنصر را با مرتبه زمانی $\theta(nLogn)$ مرتب می‌کند.

---

**۵۴ـ گزینه «۳»** در الگوریتم مذکور حلقهٔ for از 2 شروع شده و در حلقهٔ while تا زمانی که $A[j] < A[j-1]$ باشد، عناصر آرایه یکی به جلو شیفت داده می‌شود، (البته در این الگوریتم دستور $A[j-1] := Tmp$ جا افتاده است) بنابراین در حلقهٔ While درج عنصر $A[j]$ در عناصر $A[0]..A[j-1]$ انجام می‌شود.

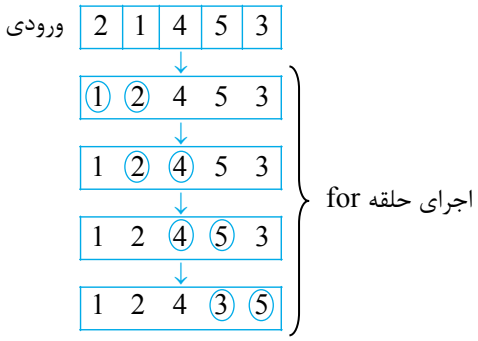
---

**۵۵ـ گزینه «۱»** ابتدا 16 با 50 و بعد با 21 و بعد با 33 جابه‌جا می‌شود و با ادامه این روند داریم:

$$A[1\cdots14] = 50,33,41,30,32,40,36,20,3,21,16,34,23$$

---

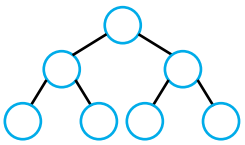
**۵۶ـ گزینه «۲»** در هر بار فراخوانی الگوریتم دو عنصر متوالی در صورت لزوم با یکدیگر تعویض می‌گردند. بنابراین مرتب‌سازی حبابی (bubble) انجام می‌پذیرد و پس از هر بار اجرای حلقه for یکی از عناصر در مکان صحیح خود قرار می‌گیرد و الگوریتم بر روی n-1 عنصر باقیمانده اجرا می‌گردد. به عنوان مثال مرحله اول اجرای الگوریتم بر روی یک لیست پنج عنصری به صورت زیر خواهد بود:




---

**۵۷ـ گزینه «۳»** تعداد برگ‌های یک درخت پر برابر است با $\left\lceil \frac{n}{2} \right\rceil$ بنابراین، اگر $h = 0$ باشد باید این رابطه بدست آید که فقط در گزینه سوم صادق است.

به عنوان مثال درخت heap زیر را در نظر بگیرید:



در این درخت 2 گره با عمق $h = 1$ وجود دارد که با توجه به گزینه‌ها داریم:

گزینه ۱ : $\left\lceil \frac{8}{2} \right\rceil = 4$ گزینه ۲ : $\frac{8}{2} = 4$ گزینه ۳ : $\left\lceil \frac{7}{4} \right\rceil = 2$ گزینه ۴ : $\left\lceil \frac{7}{2} \right\rceil = 4$

---

**۵۸ـ گزینه «۳»** عمق فراخوانی‌ها برابر $Lg2000$ می‌باشد که 11 خواهد بود. بنابراین با توجه به این‌که هر آدرس بازگشت به 4 بایت حافظه نیاز دارد در مجموع 44 بایت حافظه مصرف می‌شود.

---

**۵۹ـ گزینه «۳»** مقایسه‌ها در مرتب‌سازی حبابی (در یک لیست با n عنصر) همواره برابر $\frac{n(n-1)}{2}$ می‌باشد.

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۶۰ـ گزینه «۳»** در Quick Sort، پیچیدگی زمانی در حالت متوسط $O(n\log n)$ و در بدترین حالت $O(n^2)$ است.

---

**۶۱ـ گزینه «۲»** در مرتب‌سازی ادغامی پیچیدگی حالت میانگین و بدترین حالت با هم برابر و مساوی $O(n\log n)$ می‌باشد و در این مرتب‌سازی نیاز به یک آرایهٔ کمکی است.

---

**۶۲ـ گزینه «۴»** در این الگوریتم هر جا که $M[K] > M[L]$ شود یک جابه‌جایی داریم، پس این الگوریتم مرتب‌سازی حبابی است و نمی‌تواند مرتب‌سازی انتخابی باشد و چون در هر تکرار، عنصر kام با $(k-1)$ام تا یکم مقایسه می‌شود، اگر بزرگ‌تر بود جابه‌جایی صورت می‌گیرد؛ پس مرتب‌سازی به صورت نزولی خواهد بود.

---

**۶۳ـ گزینه «۴»** ورودی زیر را برای مرتب‌سازی insertion در نظر بگیرید:

$$2,3,4,5,\ldots,n,1$$

در این حالت عدد 1 با تمام عناصر دیگر آرایه مقایسه خواهد شد.

ورودی زیر را برای مرتب‌سازی bubble در نظر بگیرید:

$$n,1,2,3,4,\ldots,n-1$$

در این حالت عدد n با تمام عناصر دیگر آرایه مقایسه خواهد شد.

ورودی زیر را برای مرتب‌سازی merge در نظر بگیرید:

$$n,n-1,n-2,\ldots,1$$

در این حالت عنصر میانه آرایه با تمام عناصر دیگر آرایه مقایسه خواهد شد.

اما در مرتب‌سازی heap حداکثر تعداد مقایسه‌های یک عنصر از مرتبه عمق heap یعنی، $\Omega(\log_2^n)$ می‌باشد.

---

**۶۴ـ گزینه «۱»** فرض کنید دو عنصر کوچک آرایه به ترتیب $\min 1$ و $\min 2$ و بزرگترین عنصر آرایه max باشد (که این عناصر در زمان $O(n)$ تعیین می‌گردند. حال در صورتی که $\min 1+\min 2<\max$ بنابراین. می‌توان نتیجه گرفت که برای تمام عناصر آرایه، رابطه موردنظر برقرار نیست. اما اگر $\min 1+\min 2>\max$ آنگاه با توجه به این که حاصل‌جمع هر دو عنصر دلخواه آرایه مانند x و y از $\min 1+\min 2$ بزرگتر است و هر عنصر آرایه مانند z از max کوچکتر است بنابراین رابطه موردنظر برقرار است.

---

**۶۵ـ گزینه «۴»** بهترین الگوریتم آن است که ابتدا آرایه B را به صورت صعودی مرتب کنیم که با بهترین الگوریتم دارای مرتبه اجرایی $O(m\log m)$ است، سپس تک‌تک عناصر آرایه A را به روش جستجوی دودویی در آرایه B جستجو کرده و اگر عنصر موردنظر در B نبود، عمل درج در C را انجام دهیم که مرتبه اجرایی این عمل $O(n\log m)$ خواهد بود، عناصر B را نیز در C درج می‌کنیم. بنابراین مرتبه اجرایی کل الگوریتم‌ها برابر $O((m+n)\log m)$ است.

---

**۶۶ـ گزینه «۲»** برای ادغام این k فایل باید ابتدا k بلوک که برابر r رکورد است از هر کدام از این فایل‌ها خوانده شود و در بلوک مربوط به فایل خروجی نوشته شود، پس هر فایل $\left\lceil \frac{n_i}{r} \right\rceil$ بلوک برای خواندن لازم دارد.

---

**۶۷ـ گزینه «۳»** الگوریتم Quick Sort در صورتی دارای زمان $\Omega(n^2)$ می‌باشد که در هر فراخوانی بر روی لیست n عنصری ورودی، بزرگترین یا کوچکترین عنصر لیست را به عنوان محوری انتخاب نماید. بنابراین احتمال این که در اولین فراخوانی بدترین عنصر محوری انتخاب گردد، $\frac{2}{n}$ می‌باشد. احتمال این‌که در دومین فراخوانی بدترین عنصر محوری انتخاب شود $\frac{2}{n-1}$ می‌باشد و ... . بنابراین احتمال این‌که Quick Sort دارای زمان $\Omega(n^2)$ باشد عبارتست از:

$$\frac{2}{n}\times\frac{2}{n-1}\times\frac{2}{n-2}\times\ldots 2=\frac{2^n}{n!}$$

---

www.modaresanesharif.ac.ir

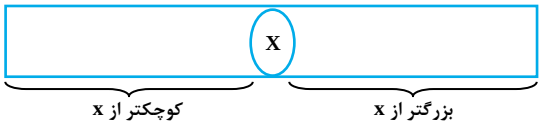
www.modaresanesharif.ac.ir



**۶۸ـ گزینه «۲»** اگر داده‌ها را به صورت یک heap فرض کنیم که این داده‌ها نه می‌توانند minheap باشند و نه maxheap ، بنابراین روش مرتب‌سازی heapsort نمی‌باشد و نیز بعد از دو مرحله با توجه به عنصر محوری در Quicksort ، Quicksort هم نمی‌تواند باشد و همچنین insertionsort هم نمی‌تواند باشد، زیرا باید داده‌ها از عنصر چهارم به بعد نزولی باشند که این طور نیست. ولی با اجرای دو مرحله mergesort بر روی داده‌های نزولی مورد نظر جواب حاصل می‌شود.

---

**۶۹ـ گزینه «۳»** برای انجام این کار می‌توان ابتدا عمل partition را حول x انجام داد (با زمان $O(n)$) سپس، آرایه به شکل زیر تبدیل می‌شود:




حال اگر دو عنصر مانند y و z وجود داشته باشند که $z+y=x$ باشد، آنگاه این دو عنصر حتماً قبل از x قرار دارند. بنابراین کافیست عناصر کوچکتر از x را مرتب کنیم (با زمان $O(n\log n)$) و سپس به ازای هر عنصر مانند y (که قبل از x قرار دارد) عنصر $x-y$ را در زیر آرایه سمت چپ جست و جو می‌کنیم (با زمان $O(n\log n)$).

---

**۷۰ـ گزینه «۲»** به مرتب‌سازی‌ای پایدار گویند که ترتیب اولیه رکوردهای هم کلید (منظور برابری کلید مرتب‌سازی در این رکوردها است) را حفظ می‌کند. مرتب‌سازی ادغامی و درجی پایدار هستند.

---

**۷۱ـ گزینه «۱»** در مرتب‌سازی درجی (Insertion sort) بهترین حالت زمانی است که آرایه از قبل مرتب باشد. در این صورت مرتبهٔ اجرایی الگوریتم $O(n)$ خواهد بود.

---

**۷۲ـ گزینه «۴»** در الگوریتم مرتب‌سازی درجی بر روی آرایه A با n عنصر حداکثر $\frac{n(n-1)}{2}=\frac{n^2-n}{2}$ وارونگی وجود دارد و بنا بر صورت مسئله، احتمال اینکه یک زوج اندیس دلخواه از A یک وارونگی باشد، برابر $\frac{1}{2}$ است. بنابراین میانگین تعداد وارونگی‌های یک آرایه A با عناصر متمایز $\frac{1}{2}\times\frac{n^2-n}{2}=\frac{n^2-n}{4}$ می‌باشد.

---

**۷۳ـ گزینه «۱»** در این الگوریتم مرتب‌سازی ادغامی (Merge sort)، تعداد مراحل ادغام کاهش می‌یابد، چون در مراحلی که فاصله میان عناصر دو لیست در حال ادغام، کمتر از b باشد هزینه عملیات صفر است. لذا، در کل به جای $\lg n$ مرحله ادغام، $\lg\frac{n}{b}$ مرحله ادغام خواهیم داشت. ولی کماکان هزینه هر مرحله ادغام $\theta(n)$ است. در نهایت مرتبه کل عملیات برابر $\theta(n\lg(\frac{n}{b}))$ است.

---

**۷۴ـ گزینه «۳»** در مرحله اول باید 1 به ابتدای لیست منتقل شود که این کار با جابجا کردن درایه‌های متوالی نیاز به 4 جابجایی دارد و عناصر به شکل روبرو درمی‌آید: $3,5,4,2,1$

سپس باید عدد 3 را منتقل کنیم که این عمل نیز نیاز به 2 جابجایی درایه‌های متوالی دارد و بعد از جابجایی نتیجه حاصل می‌شود و داریم: $5,4,3,2,1$

که در مجموع 6 جابجایی نیاز می‌باشد.

---

**۷۵ـ گزینه «۲»** رابطه بازگشتی داده شده به merge sort نزدیک‌تر می‌باشد.

---

**۷۶ـ گزینه «۱»** زیرا در بهترین حالت رابطه به صورت $T(n)=2T(\frac{n}{2})+\theta(1)$ و در بدترین $T(n)=T(n-1)+\theta(1)$ خواهد بود.

---

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۷۷ـ گزینه «۴»** برای به دست آوردن اشتراک دولیست نامرتب A و B به صورت بهینه می‌توان هر دو لیست را با بهترین الگوریتم مرتب کرد که مرتبهٔ اجرایی $O(n\lg n)$ خواهد بود، سپس با مقایسه عنصر به عنصر به صورت مرتب اشتراک دو لیست را به دست آورد که این عمل نیز در مرتبهٔ $O(n)$ انجام می‌شود. در هر دو حالت میانگین و بدترین حالت مرتبهٔ اجرایی $O(n\lg n)$ خواهد بود.

---

**۷۸ـ گزینه «۳»** به عنوان مثال نحوه اجرای الگوریتم داده شده بر روی آرایه زیر را در نظر بگیرید:

| 5 | 1 | 2 | 3 | 4 |
|---|---|---|---|---|

⇓

| 4 | 5 | 1 | 2 | 3 |
|---|---|---|---|---|

⇓

| 2 | 3 | 4 | 5 | 1 |
|---|---|---|---|---|

⇓

| 3 | 4 | 5 | 1 | 2 |
|---|---|---|---|---|

⇓

| 5 | 1 | 2 | 3 | 4 |
|---|---|---|---|---|

⇓

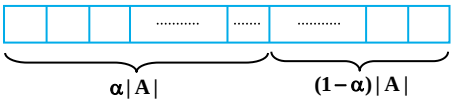
با توجه به این که همان آرایه اولیه ایجاد شده بنابراین یک حلقه بی‌نهایت ایجاد می‌گردد.

---

**۷۹ـ گزینه «۲»** می‌توان با استفاده از AVL مرتب‌سازی را در زمان $O(n\log k)$ انجام داد.

---

**۸۰ـ گزینه «۳»** فرض کنید در حالت کلی آرایه به نسبت $\alpha,(1-\alpha)$ تقسیم شود.




آنگاه می‌توان هزینه زمانی را به صورت زیر در نظر گرفت:

$$T(n)=T(\alpha n)+T((1-\alpha)n)+C_1n+C_2$$
$$=T(\alpha^2 n)+T(\alpha(1-\alpha)n)+C_1\alpha n+C_2+$$
$$T((1-\alpha)\alpha n)+T((1-\alpha)^2 n)+C_1(1-\alpha)n+C_2$$
$$=T(\alpha^2 n)+2T(\alpha(1-\alpha)n)+T((1-\alpha)^2 n)+C_1n+2C_2$$
$$=\cdots$$
$$=T(\alpha^k n)+\cdots$$

که داریم: $\{\alpha^k n=1 \Rightarrow n=(\frac{1}{\alpha})^k \Rightarrow k=Lg_{\frac{1}{\alpha}} n=\}$

بنابراین در کل مرتبه زمانی عبارتست از: $C_1Lg_{\frac{1}{\alpha}} n\times n+2Lg_{\frac{1}{\alpha}} n\times C_2$

مقدار $\alpha$ و حالات مختلف آن تأثیری بر کلاس پیچیدگی ندارد و فقط ضریب پیچیدگی کلاس را تغییر می‌دهد.

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



# فصل پانزدهم

## «مرتب‌سازی‌های غیرمقایسه‌ای»

### تست‌های طبقه‌بندی شده کنکوری فصل پانزدهم

**۱ـ الگوریتم ذیل به چه روشی Sort را انجام می‌دهد؟** (مهندسی کامپیوتر ـ آزاد۸۲)

```
for i = 1    to n – 1 do
{  k = 1
   for j = 2 to n – i + 1 do
   if a [j] > a [k] then  k = j
   swap (a [k] , a [n – i + 1]);
}
```

۱) counting sort

۲) radix sort

۳) selection sort

۴) shell sort

**۲ـ کدام گزینه نادرست است؟** (علوم کامپیوتر ـ سراسری ۸۵)

۱) الگوریتم‌های sort که پیچیدگی زمان آن خطی می‌باشند از حافظه کمکی استفاده می‌کنند.

۲) الگوریتم‌های sort که از مقایسه استفاده می‌کنند دارای حداقل پیچیدگی زمان $O(n\log n)$ می‌باشند.

۳) الگوریتم‌های sort که به صورت Divide-conquer حل می‌شوند دارای $O(n\log n)$ می‌باشند.

۴) الگوریتم‌های sort که از مقایسه استفاده نمی‌کنند می‌توانند دارای پیچیدگی زمان $O(n)$ باشند.

**۳ـ فرض کنید $S_1, S_2, \ldots, S_n$ آرایه‌هایی باشند که تعداد عنصر هر کدام از آن‌ها عدد صحیحی بین 1 تا k است (یعنی $S_i$ ها آرایه می‌باشند که $1 \le |S[i]| \le k$) و داریم: $\sum_{i=1}^{n} |S[i]| = k$ این بدین معنی است که مجموع تعداد عناصر تمام آن‌ها k است. بهترین الگوریتم مرتب‌سازی کل این آرایه‌ها دارای چه مرتبه زمانی است؟** (علوم کامپیوتر ـ سراسری ۸۵)

۱) $O(k)$ ۲) $O(n)$ ۳) $O(k\log n)$ ۴) $O(k\log k)$

**۴ـ پیچیدگی زمانی الگوریتم مرتب‌سازی پایه‌ای (Radix Sort) در بهترین حالت برابر است با:** (مهندسی کامپیوتر ـ آزاد ۸۶)

۱) $O(n^2)$ ۲) $O((\log n)^2)$ ۳) $O(n^2\log n)$ ۴) $O(n\log n)$

**۵ـ کدام یک از الگوریتم‌های مرتب‌سازی زیر در الگوریتم Radix sort برای بالا بردن سرعت استفاده می‌شود؟** (علوم کامپیوتر ـ سراسری ۸۷)

۱) Quick sort ۲) heap sort ۳) Insertion sort ۴) Merge sort

**۶ـ الگوریتم Radix Sort را روی n عدد در فاصله $[0, n^2-1]$ اجرا می‌کنیم. پایه استفاده شده در Radix Sort برابر n است. متوسط اجرای این الگوریتم چقدر است؟** (مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۸)

۱) $\theta(n^2)$ ۲) $\theta(n)$ ۳) $\theta(n\text{Logn})$ ۴) $\theta^2(n^2\text{Logn})$

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



## پاسخنامه تست‌های طبقه‌بندی شده کنکوری فصل پانزدهم

**۱ـ گزینه «۳»** در این الگوریتم ابتدا عنصر 1ام با عناصر 2 تا n مقایسه شده و اندیس بزرگ‌ترین مقدار در k قرار گرفته و در آخر a[k] با a[n] جابه‌جا می‌شود و با ادامه این روند آرایه مرتب خواهد شد.

**۲ـ گزینه «۳»** در گزینه ۳ ادعایی کلی شده که مثال نقض دارد چرا که الگوریتم مرتب‌سازی سریع در بدترین حالت، پیچیدگی $O(n^2)$ دارد و از الگوریتم‌های تقسیم و غلبه به شمار می‌آید.

**۳ـ گزینه «۴»** حالتی را در نظر بگیرید که کل عناصر در داخل $S_1$ قرار داشته باشند.

**۴ـ گزینه «۴»** مرتبه زمانی مرتب‌سازی مبنایی در بهترین حالت $\theta(n\log n)$ می‌باشد.

**۵ـ گزینه «۴»** باید از مرتب‌سازی متعادل استفاده کرد، یعنی مکان رکوردهای هم کلید نباید جابه‌جا شود.

**۶ـ گزینه «۲»** می‌دانیم در Radix sort، کل زمان اجرا $\theta(d(n+r))$ است، d برابر ارقام اعداد و r مبنا یا پایه اعداد می‌باشد و در این سؤال $r = n$ است، بنابراین زمان اجرایی برابر $\theta(d(n+n))$ خواهد بود و چون d خیلی از n کوچک‌تر است، پس زمان اجرا برابر $\theta(n)$ می‌شود.

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



# فصل شانزدهم

## «مسأله انتخاب (Selection)»

### تست‌های طبقه‌بندی شده کنکوری فصل شانزدهم

۱ـ الگوریتم زیر برای پیدا کردن عناصر ماکزیمم و مینیمم یک آرایه با n عنصر پیشنهاد شده است: (مهندسی کامپیوتر ـ سراسری ۷۵)

```
Procedure minmax (i,j:integer ;var min,max :real);
var k :integer ;
 min1 ,min2 ,max1 ,max2 : real;
begin
        if i + 1 = j then
            if A[i]>A[j] then
                    begin
                    min := A[j];
                    max :=A[i]
                    end
                else
                    begin
                        min :=A[i];
                        max :=A[j];
                    end
            else
                begin
                    k:=(i+j) div2;
                    minmax (i,k,min1,max1);
                    minmax (k + 1, j,min2 ,max2);
                    if min1 < min2 then
                        min:=min1
                    else
                        min := min2 ;
                    if max1 < max2 then
                        max:=max1
                    else
                        max:=max2
                end
end;
```

تعداد مقایسه‌های این الگوریتم (فقط مقایسه‌هایی که زیر آن‌ها خط کشیده شده است) برای $n = 8$ چقدر است؟

۱) 12 ۲) 13 ۳) 10 ۴) 11

۲ـ برای بدست آوردن بزرگترین و کوچکترین عنصر در یک لیست غیر مرتب بهترین الگوریتمی که می‌توان ارائه داد چند مقایسه خواهد داشت؟ (برای nهای زوج) (مهندسی کامپیوتر ـ سراسری ۸۱)

۱) $2(n-1)$ ۲) $\frac{3n}{2}-2$ ۳) $\frac{n-1}{2}+1$ ۴) $(n-1)+(n-2)$

۳ـ آرایه‌ی $S[1..N]$ از اعداد صحیح داده شده است. با داشتن $K$ می‌خواهیم $i$ و $j$ ای را (در صورت وجود) پیدا کنیم که $S[i]+S[j]=K$. برای این کار، الگوریتم زیر پیشنهاد شده است: (مهندسی کامپیوتر ـ سراسری ۸۲)

```
i ← 1; j ← n
while  i ≤ j  do
        if  S[i]+S[j]=K  then  return  i,j
        if  S[i]+S[j] < K  then   i ← i+1
        if  S[i]+S[j] > K  then   j ← j-1
end
return    not - possible
```

۱) همواره درست کار می‌کند.
۲) اگر S به صورت نزولی مرتب باشد درست کار می‌کند.
۳) اگر S به صورت صعودی مرتب باشد درست کار می‌کند.
۴) برای هر حالت ورودی ممکن است هیچگاه جواب پیدا نشود.

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۴ـ** الگوریتم زیر برای پیدا کردن اندیس‌های کوچکترین و بزرگترین عناصر در یک آرایه **N** تایی **A** داده شده است. متوسط تعداد مقایسه‌های دو عنصر از **A** چند تاست؟ (مهندسی کامپیوتر ـ سراسری ۸۲)

```
min ← 1; max ← 1
for i ← 2 to N do
    choose B = true or B = false with qual probabilities
    if B then if A[i] < A[min]
            Then min ← i
        else if A[i] > A[max]
            then max ← i
    else if A[i] > A[min]
        then max ← i
    else if A[i] < A[min]
        then min ← i
```

۱) $n$

۲) $2(n-1)$

۳) $\frac{3n}{2}$

۴) $\frac{3(n-1)}{2}$

**۵ـ** دومین کوچکترین عنصر بین **n** عنصر را با چند مقایسه می‌توان به دست آورد؟ (بهترین جواب ممکن را انتخاب کنید.) (مهندسی کامپیوتر ـ سراسری ۸۲)

۱) $n+\lceil \log_2 n \rceil - 2$　　۲) $n+\lceil \log_2 n \rceil - 1$　　۳) $n+\lfloor \log_2 n \rfloor - 2$　　۴) $n+\lfloor \log_2 n \rfloor - 1$

**۶ـ** در الگوریتم پیداکردن **k** امین عنصر از **n** عنصر، ابتدا همه عناصر را به دسته‌های 5 تائی (به جز احتمالاً یک دسته) تقسیم می‌کنیم، میانه هر دسته را به دست می‌آوریم و سپس میانهٔ میانه‌ها را به صورت بازگشتی پیدا می‌کنیم. این عنصر را به عنوان محور انتخاب می‌کنیم و عمل **Partition** را بر روی آرایه عناصر انجام می‌دهیم. پس از آن همین الگوریتم را به صورت بازگشتی (و برای یک **k** دیگر) بر روی یکی از بخش‌ها اجرا می‌کنیم تا عنصر مورد نظر پیدا شود. زمان اجرای این الگوریتم توسط کدامیک از رابطه‌های بازگشتی زیر بیان می‌شود؟ (مهندسی کامپیوتر ـ سراسری ۸۲)

۱) $T(n)=T\left(\left\lceil \frac{n}{5} \right\rceil\right)+T\left(\frac{3n}{10}-6\right)+O(n)$

۲) $T(n)=T\left(\left\lfloor \frac{n}{5} \right\rfloor\right)+T\left(\frac{3n}{10}-6\right)+O(n)$

۳) $T(n)=T\left(\left\lfloor \frac{n}{5} \right\rfloor\right)+T\left(\frac{7n}{10}+6\right)+O(n)$

۴) $T(n)=T\left(\left\lceil \frac{n}{5} \right\rceil\right)+T\left(\frac{7n}{10}+6\right)+O(n)$

**۷ـ** پیدا کردن **k** امین کوچک‌ترین عنصر بین **n** عنصر در بهترین حالت در چه مرتبه‌ای انجام می‌شود؟ (علوم کامپیوتر ـ سراسری ۸۳)

۱) $O(n)$　　۲) $O(n \lg n)$　　۳) $O(nk)$　　۴) $O(n^2)$

**۸ـ** الگوریتم پیدا کردن دو عنصر بیشینه و کمینه در یک آرایه با **N** عنصر به صورت زیر است: (علوم کامپیوتر ـ سراسری ۸۳)

```
MINMAX (A)
1 –  min ← 1
2 –  max ← 1
3 –  for i ← 2 to N
4 –      do if A[i] < A[min]
5 –          then min ← i
6 –          else if A[i] > A[max]
7 –                  then max ← i
```

حداکثر و حداقل تعداد مقایسه‌های دو عنصر از آرایه (سطر **4** و **6**) به ترتیب برابر است با:

۱) هر دو $n$　　۲) هر دو $n-1$　　۳) $2n$ و $n$　　۴) $2(n-1)$ و $n-1$

**۹ـ** دومین کوچک‌ترین عنصر بین **n** عنصر را با چند مقایسه می‌توان به دست آورد؟ (بهترین جواب ممکن کدام است؟) (علوم کامپیوتر ـ سراسری ۸۳)

۱) $n+\lceil \log n \rceil - 1$　　۲) $n+\lceil \log n \rceil - 2$　　۳) $n+\lfloor \log n \rfloor - 1$　　۴) $n+\lfloor \log n \rfloor - 2$

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



۱۰ـ در الگوریتم پیداکردن دو عنصر بیشینه و کمینه در یک آرایه با N عنصر (مطابق شکل زیر)، حداکثر و حداقل تعداد مقایسه‌های دو عنصر از آرایه به ترتیب برابر است با: (مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۴)

```
MINMAX(A)
1  min ← 1
2  max ← 1
3  for i ← 2  to  N
4     do if  A[i] < A[min]
5        then  min ← i
6        else  if A[i] > A[max]
7                 then max ← i
```

۱) $2n$ و $n$

۲) هر دو $n-1$

۳) هر دو $n$

۴) $2(n-1)$ و $(n-1)$

۱۱ـ دو آرایه $X[1..n]$ و $Y[1..n]$ مرتب شده‌اند. سریع‌ترین الگوریتم برای یافتن میانه $2n$ عضو آرایه‌های X و Y کدام است؟ (مهندسی کامپیوتر ـ سراسری ۸۵)

۱) روش استفاده از ادغام است که $\theta(n)$ زمان نیاز دارد.

۲) روش استفاده از ادغام است که $\theta(\log n)$ زمان نیاز دارد.

۳) یک الگوریتم تقسیم و حل است که نمونه‌های nتایی را به زیر نمونه‌های $\frac{n}{2}$ تایی تبدیل می‌کند و $\theta(\sqrt{n})$ زمان نیاز دارد.

۴) یک الگوریتم تقسیم و حل است که نمونه‌های n تایی را به زیرنمونه‌های $\frac{n}{2}$ تایی تبدیل می‌کند و $\theta(\log n)$ زمان نیاز دارد.

۱۲ـ کمترین مرتبه زمانی الگوریتم پیدا کردن i امین کوچکترین عنصر از میان n عنصر کدام است؟ (مهندسی فناوری اطلاعات (IT) ـ سراسری ۸۵)

۱) $O(n^2)$ ۲) $O(n)$ ۳) $O(n\lg n)$ ۴) هیچ‌کدام

۱۳ـ هرگاه بخواهیم در یک آرایه شامل n عنصر $(n = 2k)$ عناصر مینیمم و ماکزیمم آن‌را به‌دست آوریم. چه تعداد مقایسه باید انجام دهیم؟ (مهندسی کامپیوتر ـ آزاد ۸۵)

۱) $2k+3$ ۲) $3k+1$ ۳) $3k-2$ ۴) $3k-1$

۱۴ـ n عدد نامرتب و نامساوی داده شده‌اند. می‌خواهیم جمع کوچکترین $\sqrt{n}$ عددهای این اعداد را پیدا کنیم . یک الگوریتم کارا این مسئله را در چه زمانی می‌تواند حل کند؟ (مهندسی کامپیوتر ـ سراسری ۸۶)

۱) $O(\sqrt{n})$ ۲) $O(n)$ ۳) $O(n\lg n)$ ۴) $O(\sqrt{n}\lg n)$

۱۵ـ محاسبهٔ مجموع $\sqrt{n}$ کوچک‌ترین عناصر در یک آرایه نامرتب به طول n دارای پیچیدگی کدام است؟ (علوم کامپیوتر ـ سراسری ۸۶)

۱) $O(\sqrt{n})$ ۲) $O(n)$ ۳) $O(n\sqrt{n})$ ۴) $O(n\lg n)$

۱۶ـ اگر n عنصر نامرتب داشته باشیم می‌توان k عنصر بعد از Median را به صورت مرتب در پیچیدگی زمانی زیر چاپ کرد. (علوم کامپیوتر ـ سراسری ۸۶)

۱) $O(kn)$ ۲) $O(k\lg n)$ ۳) $O(n+k\lg k)$ ۴) $O(n+k\lg n)$

۱۷ـ تعداد مقایسه‌های لازم برای مشخص نمودن مینیمم و ماکزیمم عناصر ذخیره شده در یک آرایه یک بعدی شامل n عنصر را $T(n)$ فرض می‌کنیم. در این صورت رابطه $T(n)$ کدام است؟ (فرض کنید $T(1)=0$ و $T(2)=1$ باشد). (مهندسی کامپیوتر ـ آزاد ۸۶)

۱) $T(n)=\frac{3T(n-2)}{2}$ ۲) $T(n)=T(n-1)+T(n-2)+2$

۳) $T(n)=2T(n-2)+1$ ۴) $T(n)=T(n-2)+3$

۱۸ـ می‌خواهیم آرایه‌ی A به طول n را که در آن حداکثر k عدد مجزا از هم وجود دارد و داریم $k < \sqrt{n}$ را مرتب کنیم. (مهندسی کامپیوتر ـ سراسری ۸۷)

۱) این کار را می‌توان با درجه $(n \log k)$ انجام داد.

۲) این کار را می‌توان با درجه $(n+k \log k)$ انجام داد.

۳) این کار را می‌توان با درجه $(n)$ و مستقل از k انجام داد.

۴) این کار را نمی‌توان با درجه‌ی پیچیدگی‌ای کم‌تر از $n \log n$ انجام داد.

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



۱۹ـ آرایه‌ی n عضوی A از اعداد و یک عدد $m < n/2$ داده شده‌اند. می‌خواهیم کلیه درایه‌های آرایه $mins[1..n-m+1]$ را محاسبه کنیم. برای $1 \le i \le n-m+1$ درایه iام این آرایه به صورت زیر تعریف می‌شود: $mins[i] = min\{A[i], A[i+1], ..., A[i+m-1]\}$. کدام یک از گزینه‌های زیر زمان اجرای یک الگوریتم کارا برای حل این مسئله است؟ (بهترین جواب را انتخاب کنید.) (مهندسی کامپیوتر ـ سراسری ۸۷)

۱) $O(n \lg m)$ ۲) $O(m \lg n)$ ۳) $O(nm)$ ۴) $O((n+m)\lg(n+m))$

۲۰ـ اگر X[1..n] و Y[1..n] دو آرایه n تایی به صورت مرتب شده باشند کمترین زمان برای پیدا کردن Median (عنصر میانه) این 2n عدد برابر است با: (علوم کامپیوتر ـ سراسری ۸۷)

۱) $O(1)$ ۲) $O(n \lg n)$ ۳) $O(n)$ ۴) $O(\lg n)$

۲۱ـ اگر در داخل یک آرایه A[1...n] تمام اعداد صحیح 0..n به صورت باینری فقط یک بار دیده شوند و فقط یک عدد صحیح $x \in [0...n]$ در داخل آرایه نباشد، در این صورت پیچیدگی زمانی که می‌توان عدد از دست رفته x را می‌توان یافت کدام است؟ (علوم کامپیوتر ـ سراسری ۸۷)

۱) $O(1)$ ۲) $O(n)$ ۳) $O(\lg n)$ ۴) $O(n \lg n)$

۲۲ـ فرض کنیم آرایه A دارای n عنصر باشد، اگر عنصر x در آرایه وجود داشته باشد، به طور متوسط چه تعداد مقایسه بایستی انجام گیرد تا محل x در آرایه مشخص شود؟ (مهندسی کامپیوتر و مهندسی فناوری اطلاعات (IT) ـ آزاد ۸۷)

۱) $\frac{n}{2}$ ۲) $\frac{n+1}{2}$ ۳) $\frac{n-1}{2}$ ۴) $\frac{2n-1}{2}$

۲۳ـ آرایه A[1...3n] از اعداد داده شده است. می‌خواهیم با مقایسه‌ی اعداد آرایه، دو عدد x و y $(x < y)$ را به دست آوریم به طوری که n عنصر A مقداری کم‌تر از x، n عنصر A مقداری بین x و y و n عنصر بقیه مقداری بیشتر از y داشته باشند. یک الگوریتم کارا برای حل این مسئله به میزان M(n) حافظه‌ی اضافی (علاوه بر حافظه‌ی A) مصرف می‌کند و به زمان T(n) نیاز دارد. کدام یک، بهترین جواب برای این مسئله است؟ (مهندسی کامپیوتر ـ سراسری ۸۸)

۱) $T(n) = O(n)$ و $M(n) = O(n)$
۲) $T(n) = O(n)$ و $M(n) = O(1)$
۳) $T(n) = O(n^2)$ و $M(n) = O(1)$
۴) $T(n) = O(n \lg n)$ و $M(n) = O(n)$

۲۴ـ مینیمم و ماکزیمم اعداد ذخیره شده در آرایه A[1..n] باچند مقایسه بین این اعداد به‌دست می‌آید؟ (فرض کنید $n = 2K+1$ و K عدد طبیعی است). (مهندسی کامپیوتر ـ آزاد ۸۸)

۱) $3k+1$ ۲) $\frac{3k}{2}-1$ ۳) $3k$ ۴) $\frac{3k}{2}-\frac{3}{2}$

۲۵ـ فرض کنید که $n = 2^k$. می‌خواهیم عنصر بیشینه (بزرگترین) را در یک ماتریس به اندازه‌ی $n \times n$ بیابیم. برای این کار ماتریس را به چهار قسمت مساوی و هر کدام را به اندازه‌ی $\frac{n}{2} \times \frac{n}{2}$ تقسیم می‌کنیم، بیشینه‌ی هر کدام را به صورت بازگشتی به دست می‌آوریم و بین آن‌ها جواب نهایی را پیدا می‌کنیم. تعداد دقیق مقایسه‌های عناصر با هم در این الگوریتم چندتاست؟ (مهندسی کامپیوتر ـ سراسری ۸۹)

۱) $n^2-1$ ۲) $2n-1$ ۳) $3(\log_2 n)$ ۴) $3(\log_2 n - 1)$

۲۶ـ با چند مقایسه می‌توان مینیمم و ماکزیمم اعداد ذخیره شده در یک آرایه یک بعدی با n عنصر را به‌دست آورد؟ (فرض کنید n زوج باشد). (مهندسی کامپیوتر ـ آزاد ۸۹)

۱) $\frac{3n}{2}-2$ ۲) $\frac{3n}{2}$ ۳) $\frac{3n}{2}+1$ ۴) $\frac{3n}{2}-1$

۲۷ـ پیداکردن kامین کوچکترین عنصر بین n عنصر در بهترین حالت در چه مرتبه‌ای انجام می‌شود؟ (مهندسی نرم‌افزار ـ آزاد ۹۱)

۱) $o(n)$ ۲) $o(n \log n)$ ۳) $o(nk)$ ۴) $o(n^2)$

www.modaresanesharif.ac.ir

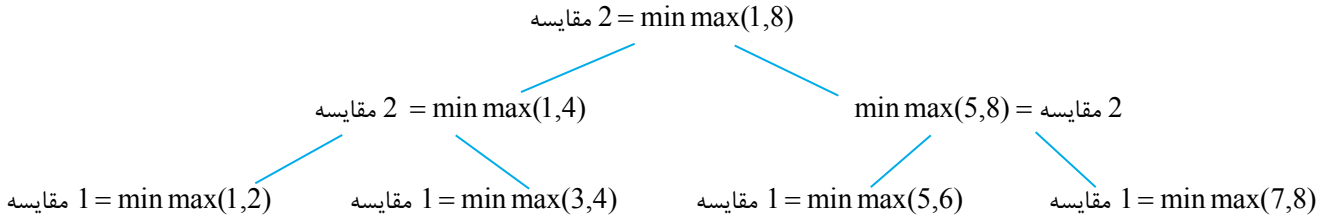
www.modaresanesharif.ac.ir



## پاسخنامه تست‌های طبقه‌بندی شده کنکوری فصل شانزدهم

**۱ـ گزینه «۳»** در روال minmax، هر گاه $i+1=j$ باشد شرط $A[i]>A[j]$ بررسی می‌شود و در غیر این صورت دو شرط $\min 1<\min 2$ و $\max 1<\max 2$ بررسی خواهد شد، بنابراین در فراخوانی روال با $i$ و $j$ متوالی یک شرط از شرط‌هایی که زیر آن‌ها خط کشیده شده بررسی می‌شود و برای سایر $i$ و $j$ها دو شرط بررسی می‌شود.

و با رسم درخت فراخوانی‌های روال minmax برای $n=8$ داریم:




**۲ـ گزینه «۲»** بهترین الگوریتم برای پیدا کردن کوچکترین و بزرگترین عنصر در یک آرایه دارای زمان به صورت زیر خواهد بود:

$$n \text{ فرد}: \frac{3n}{2}-\frac{3}{2} \quad ، \quad n \text{ زوج}: \frac{3n}{2}-2$$

**۳ـ گزینه «۳»** اگر لیست داده شده صعودی باشد آنگاه در صورت وجود $i$ و $j$ این الگوریتم جواب را پیدا خواهد کرد.

به عنوان مثال آرایه نزولی زیر را در نظر بگیرید:

| 5 | 4 | 3 | 2 | 1 |
|---|---|---|---|---|

با فرض این که $k=7$ باشد آنگاه نحوه اجرای الگوریتم به صورت زیر خواهد بود:

| 5 | 4 | 3 | 2 | 1 |
|---|---|---|---|---|

↑ ↑ $S[i]+S[j]<K$

$i=1$ $j=5$

$\Rightarrow$

| 5 | 4 | 3 | 2 | 1 |
|---|---|---|---|---|

↑ ↑ $S[i]+S[j]<K$

$i=2$ $j=5$

$\Rightarrow$

| 5 | 4 | 3 | 2 | 1 |
|---|---|---|---|---|

↑ ↑ $S[i]+S[j]<K$

$i=4$ $j=5$

$\Rightarrow$

| 5 | 4 | 3 | 2 | 1 |
|---|---|---|---|---|

↑ ↑ return not − possible

$i=5$ $j=6$

$\Rightarrow$

| 5 | 4 | 3 | 2 | 1 |
|---|---|---|---|---|

↑ ↑ $S[i]+S[j]<K$

$i=3$ $j=5$

$\Rightarrow$

| 5 | 4 | 3 | 2 | 1 |
|---|---|---|---|---|

↑ ↑ $S[i]+S[j]<K$

$i=5$ $j=5$

همان‌گونه که مشاهده می‌گردد با توجه به این که حاصل‌جمع اعداد 2 و 5 و همچنین حاصل‌جمع اعداد 4 و 3 برابر $k$ می‌باشد الگوریتم جواب درستی را محاسبه نکرده است. اما اگر لیست مرتب باشد الگوریتم به درستی عمل خواهد نمود:

| 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|

↑ ↑ $S[i]+S[j]<k$

$i=1$ $j=5$

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



$\Rightarrow$ | 1 | 2 | 3 | 4 | 5 |

$i=2$ $j=5$ $\quad S[i]+S[j]=5 \Rightarrow \text{return } 2,5$

---

**۴ـ گزینه «۴»** برای متغیر تصادفی B می‌توان دو حالت زیر را در نظر گرفت:

حالت اول: با احتمال $\frac{1}{2}$ متغیر B برابر True می‌باشد در این حالت اگر $A[i] < A[min]$ باشد آنگاه تنها یک مقایسه انجام خواهد شد و اگر شرط $A[i] < A[min]$ برقرار نباشد یک مقایسه دیگر نیز صورت می‌پذیرد بنابراین با توجه به این که با احتمال $\frac{1}{2}$ شرط برقرار است و با احتمال $\frac{1}{2}$ نیز شرط برقرار نمی‌باشد بنابراین میانگین تعداد مقایسه‌ها در این حالت عبارتست از:

$$\frac{1}{2}\times 1+\frac{1}{2}\times 2=\frac{3}{2}$$

حالت دوم: با احتمال $\frac{1}{2}$ مقدار متغیر B برابر False می‌باشد بنابراین با همان استدلال قبلی میانگین تعداد مقایسه‌ها عبارتست از:

$$\frac{1}{2}\times 1+\frac{1}{2}\times 2=\frac{3}{2}$$

درکل میانگین تعداد مقایسه‌ها برای $n-1$ عنصر آرایه عبارتست از:

$$(\frac{1}{2}\times\frac{3}{2}+\frac{1}{2}\times\frac{3}{2})\times(n-1)=\frac{3(n-1)}{2}$$

حالتی که $B=True$ باشد — حالتی که $B=False$ باشد — تمام عناصر آرایه به جز اولین عنصر آرایه

---

**۵ـ گزینه «۱»** دومین کوچکترین عنصر در یک آرایه n عنصری را می‌توان با $n+\lceil \log_2^n \rceil -2$ مقایسه تعیین نمود. (به عنوان مثال در یک آرایه دو عنصری تنها یک مقایسه و در یک آرایه سه عنصری سه مقایسه نیاز می‌باشد)

---

**۶ـ گزینه «۴»** اگر از الگوریتم تقسیم و غلبه برای پیداکردن k امین عنصر از n عنصر استفاده کنیم و ابتدا همه عناصر را به دسته‌های 5 تایی تقسیم کنیم، آنگاه رابطه بازگشتی مربوط به زمان اجرای این الگوریتم به صورت مقابل خواهد بود:

$$T(n)=T\left(\left\lceil \frac{n}{5} \right\rceil\right)+T\left(\frac{7n}{10}+6\right)+O(n)$$

---

**۷ـ گزینه «۱»** با استفاده از ایده محورگیری با پیچیدگی خطی می‌توان به این عنصر رسید. (به متن درس مراجعه کنید).

---

**۸ـ گزینه «۴»** حداکثر زمانی است که هر دو دستور 4 و 6 اجرا شوند و در حالت حداقل، ما حتماً اجرای دستور 4 را خواهیم داشت.

---

**۹ـ گزینه «۲»** برای دومین کوچک‌ترین عنصر در بین n عنصر حداقل تعداد مقایسه‌ها برابر است با: $n+\lceil \log n \rceil -2$

---

**۱۰ـ گزینه «۴»** کمترین تعداد مقایسه، زمانی رخ می‌دهد که شرط موجود در خط 4 همواره برقرار باشد که در این حالت، مقایسه مربوط به خط شماره 6 انجام نمی‌شود و در نهایت $n-1$ مقایسه صورت خواهد پذیرفت (این حالت در یک آرایه مرتب نزولی رخ می‌دهد). بیشترین تعداد مقایسه، زمانی رخ خواهد داد که شرط موجود در خط 4 هیچ‌گاه برقرار نباشد. این مورد در حالتی رخ می‌دهد که اولین عنصر آرایه کوچک‌ترین عنصر آن نیز باشد که $2\times(n-1)$ مقایسه انجام خواهد شد.

به عنوان مثال اگر آرایه ورودی به صورت روبرو باشد:

| 5 | 4 | 3 | 2 | 1 |

در این حالت همواره شرط موجود در خط چهارم برقرار است و بنابراین شرط موجود در خط ششم اجرا نخواهد شد و با انجام 4 مقایسه حلقه for پایان می‌پذیرد. اما اگر آرایه به صورت روبرو باشد:

| 1 | 4 | 3 | 4 | 2 |

در این حالت شرط موجود در خط چهارم هیچگاه برقرار نیست و بنابراین پس از بررسی شرط خط چهارم، شرط خط ششم کنترل می‌گردد و در نتیجه 8 مقایسه انجام خواهد شد.

---

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۱۱ـ گزینه «۴»** برای حل این مسئله در هر یک از لیست‌ها میانه را تعیین می‌کنیم سپس میانه‌ها را مقایسه می‌نماییم و نیمه بالایی لیستی که میانه آن بزرگتر و نیمه پایینی لیستی که میانه آن کوچکتر است را حذف می‌کنیم و به همین ترتیب ادامه می‌دهیم با توجه به این که هر بار نصف یک لیست حذف می‌شود زمان $\theta(\log n)$ خواهد بود و البته مشخص است که روش استفاده شده از نوع تقسیم و غلبه است.

به عنوان مثال دو آرایه $X$ و $Y$ را به صورت زیر در نظر بگیرید:

| | | | | | | |
|---|---|---|---|---|---|---|
| X | 1 | 3 | 5 | 7 | 9 | میانه = 5 |
| Y | 2 | 4 | 6 | 8 | 10 | میانه = 6 |

حال با توجه به این که $5 < 6$ بنابراین نیمه کوچکتر لیست $X$ و نیمه بزرگتر لیست $Y$ را حذف می‌نماییم (نیمه کوچکتر لیست $X$ و نیمه بزرگتر لیست $Y$ به ترتیب از میانه کلی کوچکتر و بزرگتر خواهند بود و نیازی به بررسی آن‌ها نمی‌باشد) بنابراین لیست‌های $X$ و $Y$ به صورت زیر تبدیل می‌شوند:

| | | | | |
|---|---|---|---|---|
| X | 5 | 7 | 9 | میانه = 7 |
| Y | 2 | 4 | 6 | میانه = 4 |

با توجه به این‌که $4 < 7$ آرایه‌های $X$ و $Y$ به صورت زیر تغییر می‌نمایند:

| | | | |
|---|---|---|---|
| X | 5 | 7 | میانه = 5 |
| Y | 4 | 6 | میانه = 4 |

در نهایت در مرحله بعدی داریم:

| | | |
|---|---|---|
| X | 5 | |
| Y | 4 | 6 |

که میانه 5 خواهد بود.

بنابراین الگوریتم زیر را می‌توان برای این مسئله در نظر گرفت:

```
Median2array(X, Y, n)
{
  m1 = x[n/2]
  m2 = y[n/2]
  if  n ≤ 2
       return  median  of  X, Y
  if(m1 < m2)
       Median2arrary (X[n/2 ... n], Y[1 ... n/2], n/2)
  else if(m1 > m2)
     Median2array(x[1 ⋯ n/2], y[n/2 ⋯ n], n/2)
else
       return m1
}
```

همان‌گونه که مشاهده می‌گردد تابع پیچیدگی زمانی این الگوریتم به صورت مقابل می‌باشد: $T(n) = T(\frac{n}{2}) + \theta(1) \Rightarrow T(n) = \theta(\log_2^n)$

---

**۱۲ـ گزینه «۲»** کمترین مرتبهٔ یافتن $i$ امین کوچکترین عنصر در یک لیست $n$ عنصری برابر $\theta(n)$ می‌باشد. اگر لیست از قبل مرتب باشد، مرتبه یافتن $i$ امین عنصر از $\theta(1)$ می‌باشد.

---

**۱۳ـ گزینه «۳»** تعداد مقایسه‌ها در یک آرایه $n$ عنصری برای یافتن همزمان min و max عبارتست از:

اگر $n$ زوج باشد: $\frac{3n}{2} - 2$

اگر $n$ فرد باشد: $\frac{3n}{2} - \frac{3}{2}$

بنابراین با توجه به این‌که $n$ زوج است تعداد مقایسه‌ها عبارتست از: $\frac{3 \times 2k}{2} - 2 = 3k - 2$

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۱۴ـ گزینه «۲»** یافتن $\sqrt{n}$ عدد کوچک یک لیست n عنصری در زمان $O(n)$ امکان‌پذیر است.

---

**۱۵ـ گزینه «۲»** کافیست ابتدا k امین کوچکترین را در زمان $O(n)$ یافت و سپس عمل افراز را حول این عنصر انجام داد تا عناصر کوچکتر به قبل از آن منتقل شود و سپس عناصر موردنظر در زمان $O(\sqrt{n})$ با یکدیگر جمع می‌شوند.

---

**۱۶ـ گزینه «۳»** کافی است در محورگیری $\frac{n}{2}$ و $\frac{n}{2}+k$ در جای خود قرار گیرند (زمان هر دوی اینها $O(n)$ است) بعد از این می‌توانیم این k عنصر را در زمان $O(klogk)$ مرتب کنیم.

---

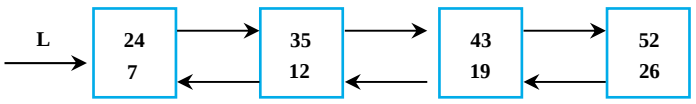
**۱۷ـ هیچ‌کدام از گزینه‌ها صحیح نیست.** تعداد مقایسه‌ها به‌صورت روبه‌رو می‌باشد: $T(n)=\frac{3(n-1)}{2}+1$

---

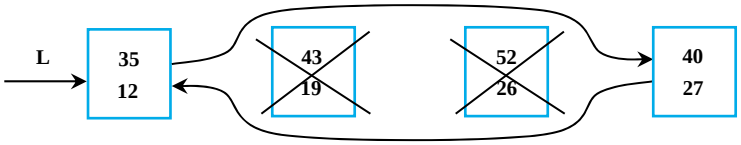
**۱۸ـ گزینه «۱»** اگر $k=1$ باشد آنگاه لیست مرتب است و نیاز به هیچ عملیاتی نیست که این موضوع فقط در گزینه اول صادق است.

---

**۱۹ـ هیچ‌کدام از گزینه‌ها صحیح نیست.** فرض کنید از یک لیست پیوندی دوسویه استفاده کنیم. در ابتدا اولین عنصر A را به همراه شماره اندیس ۱ به عنوان اولین عنصر از لیست در نظر می‌گیریم. سپس $m-1$ عنصر بعدی A را به شکل زیر به سمت راست لیست اضافه می‌کنیم: «ابتدا عنصر جدید را به همراه شماره اندیس آن در آرایه A در انتهای لیست قرار می‌دهیم، سپس تا زمانی که عنصر سمت چپ آن از آن بزرگتر باشد، عنصر سمت چپ آن را حذف می‌کنیم.» پس از اضافه نمودن این عناصر به لیست پیوندی به روش فوق، اولین عنصر از لیست همان $\min s[1]$ خواهد بود. از این پس به ازای هر عنصر $m+1\le k\le n$ به لیست پیوندی، ابتدا اندیس اولین عنصر را بررسی می‌کنیم و اگر کوچکتر از $k-m+1$ بود آن را حذف می‌کنیم. سپس عنصر جدید را مشابه با روش قبل به سمت راست لیست اضافه می‌کنیم. عنصر $\min s[k-m+1]$ همان اولین عنصر از لیست پیوندی پس از اضافه نمودن عنصر $A[k]$ است. برای مثال فرض کنید لیست پیوندی به‌صورت زیر باشد: (عدد بالایی مقدار و عدد پایینی اندیس است)



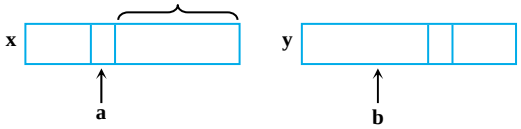

فرض کنید $m=20$ بوده و بخواهیم 27امین عنصر A با مقدار 40 را به لیست اضافه کنیم. ابتدا عنصر با مقدار 24 و اندیس 7 حذف می‌شود، زیرا مقدار 7 کمتر از $27-20+1=8$ است. سپس عنصر با مقدار 40 و اندیس 27 به سمت راست لیست اضافه می‌شود. با توجه به اینکه مقدار دو عنصر 52 و 43 از آن بیشتر هستند این دو عنصر از لیست حذف می‌شوند و لیست به شکل زیر خواهد شد که اولین عنصر از آن مقدار 35 دارد.




در این روال، لیست پیوندی همواره صعودی است و هر عنصر دقیقاً یک مرتبه به لیست اضافه شده و حداکثر یک مرتبه حذف می‌شود. این روال برای n عنصر تکرار می‌شود. در نتیجه مرتبه محاسبه مقادیر خواسته شده $O(n)$ است.

---

**۲۰ـ گزینه «۴»** با یک ایده‌ی تقسیم و غلبه می‌توان در $O(Log)$ میانه را یافت. میانه‌ی هر یک را در نظر بگیرید و آن دو را با هم مقایسه کنید.




$$a<b: Median(x[\frac{n}{2},...,n],\ y[1,...,\frac{n}{2}])$$

$$a>b: Median(x[1,...,\frac{n}{2}],\ y[\frac{n}{2},...,n])$$

$$a=b: Median=a$$

---

www.modaresanesharif.ac.ir

www.modaresanesharif.ac.ir



**۲۱ـ گزینه «۲»** گزینه ۱ هرگز نمی‌تواند تحقق پیدا کند. برای گزینه ۳ نیز هیچ طرحی نداریم اما اگر با روش زیر عمل کنیم در زمان خطی آن را خواهیم یافت.

```
for i = 1 to n
    A[i] = 0
for i = 1 to n
    if (A[i] ≠ i & A[i] ≠ 0) then
        exchange (A[i], A[A[i]])
for  i = 1   to n
      if A[i] = 0 then
        return  i
else
return 0
```

---

**۲۲ـ گزینه «۲»** در محاسبهٔ متوسط تعداد مقایسه‌ها برای جستجوی موفق $x$ در آرایه $A$ به روش جستجوی خطی، باید برای هر حالتی تعداد مقایسه‌ها را به دست آورده و مجموع آن‌ها را بر کل حالات ممکن تقسیم کنیم، داریم:

$$A(n)=\frac{1+2+\cdots+n}{n}=\frac{\frac{n(n+1)}{2}}{n}=\frac{n+1}{2}$$

---

**۲۳ـ گزینه «۲»** با استفاده از الگوریتم انتخاب میانه می‌توان این مسئله را حل کرد و نیازی به حافظه کمکی نیست. کافیست $n$ امین کوچکترین عنصر و $n$ امین بزرگترین عنصر هر یک در زمان $O(n)$ تعیین شوند که به ترتیب $x$ و $y$ را مشخص می‌کنند.

---

**۲۴ـ گزینه «۳»** با توجه به این‌که تعداد عناصر آرایه فرد می‌باشد بنابراین تعداد مقایسه‌ها عبارتست از:

$$\frac{3n}{2}-\frac{3}{2}$$

بنابراین تعداد مقایسه‌ها برابر است با:

$$\frac{3(2k+1)}{2}-\frac{3}{2}=3k$$

---

**۲۵ـ گزینه «۱»** به ازای $n=2$ ماتریس به صورت مقابل است:

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix}$$

در کل به سه مقایسه نیاز است. به ازای $n=4$ داریم:

$$\left[\begin{array}{cc:cc} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ \hdashline a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{array}\right]$$

در کل تعداد مقایسه‌ها برابر است با $4\times 3+3=15$. رابطه بازگشتی مرتبه زمانی به صورت مقابل می‌باشد: $T(n)=4T(\frac{n}{2})+3\ ,\ T(2)=3$

تنها گزینه اول در این رابطه صادق است.

---

**۲۶ـ گزینه «۱»** در حالت کلی تعداد مقایسه‌ها برای یافتن بزرگ‌ترین و کوچک‌ترین کلید به‌صورت همزمان عبارتست از:

اگر $n$ زوج باشد: $\frac{3n}{2}-2$

اگر $n$ فرد باشد: $\frac{3n}{2}-\frac{3}{2}$

---

**۲۷ـ گزینه «۱»** با استفاده از الگوریتم تقسیم و غلبه (بیان شده در متن درس) می‌توان در زمان خطی عنصر موردنظر را یافت.

www.modaresanesharif.ac.ir