

سئوالات فرد 96-97-2

سؤال 1. تابع زیر را در نظر بگیرید ، پیچیدگی تابع زیر کدام است ؟

```
l=n
While (l>1)
{
J=l
While (J,>n)
J=2*j;
i--;
}
```

1. $O(\log \log n)$ 2. $O(\log n)$ 3. $O(n)$ 4. $O(n \log n)$

جواب : گزینه 4

حلقه While بیرونی از مرتبه $O(n)$ است .
در حلقه While اگر شمارنده با دستور $i=j*k$ تغییر کند ، مرتبه اجرایی آن $O(\log_k n)$ است .
حلقه while درونی نیز از مرتبه $O(\log_2 n)$ می باشد .
بنابراین ، الگوریتم بالا از مرتبه $O(n \log_2 n)$ است .

سؤال 3. در روش جستجوی خطی برای یافتن یک عنصر ، درون آرایه n عنصری به چه تعداد مقایسه نیاز است ؟ ج

1. n 2. $n+1$ 3. $n-1$ 4. 1

کلیه جواب ها غلط می باشد .

بهترین حالت این است که عنصر مورد جستجو در خانه اول آرایه باشد یعنی

$$T(n)=O(1)$$

بدترین حالت این است که عنصر مورد جستجو در خانه آخر آرایه باشد یعنی

$$T(n)=O(n)$$

به طور متوسط به $\frac{n+1}{2}$ مقایسه نیاز است .

سؤال 5. مرتبه پیچیدگی زمانی تابع زیر چیست ؟
جواب : گزینه 2

```
Int f(int n)
{
  If(n==1)
  Return 1;
Else
  Return (n+f(n-1));
}
```

1. n^2 2. n 3. $n \log n$ 4. $n^2 \log n$

زمان اجرای تابع $f(n)$ را $T(n)$ در نظر می گیریم .

$$6. T_{(n)} = \begin{cases} o(1) & \text{if } n = 0 \\ o(1) + T(n-1) & \text{if } n > 0 \end{cases}$$

7.

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

در حالت کلی داریم :

$$T(n)=aT(n-k)+C \rightarrow T_{(n)} = \begin{cases} T(n) = O\left(a^{\frac{n}{k}}\right) & \text{if } a \neq 1 \\ T(n) = O(n) & \text{if } a = 1 \end{cases}$$

بنابراین زمان اجرای تابع بازگشتی $f(n)$ برابر $O(n)$ است و گزینه 2 صحیح است .

سؤال 7 . داده 54 را با چندمین مقایسه در آرایه مقابل با روش bin search پیدا می کنیم ؟

(2,8,10,18,18,20,30,54)

1 .4

7 .3

4 .2

3 .1

جواب : گزینه اول صحیح است .

1	2	3	4	5	6	7
2	8	10	18	20	30	54

جایگاه اول + جایگاه آخر = 1+7 تقسیم بر 2 مساوی ست با = 4 mid

$$(1+7)/2=4$$

Mid

54>18 لذا در نیمه دوم آرایه جستجو می کنیم :

20	30	54
----	----	----

با در نظر داشتن خانه های پنجم و هفتم :

$$(5+7)/2=6$$

mid

عدد خانه 6 ام مساوی ست با 30

30<54 بنابراین در نیمه دوم دوم آرایه جستجو می کنیم :

54

با 3 مقایسه به داده 54 رسیدیم .

سؤال 9 . بهترین الگوریتم ضرب دو عدد n بیتی دارای کدام یک از پیچیدگی ها می باشد ؟ د

4 . $O(n^{\log_2 3})$

3 . $O(n \log n)$

2 . $O(n^2)$

1 . $O(n)$

$$T(n)=3T\left(\frac{n}{2}\right)+Cn \rightarrow T(n)=\Theta(n^{\log_2 3})$$

بنابراین گزینه 4 صحیح است .

سؤال 11. می خواهیم از روش استراسن ، دو ماتریس را در هم ضرب کنیم ، کدام گزینه صحیح نیست ؟

- 1- پیچیدگی مسئله کاهش می یابد .
- 2- استراسن از روش تقسیم و حل استفاده می کند .
- 3- استراسن یک روش خاص می باشد و همواره کاربرد ندارد .
- 4- استراسن مرتبه ضرب را از n^3 کاهش می دهد .

جواب : گزینه 3

4 صحیح است

روش استراسن به 7 عمل ضرب و 18 عمل جمع و تفریق نیاز دارد . به صورت کلی در روش تقسیم و حل ابتدا مسئله به دو یا چند زیر مسئله کوچکتر تقسیم می شود ، اگر این زیر مسئله ها هنوز بزرگ بودند و لازم بود که باز هم به زیر مسئله های کوچکتر تقسیم شوند این کار را انجام می دهیم .
روش استراسن در مقایسه با روش استاندارد که از مرتبه $\theta(n^3)$ می باشد ، کارآمدتر است .
بنابر این گزینه 3 اشتباه است .

سؤال 13. با توجه به الگوریتم زمان بندی با مهلت معین ، اگر مهلت ها و زمان ها و بهره ها به صورت جدول زیر باشد ، بیشترین قابل دستیابی ، کدام یک از موارد زیر است ؟

کار	مهلت	بهره
1	2	30
2	1	35
3	2	25
4	1	40

70 .4

65 .3

90 .2

60 .1

سؤال 17 . کدام یک از الگوریتم های زیر بر اساس روش برنامه نویسی پویا نیست ؟

1. ایجاد درخت جستجوی دودویی بهینه

2. الگوریتم پریم

3. ضرب زنجیره ی ماتریس ها

4. فروشنده ی دوره گرد

جواب : گزینه 2 - الگوریتم پریم

درخت پوشای کمینه (پریم و کراسکال) با روش حریصانه حل می شوند .

سؤال 19 . پیچیدگی الگوریتم فیبوناتچی که به روش برنامه نویسی پویا پیاده سازی شده است متعلق به کدام مرتبه زمانی است ؟ آیا این پیچیدگی را باز می توان کاهش داد ؟

1. $\log n$ و خیر

2. $\log n$ و بله

3. n و بله

4. n و خیر

جواب : گزینه 4

گزینه 4 صحیح است .

سؤال 21 . در مسئله رنگ آمیزی گراف اگر n نشان دهنده تعداد رئوس گراف و m تعداد رنگ ها باشد ، درخت فضای حالت حداکثر چند گره خواهد داشت ؟

1. $m^{n+1}-1/m$ 2. $m^{n+1}-1/m-1$ 3. $m^{n+1}+1/m$ 4. $m^{n+1}+1/m+1$

جواب : گزینه 2

تعداد گره ها در درخت فضای حالت :

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

سؤال 23. کدام یک از روش های پیمایش زیر برای گراف ها وجود دارد ؟
1. BFS 2. DFS 3. BFS, DFS 4. LFS , DFS , BFS

جواب : گزینه 3

BFS : جستجو در سطح و DFS : جستجو در عمق می باشد . گزینه 3 صحیح است .

سؤال 25 . گزینه نادرست کدام است ؟

1. تمام مسائل P به وسیله یک الگوریتم غیرقطعی در زمان چندجمله ای حل می شوند

2. تمام مسائل NP به وسیله یک الگوریتم غیرقطعی در زمان چندجمله ای حل می شوند .

3. تمام مسائل Np-hard به وسیله یک الگوریتم غیرقطعی در زمان چندجمله ای حل می شوند .

4. تمام مسائل NP-complete به وسیله یک الگوریتم غیرقطعی در زمان چندجمله ای حل می شود .

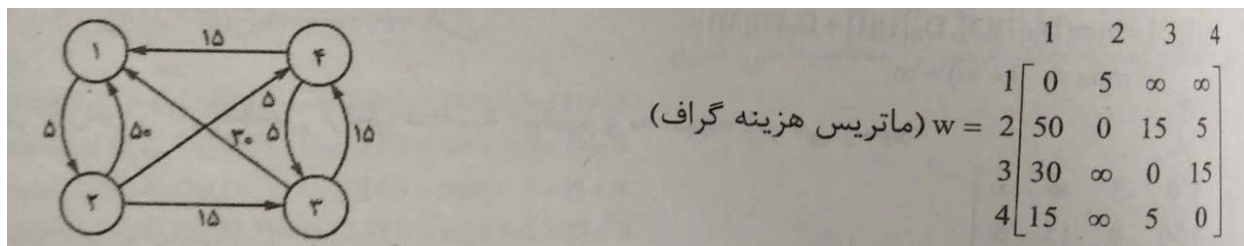
جواب : گزینه 3

گزینه 2 درست بوده و می دانیم که $P \subseteq NP$ پس گزینه 1 نیز درست است و چون $NP-Complete \subseteq NP$ پس 4 هم درست است و جواب گزینه 3 است (که غلط است)

سئالات تشریحی :

سؤال 1. الگوریتم فلوید را با ذکر یک مثال توضیح داده و از نظر پیچیدگی زمانی بررسی نمایید ؟

مسئله متداول در چنین گراف هایی پیدا کردن کوتاهترین مسیر از یک رأس به رأس دیگر است . چون ممکن است بیش از یک کوتاهترین مسیر از رأسی به رأس دیگر وجود داشته باشد مسئله ما یافتن هر یک از این کوتاهترین مسیرهاست .
مثلا در گراف مقابل با ماتریس مجاورت زیر :



هدف این است که کوتاهترین مسیر را بیابیم .
 مثلاً در گراف بالا کوتاهترین مسیری که رأس 3 را به 2 وصل کند و از رأس 1 هم بتواند بگذرد دارای طول 35 و کوتاهترین مسیری که رأس 3 را به 4 وصل کند و مجاز به عبور از دیگر رئوس نیز باشد دارای طول 15 است .

اگر بخواهیم این مسئله را با بررسی همه حالات حل کنیم مرتبه زمانی اجرای آن فاکتوریل خواهد شد . ولی با روش برنامه نویسی پویا ، الگوریتمی با زمان بهتر ارائه خواهیم داد . و در این مسئله اصل بهینگی برقرار است .

طبق فرمول بازگشتی ، و یا داشتن D_0 ، ابتدا D_1 ، سپس D_2 و ... D_n را بدست می آوریم .

این ماتریس نشان دهنده همه کوتاهترین مسیرها از رأس i به j است .

$$D_k[i][j] = \min(D_{k-1}[i][j], D_{k-1}[i][k] + D_{k-1}[k][j])$$

برای بدست آوردن خود مسیر بهینه نیز نیاز به آرایه دو بعدی P داریم که هر بار بعد از محاسبه $D_k[i][j]$ آن مقدار از K و $K-1$ را که مینیمم است به خود اختصاص می دهد .

سؤال 3 . برای مسئله کوله پشتی صفر و یک با ظرفیت کوله پشتی $W=13$ و پنج قطعه داده شده به صورت جدول زیر با استفاده از روش انشعاب و تحدید مسئله را حل کرده و درخت فضای حالت هرس شده را برای آن رسم نمایید .

i	P_i	W_i	
1	20	2	10
2	30	5	6
3	12	3	4
4	3	1	3
5	35	7	5

سؤال 5. کلاس NP و مجموعه مسائل مربوط را تعریف کنید و دو مثال بزنید .

برای مسائل کلاس NP باید کامپیوتر علاوه بر توانایی اجرای دستورهای معین و قطعی ، قادر باشد برخی دستورات نامعین را نیز اجرا کند . یک دستور نامعین دستوری است که نتیجه اجرای آن از قبل قابل پیش بینی نباشد . برای مثال فرض کنید دستوری داشته باشیم که بخواهد از بین 100 شیء یکی را انتخاب کند . قبل از اجرای چنین دستوری نمی توان پیش بینی کرد که دقیقا کدامیک از اشیاء انتخاب خواهد شد . چنین کامپیوتری را یک کامپیوتر نامعین می نامند . الگوریتمهایی که برای یک کامپیوتر نامعین طراحی می شوند الگوریتمهای نامعین نامیده می شوند . الگوریتمهای نامعین علاوه بر دستورهایی که برای بیان الگوریتم معین وجود دارد ، باید دستورات دیگر را نیز اضافه کند . معمولا دستورهای زیر به الگوریتمهای معین اضافه می شود تا الگوریتم به نامعین تبدیل شود :

الف) تابعی که یکی از عناصر مجموعه S را به دلخواه انتخاب می کند . اولین مرحله ، مرحله حدس زدن نامیده می شود و ممکن است خروجی حاصل از این مرحله بی ربط باشد .

ب) حدس انتخاب شده و مجموعه S ورودی این تابع می باشد . خروجی این تابع ، پایان موفق یا ناموفق عملیات الگوریتم را اعلام می کند . این مرحله ، مرحله تصدیق نامیده می شود .

در بیان یک الگوریتم نامعین لزومی ندارد از همه دستورها و تابع های ذکر شده استفاده شود . بنابراین هر الگوریتم معین توسط یک کامپیوتر نامعین قابل اجرا است .

مسائل NP :

NP مجموعه تمام مسائل تصمیم گیری است که توسط الگوریتمهای غیر قطعی با زمان چندجمله ای قابل حل باشند . NP نشان دهنده چند جمله ای غیر قطعی است . برای آنکه یک مسأله تصمیم گیری در NP باشد ، باید الگوریتمی وجود داشته باشد که عمل تصدیق را در زمان چندجمله ای انجام دهد . تاکید بر این نکته ضروری است که این بدان معنا نیست که الزاما الگوریتمی با زمان چندجمله ای برای حل مسأله داریم . همانطور که چنین الگوریتمی برای مسأله فروشنده دوره گرد نداریم .