سؤالات فرد نیمسال اول ۹۹-۹۸

① مقدار $T(n)$ در کد زیر چه می‌باشد؟

```
x = 0;
for(i=0, i<n; i++)
   for(j=0; j<n, j++)
      x++;
```

۱) $T(n) = c(2n^2+2n+2)$

۲) $T(n) = c(2n^3+2n+2)$

۳) $T(n) = c(2n+2)$

۴) $T(n) = c(2n^4)$

حل) گزینه ۱

```
1) x = 0
2) for(i=0 , i<n; i++)
3) for(j=0 , j<n ; j++)
4) x++;
```

تابع زمانی قطعه کد بالا به صورت زیر محاسبه می‌شود.

سطر	هزینه	تعداد
۱	c_1	1
۲	c_2	$n+1$
۳	c_3	$n(n+1)$
۴	c_4	$n \times n$

بنابراین $T(n)$ برابر است با:

$$T(n) = c_1 + c_2(n+1) + c_3 n(n+1) + c_4 n^2$$

c را بیشترین مقدار c_1، c_2، c_3، c_4 در نظر می‌گیریم

بنابراین خواهیم داشت:

$$T(n) = c(2n^2+2n+2)$$

③ اگر $T(n) = \frac{1}{2}n^2 - 3n$ باشد، مقدار $\theta(g(n))$ چیست؟

۱) $T(n) \in \theta(n^k)$

۲) $T(n) \in \theta(nm^k)$

۳) $T(n) \in \theta(n^2)$

۴) $T(n) \in \theta(nm^2)$

حل) گزینه ۳

نشان می‌دهیم که $T(n) \in \theta(n^2)$

طبق تعریف: $T(n) \in \theta(n^2) \iff$

$\exists\, c_1, c_2, n_0$ به طوری که $\forall n \geq n_0$

$$\boxed{c_1 n^2 \leq \frac{1}{2}n^2 - 3n \leq c_2 n^2}$$ رابطه (۱)

حال رابطه‌ی (۱) را به n^2 تقسیم می‌کنیم:

$$c_1 \leqslant \frac{1}{2} - \frac{3}{n} \leqslant c_2$$

با توجه به عبارت بالا، قسمت راست، به ازای $c_2 \geqslant \frac{1}{2}$ و $n \geqslant 1$ برقرار است. به همین ترتیب برای قسمت چپ عبارت بالا به ازای $n \geqslant 7$، $c_1 \leqslant \frac{1}{4}$ حاصل می‌شود.

بنابراین به ازای $c_1 = \frac{1}{4}$ ، $c_2 = \frac{1}{2}$ ، $n_0 \geqslant 7$، عبارت $T(n) \in \Theta(n^2)$ خواهد بود.

با توجه به مثال بالا در حالت کلی اگر

$$T(n) = a_k n^k + a_{k-1} n^{k-1} + a_1 n + a_0$$

باشد، آنگاه: $T(n) \in \Theta(n^k)$

۵) تابع بازگشتی محاسبه‌ی $n!$ به صورت زیر می‌باشد. $T(n)$ آن به چه صورت می‌باشد؟

```
Int fact (int n)
{
    if (n == 0)
      return (1);
    else
      return (n * fact (n-1);
```

۱)
$$T(n) = \begin{cases} d & \text{if } n > 0 \\ T(n-1)+c & \text{if } n = 0 \end{cases}$$

۲)
$$T(n) = \begin{cases} d & \text{if } n < 0 \\ T(n+1)+c & \text{if } n > 0 \end{cases}$$

۳)
$$T(n) = \begin{cases} d & \text{if } n = 0 \\ T(n+1)+c & \text{if } n > 0 \end{cases}$$

۴)
$$T(n) = \begin{cases} d & \text{if } n = 0 \\ T(n-1)+c & \text{if } n > 0 \end{cases}$$

حل: گزینه ۴

$T(n)$ را زمان اجرای تابع fact (n) درنظر می‌گیریم.
زمان اجرای دستور if برابر با $O(1)$ می‌باشد و
زمان اجرای else دستور if برابر $O(1) + T(n-1)$
که در آن $O(1)$ زمان مربوط به عمل ضرب و فراخوانی تابع می‌باشد. بنابراین:

$$T(n) = \begin{cases} O(1) & \text{if } n = 0 \\ O(1) + T(n-1) & \text{if } n > 0 \end{cases}$$

$T(n)$ را به صورت زیر می‌توان نوشت:

$$T(n) = \begin{cases} d & \text{if } n = 0 \\ T(n-1)+c & \text{if } n > 0 \end{cases}$$

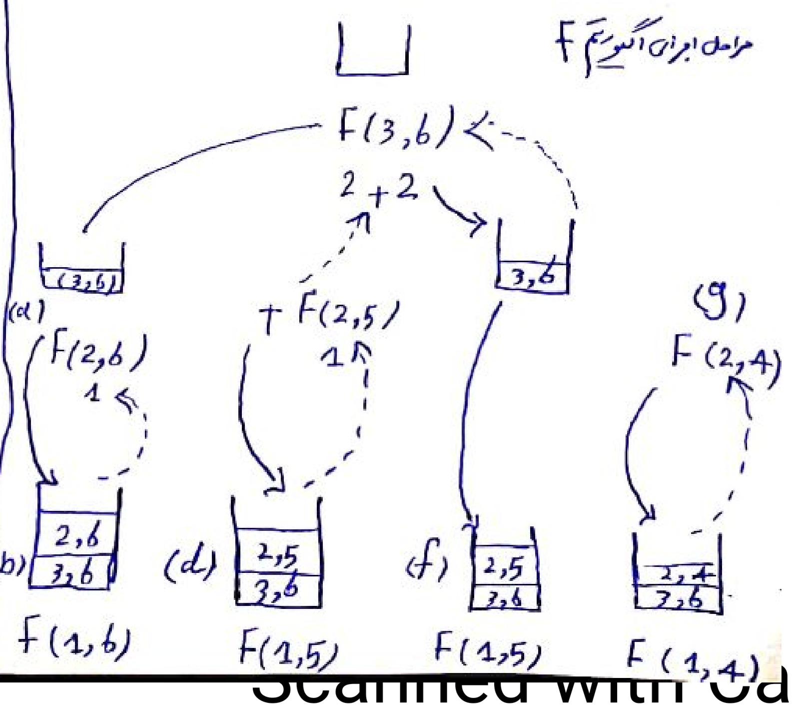
⑦ خروجی تابع زیر به ازای $F(3,6)$ چیست؟

```
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) ۲ ۲) ۳ ۳) ۴ ۴) ۵

حل) گزینه ۳

مراحل اجرای الگوریتم بالا را به ازای مقادیر داده شده در شکل زیر نمایش می دهیم.



شکل، مراحل محاسبه مقدار تابع بازگشتی را در دو مرحله فراخوانی و بازگشت نشان می دهد. و در نهایت مقدار ۴ را به عنوان خروجی نمایش می دهد.

---

⑨ رابطه‌ی بازگشتی زیر را در نظر بگیرید. مقدار $T(n)$ کدام است؟

$$T(n) = 3T(n-1) + 4T(n-2)$$

$$T(0) = 0 \;,\; T(1) = 1$$

۱) $T(n) \in O(4^n)$

۲) $T(n) \in O(m^n)$

۳) $T(n) \in O(n^m)$

۴) $T(n) \in O(n^n)$

حل) گزینه ۱

برای حل رابطه بالا، $T(n) = X^n$ قرار می دهیم. بنابراین خواهیم داشت:

$$X^n = 3X^{n-1} + 4X^{n-2} \Longrightarrow$$

$$X^2 - 3X + 4 = 0$$

جواب های معادله مشخصه عبارت اند از:

$$X_1 = -1 \;,\; X_2 = 4$$

بنابراین:

$$T(n) = C_1(-1)^n + C_2 4^n$$

خواهد بود. حال با توجه به شرایط مرزی $T(0)$ و $T(1)$ در معادله، عبارت زیر حاصل می شود:

$$\begin{cases} c_1 + c_2 = 0 \\ -c_1 + 4c_2 = 1 \end{cases}$$

که از آن $c_1 = -\frac{1}{5}$ و $c_2 = \frac{1}{5}$ بدست می آید. بنابراین:

$$T(n) = \frac{1}{5}\left(4^n - (-1)^n\right)$$

و از آنجا $T(n) \in O(4^n)$

۱۱) در کدام روش مرتب سازی از یک عنصر به عنوان عنصر محور استفاده می شود؟

۱) مرتب سازی سریع (quick sort)

۲) مرتب سازی ادغامی

۳) مرتب سازی درجی

۴) مرتب سازی تقسیم و حل

حل) گزینه ۱

در روش مرتب سازی سریع یک عنصر به عنوان عنصر محور انتخاب می شود.

۱۳) الگوریتم پیدا کردن ماکزیمم و مینیمم به چه مقدار زمان برای مقایسه نیاز دارد؟ (با فرض اینکه عمل اصلی در این الگوریتم مقایسه باشد.)

۱) $\frac{2n}{3}-2$ ۲) $\frac{3n}{3}-2$

۳) $\frac{3n}{2}-2$ ۴) $\frac{3n}{2}-1$

حل) گزینه ۴

در این مسئله هدف یافتن کوچکترین عنصر لیست و بزرگترین عنصر لیست می باشد. الگوریتم ساده برای این کار این است که اولین عنصر را بعنوان عنصر max و min در نظر بگیریم، سپس با $n-1$ عنصر دیگر مقایسه کنیم. این الگوریتم ساده، به وضوح زمان متوسطی در حدود $\frac{3n}{2}-1$ مقایسه نیاز دارد. (با فرض اینکه عمل اصلی در این الگوریتم مقایسه است.)

۱۵) در الگوریتم حریصانه کدام جزء تشکیل دهنده آن برای بررسی اینکه مشخص کند درنهایت جواب حاصل شده است یا خیر به کار می رود؟

۱) SELECT

۲) FEASIBLE

۳) SOLUTION

۴) یک تابع هدف

حل) گزینه ۳

روالی به نام Solution: برای بررسی اینکه مشخص کند جواب به حاصل شده یا خیر.

---

۱۷) کدام الگوریتم برای یافتن کوتاه‌ترین مسیرها از مبدأ واحد به مقصدهای متفاوت به‌کار می‌رود؟

۱) الگوریتم دیکسترا

۲) الگوریتم کروسکال

۳) الگوریتم پریم

۴) الگوریتم درخت پوشای مینیمم

حل) گزینه ۱

الگوریتم دیکسترا برای یافتن یکبه کوتاه‌ترین مسیرها از مبدأ واحد به مقصدهای متفاوت به‌کار می‌رود.

---

۱۹) ضرب چهار ماتریس زیر را در نظر بگیرید.

$A_{20\times 2} \times B_{2\times 30} \times C_{30\times 12} \times D_{12\times 8}$

مقدار (AB)(CD) کدام است؟

۱) ۳۶۸۰

۲) ۸۸۸۰

۳) ۱۲۳۲

۴) ۳۱۲۰

حل) گزینه ۲

$(AB)(CD) = (20\times 2\times 30) + (30\times 12\times 8) + (20\times 30\times 8) = 8880$

---

۲۱) مرتبه زمانی مسئله کوله پشتی صفر و یک با استفاده از برنامه‌نویسی پویا کدام است؟

۱) $\theta(2^n)$

۲) $\theta(m^n)$

۳) $\theta(n^3)$

۴) $\theta(mn)$

حل) گزینه ۱

چون حداکثر 2^i عنصر در سطر $i-1$ محاسبه می‌شود.

بنابراین حداکثر تعداد عناصر محاسبه شده عبارتست از:

$$1+2+2^2+\cdots+2^{n-1}=2^n-1\in\Theta(2^n)$$

بنابراین تعداد عناصر محاسبه شده در بدترین حالت $\Theta(2^n)$ می باشد.

۲۳) در الگوریتم عقبگرد برای مسئله مدارهای همیلتونی تعداد گره‌ها در فضای حالت چه مقدار می باشد؟

۱) $\dfrac{(n-1)^n+1}{n-2}$

۲) $\dfrac{(n-1)^n-1}{n-2}$

۳) $\dfrac{(n-1)^n+1}{n+2}$

۴) $\dfrac{(n+1)^n+1}{n+2}$

حل) گزینه ۲

تعداد گره‌ها در درخت فضای حالت برای این الگوریتم عبارت است از:

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

۲۵) یک گراف همبند با چند رأس حداقل می تواند $n-1$ یال داشته باشد که اگر فقط $n-1$ یال داشته باشد یک درخت نامیده می شود.

۱) n

۲) $n-1$

۳) $n+1$

۴) $n-2$

حل) گزینه ۱

یک گراف همبند با n رأس، حداقل می تواند $n-1$ یال داشته باشد، که اگر فقط $n-1$ یال داشته باشد، یک درخت نامیده می شود.

۱) تشریحی: الگوریتم مرتب سازی ادغامی (Merge Sort) را با ذکر مثال توضیح دهید؟

پاسخ) در این نوع مرتب سازی، نخست لیست اقلامی که قرار است براساس کلیدهای خاص مرتب شوند به دو قسمت تقسیم می شوند. سپس هر کدام از لیست‌ها دوباره براساس نیاز به زیرلیست‌های کوچکتر تقسیم می شوند، زیرلیست مرتب شده، سپس نتیجه آن‌ها باهم ادغام می شوند. این عمل تا زمانی که

...ن لیست مرتب نشده است، ادامه می یابد.
این روش را مرتب سازی ادغامی می نامند.
در کل منظور از ادغام دو لیست عبارت است از:
فرض کنید دو لیست مرتب موجود است، هدف ایجاد
یک لیست مرتب از ترکیب دو لیست می باشد،
لیست حاصل را ادغام دو لیست اولیه می گویند.

---

۳) تشریحی: فرض کنید A و B دو ماتریس
4×4 با درایه های زیر باشند:

$$A=\begin{bmatrix}1&0&2&3\\3&1&0&2\\1&1&2&0\\0&1&1&1\end{bmatrix}$$

و

$$B=\begin{bmatrix}1&0&1&2\\1&1&2&0\\0&3&1&1\\3&2&0&2\end{bmatrix}$$

حاصل ضرب دو ماتریس را با روش استراسن
انجام دهید.

حل) می خواهیم حاصل ضرب دو ماتریس را با روش
استراسن انجام دهیم. بنابراین هر کدام از ماتریس ها
را به دو قسمت تقسیم می کنیم و داریم:

$$A_{11}=\begin{bmatrix}1&0\\3&1\end{bmatrix} \quad و \quad A_{12}=\begin{bmatrix}2&3\\0&2\end{bmatrix}$$

$$A_{21}=\begin{bmatrix}1&1\\0&1\end{bmatrix} \quad و \quad A_{22}=\begin{bmatrix}2&0\\1&1\end{bmatrix}$$

$$B_{11}=\begin{bmatrix}1&0\\1&1\end{bmatrix} \quad و \quad B_{12}=\begin{bmatrix}1&2\\2&0\end{bmatrix}$$

$$B_{21}=\begin{bmatrix}0&3\\3&2\end{bmatrix} \quad و \quad B_{22}=\begin{bmatrix}1&1\\0&2\end{bmatrix}$$

حال اجزای P و Q و ... V را تشکیل می دهیم:

$$P=\left(\begin{bmatrix}1&0\\3&1\end{bmatrix}+\begin{bmatrix}2&0\\1&1\end{bmatrix}\right)\times\left(\begin{bmatrix}1&0\\1&1\end{bmatrix}+\begin{bmatrix}1&1\\0&2\end{bmatrix}\right)$$

$$Q=\left(\begin{bmatrix}1&1\\0&1\end{bmatrix}+\begin{bmatrix}2&0\\1&1\end{bmatrix}\right)\begin{bmatrix}1&0\\1&1\end{bmatrix}$$

$$R=\begin{bmatrix}1&0\\3&1\end{bmatrix}\left(\begin{bmatrix}1&2\\2&0\end{bmatrix}-\begin{bmatrix}1&1\\0&2\end{bmatrix}\right)$$

$$S=\begin{bmatrix}2&0\\1&1\end{bmatrix}\left(\begin{bmatrix}0&3\\3&2\end{bmatrix}-\begin{bmatrix}1&0\\1&1\end{bmatrix}\right)$$

$$T=\left(\begin{bmatrix}1&0\\3&1\end{bmatrix}+\begin{bmatrix}2&3\\0&2\end{bmatrix}\right)\begin{bmatrix}1&1\\0&2\end{bmatrix}$$

$$U=\left(\begin{bmatrix}1&1\\0&1\end{bmatrix}-\begin{bmatrix}1&0\\3&1\end{bmatrix}\right)\times\left(\begin{bmatrix}1&0\\1&1\end{bmatrix}+\begin{bmatrix}1&2\\2&0\end{bmatrix}\right)$$

$$V=\left(\begin{bmatrix}2&3\\0&2\end{bmatrix}-\begin{bmatrix}2&0\\1&1\end{bmatrix}\right)\left(\begin{bmatrix}0&3\\3&2\end{bmatrix}+\begin{bmatrix}1&1\\0&2\end{bmatrix}\right)$$

با توجه به عبارات بالا مقادیر زیر حاصل می‌شود:

$$P=\begin{bmatrix}۶ & ۳\\ ۱۰ & ۱۰\end{bmatrix},\ S=\begin{bmatrix}۴ & ۱\\ ۳ & ۲\end{bmatrix},\ Q=\begin{bmatrix}۰ & ۱\\ ۲ & ۱\end{bmatrix}$$

$$,\ S=\begin{bmatrix}-۲ & ۶\\ ۱ & ۴\end{bmatrix}$$

$$T=\begin{bmatrix}۳ & ۹\\ ۳ & ۹\end{bmatrix},\ U=\begin{bmatrix}۳ & ۱\\ -۶ & -۶\end{bmatrix},$$

$$V=\begin{bmatrix}۹ & ۱۲\\ ۲ & ۰\end{bmatrix}$$

حال عناصر ماتریس حاصل را با الگوریتم استراسن محاسبه می‌کنیم:

$$C_{۱۱}=\begin{bmatrix}۶ & ۳\\ ۱۰ & ۱۰\end{bmatrix}+\begin{bmatrix}-۲ & ۶\\ ۱ & ۴\end{bmatrix}-\begin{bmatrix}۳ & ۹\\ ۳ & ۹\end{bmatrix}+\begin{bmatrix}۹ & ۱۲\\ ۲ & ۰\end{bmatrix}$$

$$C_{۱۲}=\begin{bmatrix}۰ & ۱\\ ۲ & ۱\end{bmatrix}+\begin{bmatrix}۳ & ۹\\ ۳ & ۹\end{bmatrix}$$

$$C_{۲۱}=\begin{bmatrix}۴ & ۱\\ ۳ & ۲\end{bmatrix}+\begin{bmatrix}-۲ & ۶\\ ۱ & ۴\end{bmatrix}$$

$$C_{۲۲}=\begin{bmatrix}۶ & ۳\\ ۱۰ & ۱۰\end{bmatrix}+\begin{bmatrix}۰ & ۱\\ ۲ & ۱\end{bmatrix}-\begin{bmatrix}۴ & ۱\\ ۳ & ۲\end{bmatrix}+\begin{bmatrix}۳ & ۱\\ -۶ & -۶\end{bmatrix}$$

$$C_{۱۱}=\begin{bmatrix}۱۰ & ۱۲\\ ۱۰ & ۱۵\end{bmatrix},\ C_{۱۲}=\begin{bmatrix}۳ & ۱۰\\ ۵ & ۱۰\end{bmatrix},$$

$$C_{۲۱}=\begin{bmatrix}۲ & ۷\\ ۴ & ۶\end{bmatrix},\ C_{۲۲}=\begin{bmatrix}۵ & ۴\\ ۳ & ۳\end{bmatrix}$$

بنابراین ماتریس حاصل یعنی C برابر خواهد بود با:

$$C=\begin{bmatrix}۱۰ & ۱۲ & ۳ & ۱۰\\ ۱۰ & ۵ & ۵ & ۱۰\\ ۲ & ۷ & ۵ & ۴\\ ۴ & ۶ & ۳ & ۳\end{bmatrix}$$

۵) تشریحی: اشیاء زیر را در نظر بگیرید:

X_i	P_i	W_i
$X_۱$	۸	۱۶
$X_۲$	۵	۱۵
$X_۳$	۱۵	۲۵
$X_۴$	۱۰	۸
$X_۵$	۲۰	۱۵

جواب بهینه‌ای را برای این کوله پشتی بیابید؟

(ظرفیت کوله پشتی را برابر ۴۰ درنظر بگیرید.)

حل) قبل از حل مسئله نسبت P_i/w_i را محاسبه می‌کنیم:

P_i/w_i	$P_۱/w_۱$	$P_۲/w_۲$	$P_۳/w_۳$	$P_۴/w_۴$	$P_۵/w_۵$
	۸/۱۶	۵/۱۵	۱۵/۲۵	۱۰/۸	۲۰/۱۵

حال به ترتیب نزولی جدول بالا را مرتب می‌کنیم. حاصل به صورت ذیل خواهد بود:

x_i	P_i	w_i
$x_۵$	۲۰	۱۵
$x_۴$	۱۰	۸
$x_۳$	۱۵	۲۵
$x_۱$	۸	۱۶
[illegible]	۵	۱۵

مرحله اول: شیء با بالاترین اولویت (براساس $\left(\frac{P_i}{w_i}\right)$) عنصر x_5 می باشد، انتخاب می شود. بنابراین جدول زیر حاصل می شود:

x_i	x_1	x_2	x_3	x_4	x_5
	0	0	0	0	1

و W برابر ۲۵ می شود.

مرحله ی دوم: عنصر با بالاترین اولویت که x_4 است، انتخاب می شود. خروجی حاصل از این مرحله به صورت زیر می باشد:

x_i	x_1	x_2	x_3	x_4	x_5
	0	0	0	1	1

و مقدار W برابر با ۱۷ خواهد بود.

مرحله ی سوم: عنصر x_3 بالاترین اولویت را دارا است ولی وزن آن که ۲۵ می باشد از ظرفیت باقی مانده کوله پشتی، که برابر ۱۷ است بیشتر می باشد. بنابراین حلقه for شکسته می شود و شرط if در الگوریتم اجرا می شود. لذا در نهایت خروجی زیر حاصل می شود:

x_i	x_1	x_2	x_3	x_4	x_5
	0	0	$\frac{17}{25}$	1	1

حال با توجه به جدول خروجی الگوریتم حاصل می شود.

ارزش یا سود حاصل از کوله پشتی را به صورت زیر انتخاب می کنیم:

$$\sum_{i=1}^{5} P_i x_i = (8\times 0)+(5\times 0)+\left(15\times \frac{17}{25}\right)+(10\times 1)+(20\times 1)=40.11$$

لذا ارزش اشیا انتخاب شده در کوله پشتی برابر ۴۰٫۱۱ خواهد بود و جواب بهینه $x=\left(0,0,\frac{17}{25},1,1\right)$ می باشد.

سوالات زوج نیمسال اول ۹۴-۹۳

۲) کدام یک از روابط صحیح است؟

1) if $\lim \frac{T(n)}{f(n)} = 0 \Rightarrow T(n)\in \Theta(f(n))$

2) if $\lim \frac{T(n)}{f(n)} = \infty \Rightarrow T(n)\in O(f(n))$

3) if $\lim \frac{T(n)}{f(n)} \in R^{+} \Rightarrow T(n)\in \Theta(f(n))$

4) if $\lim \frac{T(n)}{f(n)} = 0 \Rightarrow T(n)\in \Omega(f(n))$

حل: گزینه ۳

۵) کدام گزینه صحیح است؟

۱) $3n^2 3^n + 5n^9 2^n \in O(n^2 3^n)$

۲) $n! + 7n^5 \in \Omega(n^n)$

۳) $8n^2 \log n + n^3 \in \Theta(n^2 \log n)$

۴) $n^5 + 14n^3 \in \Omega(n^7)$

حل) گزینه ۱.

---

۶) رابطه بازگشتی زیر از کدام مرتبه زمانی است؟

$$T(n) = 3T\left(\frac{n}{4}\right) + n$$

۱) $O(n)$ ۲) $O(n\log n)$

۳) $O(n^2)$ ۴) $O(n^{\log_4 3})$

حل) گزینه ۱

طرف راست رابطه بالا را طبق روش تکرار با جایگذاری تکرار می‌کنیم، بنابراین:

$$T(n) = 3T\left(\frac{n}{4}\right) + n$$

$$= 3^2 T\left(\frac{n}{16}\right) + 3\left(\frac{n}{4}\right) + n$$

$$= \dots$$

$$\leq 3^i T\left(\frac{n}{4^i}\right) + n\sum_{j=0}^{i-1}\left(\frac{3}{4}\right)^j$$

رابطه بالا را تا زمانی که به $T(1)$ نرسیدیم ادامه می‌دهیم. بنابراین: (با فرض اینکه n توانی از ۴ باشد)

$$\frac{n}{4^i} = 1 \Longrightarrow i = \log_4 n$$

لذا خواهیم داشت:

$$T(n) \leq 3^{\log_4 n} \times T(1) + n\sum_{j=0}^{\log_4 n - 1}\left(\frac{3}{4}\right)^j$$

در عبارت بالا مقدار مجموع، برای nهای بزرگ، ثابت می‌باشد، لذا خواهیم داشت:

$$T(n) \leq c_1 n^{\log_4 3} + c_2 n$$

که در آن c_1 برای $T(1)$ و c_2 برابر است با:

$$\lim_{n\to\infty}\sum_{j=0}^{\log_4 n - 1}\left(\frac{3}{4}\right)^j = 4 \Longrightarrow c_2 < 4$$

بنابراین خواهیم داشت:

$$T(n) \leq c_1 n^{\log_4 3} + 4n$$

$$\Longrightarrow T(n) \in O(n)$$

---

۷) در رابطه با مقایسه الگوریتم‌های مرتب‌سازی ادغامی و سریع کدام گزینه صحیح است؟

۱) پیچیدگی زمانی الگوریتم مرتب سازی سریع برای داده‌های از قبل مرتب شده بهتر از الگوریتم مرتب سازی ادغامی است.

۲) پیچیدگی زمانی الگوریتم مرتب سازی ادغامی در حالت متوسط بهتر از مرتب سازی سریع است.

۳) روش مرتب سازی سریع برخلاف روش مرتب سازی ادغامی به حافظه کمکی نیاز دارد.

۴) پیچیدگی زمانی هر دو روش در بهترین حالت برابر است.

حل) گزینه ۴

پیچیدگی زمانی هر دو روش در بهترین حالت برابر است.

۱۲) بهترین الگوریتم برای ضرب دو چندجمله‌ای از درجه n دارای کدام پیچیدگی زمانی است؟

۱) $O(n^{\log 3})$ ۲) $O(n^2)$

۳) $O(n)$ ۴) $O(n\log n)$

حل) گزینه ۱

بهترین الگوریتم برای ضرب دو چندجمله‌ای از درجه n الگوریتم تقسیم و حل می‌باشد که دارای پیچیدگی زمانی $O(n^{\log 3})$ می‌باشد.

۱۵) در ضرب ماتریس‌ها به روش استراسن، اگر مسأله کوچک ضرب ماتریس‌های ۲×۲ باشد، برای ضرب ماتریس ۴×۴ چند ضرب عددی صورت می‌پذیرد؟

۱) ۴۹ ۲) ۵۶

۳) ۷ ۴) ۲۸

حل) گزینه ۲

چون تبدیل می‌شود به هفت ضرب ماتریس‌های ۲×۲ و هر کدام از ضرب‌های ماتریس‌های ۲×۲ تبدیل می‌شوند به هفت ضرب عددی و هم چنین طبق صورت سؤال، مسأله کوچک ضرب ماتریس‌های ۲×۲ است.

پس: $\overset{49}{(7\times 7)}+7=56$

۱۴) یک گراف همبند و بدون جهت با n گره و $n+2$ یال داریم. کدام یک از الگوریتم‌های زیر برای تولید درخت پوشا با حداقل هزینه بر روی این گراف مناسب‌تر است؟

۱) پریم ۲) کروسکال

۳) دایکسترا ۴) فلوید

حل) گزینه ۲

دو الگوریتم پریم و کروسکال برای این گونه مسائل کاربرد دارند. در صورتی که گراف تنک باشد الگوریتم کروسکال زمان $\Theta(n\log n)$ را صرف می‌کند یعنی سریع‌تر از الگوریتم پریم عمل می‌کند.

۱۷- شش کار به شرح زیر داریم. g_i نشان دهنده سود حاصل از اجرای کار i ام است. اگر و فقط اگر بعد از زمان d_i انجام نشود.

i	۱	۲	۳	۴	۵	۶
g_i	۱۰	۷	۱۵	۲۰	۵	۳
d_i	۱	۳	۱	۳	۱	۳

حداکثر سود حاصل از اجرا چه قدر است؟

۱) ۳۲ ۲) ۳۷

۳) ۴۲ ۴) ۴۵

حل) گزینه ۳

طبق جدول زیر اعمال می‌کنیم.

مرحله	i	بهره	امکان پذیری
۰	۰	۰	✓
۱	۴	۲۰	✓
۲	۳	۲۰	✗
۳	۱	۳۰ ~~۳۵~~	✓
۴	۲	۳۷	✓
۵	۵	۴۲	✓
۶	۳	۴۲	✗

۱۸) تابع زیر را در نظر بگیرید. برای $n > k$ عمل + چند بار انجام می‌شود؟

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

۱) nk ۲) $\binom{n}{k}-1$

۳) $n(n-k)$ ۴) $\binom{n}{k}$

حل) گزینه ۲

چنانچه $T(n,k)$ را تعداد اعمال جمع در نظر بگیریم آن گاه $T(n,k)=\binom{n}{k}-1$ خواهد بود. علت این امر این است که هر عامل باید آن قدر بسط داده شود که در نهایت به 1 برسد و یک‌ها را باید با هم جمع کرد. قطعاً باید $\binom{n}{k}$ عدد یک را با هم جمع کنیم که $\binom{n}{k}$ می‌شود، در این صورت تعداد اعمال جمع $\binom{n}{k}-1$ خواهد بود.

۲۰) فرض کنید $X = aabab$ ، $Y = babb$ و ارزش اعمال درج و حذف یک واحد و ارزش عمل تغییر دو واحد باشد. ارزش بهینه تبدیل X به Y کدام است؟

۱) ۲ ۲) ۳

۳) ۴ ۴) ۵

حل) گزینه ۲ :

ابتدا طولانی ترین زیررشته مشترک x و y را پیدا می کنیم که عبارت است از bab.

درنتیجه به X، در آخرش b اضافه می کنیم (درج می کنیم) و اولین a را به b تبدیل می کنیم. پس: ۳=۱+۲

۲۲) کدام گزینه صحیح است؟

۱) در روش انشعاب و تحدید، مجموعه ای از جواب های بهینه به دست می آید ولی در روش عقبگرد، معمولاً هدف یافتن بهینه ترین جواب است.

۲) تعداد گره ها در درخت فضای حالت تولید شده به روش انشعاب و تحدید بیشتر از روش عقبگرد است.

۳) زمان اجرای الگوریتم های عقبگرد در بدترین حالت از الگوریتم های انشعاب و تحدید بهتر است.

۴) الگوی جستجو در روش عقبگرد، روش جستجوی عمقی است ولی در روش انشعاب و تحدید، جستجوی ردیفی است.

حل) گزینه ۴

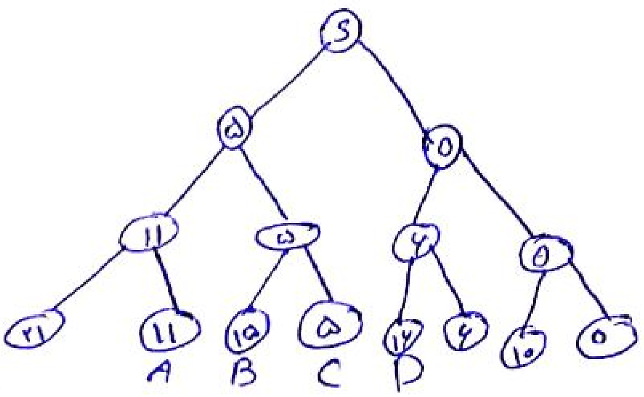
الگوی جستجو در روش عقبگرد، روش جستجوی عمقی است ولی در روش انشعاب و تحدید، جستجوی ردیفی است.

۲۴) در مسأله حاصل جمع زیرمجموعه‌ها، فرض کنید $n = ۵$ و $W = ۲۱$، اعداد داده شده به صورت زیر است:

$w_1 = ۵$ ، $w_2 = ۶$ ، $w_3 = ۱۰$

$w_4 = ۱۱$ ، $w_5 = ۱۶$

کدام یک از گره های درخت فضای حالت این مسأله، امیدبخش و قابل توسعه دادن است؟

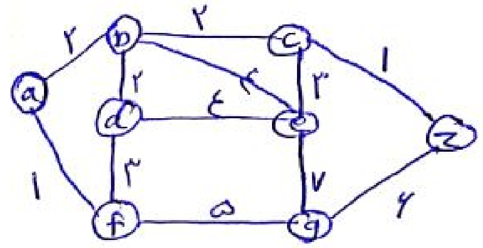


۱) A ۲) B

۳) C ۴) D

حل) گزینه ۳

۲) تشریحی: در گراف زیر، کوتاه ترین مسیر از رأس a به تمام رئوس را به کمک الگوریتم دکسترا بدست آورید. اجرای الگوریتم را مرحله به مرحله نشان دهید.



حل)

	S	d	P
a	1	-	-
b	1	۲	ab
c	1	۴	abc
d	1	۴	afd
e	1	۶	abe
f	1	1	af
g	1	۶	afg
z	1	۵	abcz

۴) در مسأله رنگ آمیزی گراف، هدف رنگ آمیزی گره های گراف G(V,E) با استفاده از m رنگ است، به طوری که هیچ دو گره مجاور همرنگ نباشند. با استفاده از روش عقب گرد، الگوریتم کاملی را برای حل این مسئله بنویسید و مرتبه زمانی الگوریتم را در بدترین حالت تحلیل نمایید.

حل) مسأله: ورودی: اعداد مثبت n و m که نشان دهنده تعداد رئوس گراف و m تعداد رنگ‌ها. گراف توسط یک آرایه دو بعدی W نشان داده می شود که سطرها و ستون های آن از یک تا n اندیس گذاری شده اند که در آن W[i][j] دارای مقدار True است. اگر بین رأس i و j یک یال وجود داشته باشد و در غیر این صورت دارای مقدار False است.

خروجی: همه‌ی رنگ آمیزی های ممکن برای گراف با استفاده از حداکثر m رنگ. خروجی مربوط به هر رنگ آمیزی یک آرایه vcolor است که در آن vcolor[i] رنگ نسبت داده شده به رأس iام (یک عدد صحیح بین یک تا m) است.

```
void mcolor (index i)
{
    int color;
    if( Promising(i))
        if( i = =n)
            cout << vcolor [i] through
                    vcolor [n];
        else
            for( color=1; color <= m
                ; color++)
```

```
{
  vcolor [i+1] = color;
  mcolor (i+1);
    }

}

bool  Promising (index i)
{

      index j;
      bool flag;
      j = 1;
      while ( j< i && Flag)
      {
        if (W[i][j] && vcolor. [i] == vcolor[j])
           Flag = False;

        j++
      }
      return  Flag;
   }
```

تعداد گره ها در درخت فضای حالت برای این الگوریتم برابر است با:

$$1 + m + m^2 + \cdots + m^n = \frac{m^{n+1} - 1}{m - 1}$$

با توجه به عبارت فوق پیچیدگی زمانی الگوریتم نمایی است. یا به عبارت دقیق تر برای $m \geq 3$ هیچ کس تا کنون الگوریتمی مطرح نکرده است که در بدترین حالت بهتر از نمایی باشد.