

"به نام خدا"

سوالات فرد نیمسال دوم 97-98

مهدیه احمدی پور

شماره دانشجویی:

970084828

سوال 1: هزینه اجرایی تابع زیر چقدر است؟

```
Bool func(int n){  
  For (int i=2; i<sqrt(n);i++)  
    If (n%i==0)return 0;  
  Return 1;
```

$O(n^2)$ (1

$O(n^0)$ (2

$O(n^3)$ (3

$O(\log n)$ (4

جواب صحیح گزینه ب

راه حل: در برنامه داده شده حلقه for از 2 تا  $n^{\frac{1}{2}}$  تکرار می گردد در بدترین حالت که شرط if اجرا نگردد حلقه به تعداد  $n^{\frac{1}{2}}$  بار اجرا خواهد شد لذا هزینه ی اجرایی برابر با گزینه ب است.

سوال 2: -

سوال 3: کدام گزینه صحیح است؟

$5n^{10}+2^n+\log n \in o(n^{100})$ (1

$10^n+2n^{1n} \in \theta(n^{1n})$ (2

$N^2 \log n + 5^n \in \Omega(2^n)$ (3

$1 \frac{n(n+1)(2+1)}{6} \in \Omega(n^4)$ (4

جواب صحیح گزینه ج

راه حل: توابع بر اساس درجه به صورت صعودی مرتب می شوند:

$1 \log n \ n \log n \ n^2 \ n^2 \log n \dots n^3 \ n^4 \dots n^a \dots 2^n \ 3^n \dots$   
 $a^n \ n! \ n^n$

$O$  یعنی بزرگترین درجه چندجمله ای و درجات بزرگتر از آن

$\Omega$  یعنی بزرگترین درجات چندجمله ای و درجات کوچکتر از آن

$\theta$  یعنی فقط بزرگترین درجه چند جمله ای

در گزینه اول بزرگترین درجه چندجمله ای  $2^n$  می باشد و  $O$  باید بزرگتر مساوی آن باشد ولی  $n^{100}$  کوچکتر است لذا صحیح نیست.

در گزینه دوم بزرگترین درجه چندجمله ای  $10^n$  می باشد و  $\theta$  باید مساوی آن باشد لذا صحیح نمی باشد.

در گزینه سوم بزرگترین درجه چندجمله ای  $5^n$  می باشد و  $\theta$  باید کوچکتر مساوی آن باشد لذا صحیح می باشد.

در گزینه چهارم بزرگترین درجه چندجمله ای  $n^3$  می باشد و  $\Omega$  باید کوچکتر مساوی آن باشد در حالی که  $n^4$  درجه بزرگتر است و صحیح نمی باشد.

سوال 4: -

سوال 5: تابع بازگشتی زیر چه کاری انجام می دهد؟

```
Int c (Node* root){  
    If (!root) return 0;  
    If (root→lchild && root→rchild);  
        Return c(root→lchild)+c(root→rchild);  
    Return 1+c(root→lchild)+c(root→rchild);  
}
```

1) شمارش کل گره های درخت به جز گره های دو فرزندی

2) شمارش کل گره های درخت

3) شمارش تعداد گره های دو فرزندی

4) شمارش تعداد گره های تک فرزندی

جواب صحیح گزینه الف

راه حل: همانطور که در تابع C برای هر گره مشاهده می شود در if دوم (خط سوم) بررسی می شود اگر فرزند چپ و راست گره تهی نباشد (گره های دو فرزندی) تابع را به طور جداگانه برای فرزند چپ و راست فراخوانی کرده مقدار برگشتی آن دو را جمع می کند در غیر این صورت (برای گره های برگ و تک فرزندی): عدد یک را به مجموعه می افزاید (یعنی آن گره را می شمارد) و سپس مقدار برگشتی فرزند چپ و راست آن را با هم جمع میکند و به مجموع می افزاید. در نتیجه تابع زمانی گره هارا می شمارد که فرزندی نباشد.

سوال 6:-

سوال 7: کدام گزینه ویژگی دو روش تقسیم و حل و برنامه نویسی پویا را به درستی بیان می کند؟

1) روش تقسیم و حل یک روش پایین به بالا و روش برنامه نویسی پویا یک روش بالا به پایین است.

2) برخلاف برنامه نویسی پویا در روش تقسیم و حل نتایج حل مسائل در هر سطح نگه داری می شود.

3) برخلاف روش تقسی و حل در روش برنامه نویسی پویا یک نمونه کوچک ممکن است چندین بار حل شود.

4) در روش تقسیم و حل برای مساله سطح  $L$  تنها از مسائل سطح  $L-1$  استفاده می شود درحالی که در برنامه نویسی پویا برای حل مساله سطح  $L$  از کلیه مسائل سطح پایین تر استفاده می شود.

جواب صحیح گزینه د

راه حل: اصل بهینگی برای مسائل روش پویاست و در روش تقسیم و حل ممکن است برقرار نباشد. روش برنامه نویسی پویا برای حل مسائل سطح  $L$  از سطوح قبلی و همچنین خود سطح  $L$  می تواند استفاده کند ولی روش تقسیم و حل برای حل مساله سطح  $L$  تنهای از مسائل سطح  $L-1$  استفاده می کند. روش تقسیم و حل یک مساله بزرگ را به مسائل کوچکتری تقسیم میکند تا اینکه مسائل کوچک قابل حل باشند. (روش بالا به پایین) اما در روش پویا ریز مسائل باهم ترکیب شده و مسائل بزرگتر را ایجاد میکنند (روش پایین به بالا)

سوال 8: -

سوال 9: در ضرب دو ماتریس  $4 \times 4$  به روش استراسن و روش معمولی چ عمل جمع و تفریق انجام می شود؟

1) استراسن = 56 معمولی = 64

2) استراسن = 72 معمولی = 48

3) استراسن = 54 معمولی = 16

4) استراسن = 18 معمولی = 4

جواب صحیح گزینه ب

راه حل: تعداد جمع در روش استراسن برابر با  $2^{(n/2)} T(18)$  است که اگر  $n=4$  باشد 72 جمع و تفریق خواهیم داشت. در روش معمولی  $n^3 - n^2$  خواهیم داشت که برابر با 48 خواهد بود.

سوال 10: -

سوال 11: الگوریتم زیر برای محاسبه جمله  $n$ م سری فیبوناچی به کار می رود روش مورد استفاده در طراحی این الگوریتم چیست؟

```
Int f(int n){  
    If (n==1 || n==2) return 1;  
    Int *a=new int [n];  
    a[0]=a[1]=1;  
    for (int i=2; i<n; i++)  
        a[i]=a[i-1]+a[i-2];  
    return a[n-1];  
}
```

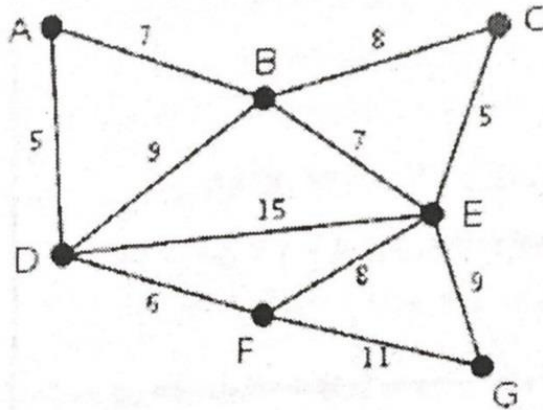
- 1) برنامه نویسی پویا
- 2) حریصانه
- 3) تقسیم و حل
- 4) عقبگرد

جواب صحیح گزینه الف

راه حل: الگوریتم داده شده از آرایه برای نگهداری نتایج استفاده میکند و چون برای به دست آوردن نتیجه سطح  $i$  از سطوح قبلی استفاده می کند ( $i-1$  و  $i-2$ ) لذا متناظر با روش برنامه نویسی پویاست.

سوال 12: -

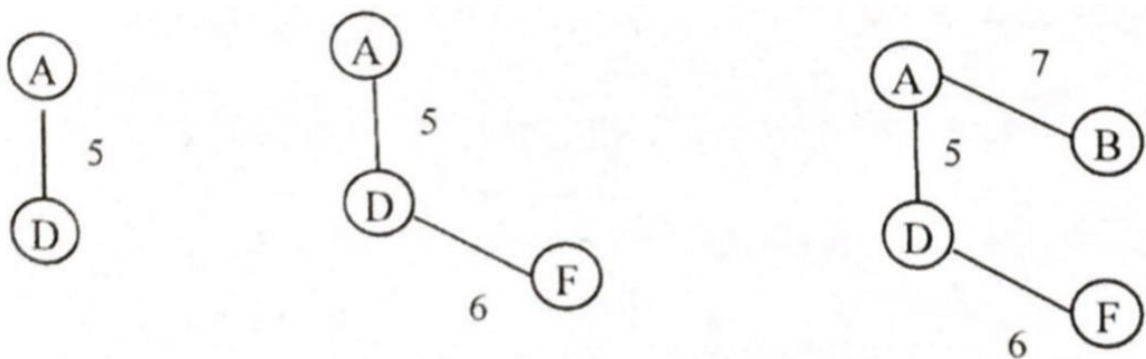
سوال 13: اگر از الگوریتم پریم برای به دست آوردن درخت پوشای  
مینم گراف زیر استفاده شود کدام راس در مرحله سوم استفاده می  
شود؟ (شروع از d)



- A(1
- E(2
- B(3
- F(4

جواب صحیح گزینه ج

راه حل: در الگوریتم پریم در هر بار راس جدیدی به مجموعه اضافه  
میگردد با این شرط که این راس با وزن یال کمتری به مجموعه وصل  
شود. ابتدا A را در نظر میگیریم از مجموعه رئوسی که به آن متصل  
است راس D با کمترین وزن انتخاب می شود سپس از مجموعه  
رئوس دیگری که به رئوس A و D متصل است راس D با کمترین  
وزن انتخاب می شود. سپس از مجموعه رئوس دیگری که به رئوس  
A و D متصلند F به کمترین وزن انتخاب می شود راس سوم خواهد  
بود.



سوال 14: -

سوال 15: گرافی با ماتریس مجاورت زیر مفروض است. طول کوتاه ترین مسیر از راس 1 به 4 با استفاده از الگوریتم دیکسترا کدام است؟

$$W = \begin{matrix} & \begin{matrix} 1 & 2 & 3 & 4 \end{matrix} \\ \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \end{matrix} & \begin{bmatrix} 0 & 2 & 9 & \infty \\ 1 & 0 & 6 & 4 \\ \infty & 7 & 0 & 8 \\ 6 & 3 & \infty & 0 \end{bmatrix} \end{matrix}$$

14(1

6(2

8(3

9(4

جواب صحیح گزینه ب

راه حل: ابتدا جدول مسیر مربوط به ماتریس را ترسیم می نماییم ابتدا نخستیت کوتاه ترین مسیر از  $V_1$  انتخاب می شود. با توجه به ماتریس مشخص می شود که  $V_2$  بهترین گره انتخابی می باشد. این مسیر را انتخاب سپس جدول را با استفاده از مسیر جدید به روز رسانی می کنیم:

| گره ها | s | dis      | p  |
|--------|---|----------|----|
| V1     | 1 | 0        | V1 |
| V2     | 0 | 2        | V1 |
| V3     | 0 | 9        | V1 |
| V4     | 0 | $\infty$ | v  |



| گره ها | s | dist | p      |
|--------|---|------|--------|
| V1     | 1 | 0    | V1     |
| V2     | 1 | 2    | V1v2   |
| V3     | 0 | 8    | V1v2v3 |
| V4     | 0 | 6    | V1v2v4 |



سپس کوتاه ترین مسیر بعدی را انتخاب می کنیم که مسیر با مقصد  $V_3$  می باشد و بعد از آن مسیر با مقصد  $V_4$  می باشد. جدول در هر لحظه به روز رسانی می شود:

| گره ها | s | dist | p      |
|--------|---|------|--------|
| V1     | 1 | 0    | V1     |
| V2     | 1 | 2    | V1v2   |
| V3     | 0 | 8    | V1v2v3 |
| V4     | 0 | 6    | V1v2   |



| گره ها | s | dist | p      |
|--------|---|------|--------|
| V1     | 1 | 0    | V1     |
| V2     | 1 | 2    | V1v2   |
| V3     | 0 | 8    | V1v2v3 |
| V4     | 0 | 6    | V1v2v4 |

مشاهده می شود طول کوتاه ترین مسیر از راس 1 به 4 برابر با 6 است. با مسیر  $V_1 V_2 V_4$

سوال 16: -

سوال 17: تحلیل پیچیدگی زمانی در بدترین حالت برای الگوریتم فلوید چقدر است؟

$\theta(n^2)$ (1

$\theta(n^n)$ (2

$\theta(n^3)$ (3

$\theta(3^n)$ (4

جواب صحیح گزینه ج

راه حل: مرتبه زمانی الگوریتم فلویید گزینه سوم می باشد.

سوال 18 : -

سوال 19: فرض کنید سه کلید موجود اس. اگر احتمال مساوی بودن کلید مورد جستجو با هر یک از کلیدها به صورت زیر باشد حداقل زمان جستجوی میانگین برای درخت جستجوی دودویی چقدر است؟

Key1=0.7

Key2=0.2

Key3=0.1

2.1(1

1.8(2

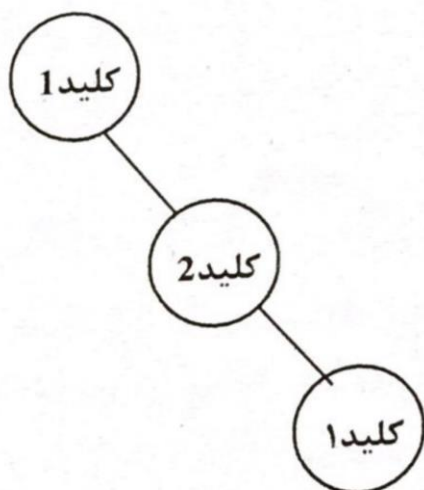
1.7(3

1.4(4

جواب صحیح گزینه د

راه حل:

$$1 \times 0.7 + 2 \times 0.2 + 3 \times 0.1 = 1.4$$



سوال 20: -

سوال 21: برای حل مساله  $n$  وزیر به روش عقبگرد اگر وزیر  $m$  در ستون  $j$  قرار داشته باشد در چه صورت مورد حمله وزیر  $k$  در ستون  $i$  قرار می گیرد؟

$$j=k(1)$$

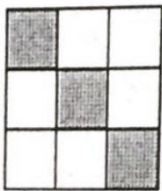
$$i+j=k+1(2)$$

$$i=1(3)$$

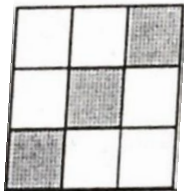
$$j-i=k-1(4)$$

جواب صحیح گزینه ب

راه حل:



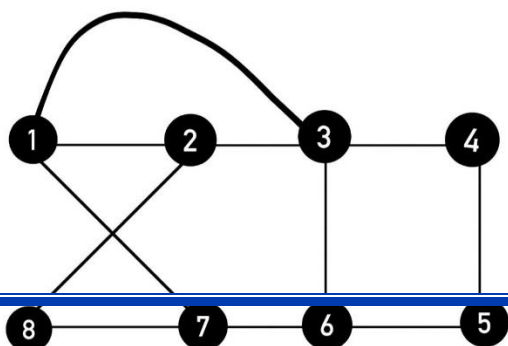
$$I-K=J-L \rightarrow I=J=K-L$$



$$I-K=L-J \rightarrow I+J=L+K$$

سوال 22: -

سوال 23: گراف با ماتریس مجاورت زیر را در نظر بگیرید. برای یافتن یک دور هامیلتونی از راس به خودش با استفاده از روش عقبگرد. تعداد کل گره ها و تعداد گره های امید بخش در دومین سطح درخت فضای حالت چقدر است؟ (ریشه در سطح یک قرار دارد)



- 1) کل گره ها = 5 . گره های امیدبخش = 3
- 2) کل گره ها = 4 . گره های امیدبخش = 4
- 3) کل گره ها = 5 . گره های امیدبخش = 4
- 4) کل گره ها = 3 . گره های امیدبخش = 2

جواب صحیح گزینه د

راه حل: دور همیلتونی از یک راس دوری است که از هر راس فقط یک بار عبور کرده باشد.

در الگوریتم عقبگرد اگر راس 1 در سطح 1 باشد در سطح 2 سه راس متاصل به آن وجود خواهد داشت: راس 2 و 3 و 7  
دور همیلتونی از این سه مسیر عبارت است از:

1,2,8,7,6,5,4,3,1

1,3,4,5,6,7,8,2,1

اما برای مسیر ... 1,7 نمی توان دوری یافت لذا دو گره امیدبخش وجود دارد.

سوال 24: -

سوال 25: کدام یک از مسائل زیر در کلاس NP قرار دارد؟

- 1) ضرب زنجیره ای ماتریس ها
- 2) تعیین کلیه مدارهای هامیلتونی یک گراف
- 3) حاصل جمع زیرمجموعه ها

4) کوله پشتی کسری

جواب صحیح گزینه ج

راه حل: مرتبه زمانی حاصل جمع زیر مجموعه ها نمایی است لذا در کلاس NP قرار دارد.

## "سوالات تشریحی"

سوال 1: رابطه بازگشتی زیر را حل کنید.

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

راه حل تشریحی:

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

$$\rightarrow T(n) = 2(2T(n-2) + 3^{n-1}) + 3^n$$

$$\rightarrow T(n) = 2^2 T(n-2) + 2 \times 3^{n-1} + 3^n$$

$$\rightarrow T(n) = 2^2 (2T(n-3) + 3^{n-2}) + 2 \times 3^{n-1} + 3^n$$

$$\rightarrow T(n) = 2^3 T(n-3) + 2^2 \times 3^{n-2} + 2 \times 3^{n-1} + 3^n$$

$$\rightarrow T(n) = 2^k T(n-k) + (3^n - 2^n) 3^{n-k+1}$$

$$\rightarrow T(n) = 2^n T(0) + 3(3^n - 2^n)$$

$$\rightarrow T(n) = 2^n + 3(3^n - 2^n) \rightarrow T(n) = 3^{n+1} - 2^{n+1}$$

## سوال 2: -

سوال 3: هفت کار به شرح زیر داریم سود حاصل از کار  $i$ م است به شرطی که بعد از زمان انجام نشود. با این فرض که هر کار در واحد زمان انجام می شود ترتیب انجام کارها و حداکثر سود حاصل از اجرا را به دست آورید.

| i | 1  | 2  | 3 | 4  | 5  | 6  | 7  |
|---|----|----|---|----|----|----|----|
|   | 85 | 50 | 9 | 53 | 66 | 40 | 35 |
|   | 2  | 4  | 4 | 1  | 2  | 3  | 2  |

راه حل تشریحی:

ابتدا کارها را بر اساس سود به صورت نزولی مرتب می نماییم سپس با استفاده از جدول الگوریتم زمانبندی با مهلت معین با انتخاب کارها به ترتیب از بالا به پایین، مجموعه امکان پذیر را به دست می آوریم. کارها به ترتیب بررسی می شوند، اگر مطابق با مهلت مشخص شده برای آن کار، زمانی پیدا شد که کارهای دیگر اجرا نمی شود آن کار را به مجموعه اضافه می کنیم، مثلا اگر مهلت کاری ۳ باشد و دو کار با مهلت های ۱ و ۳ قبلا انتخاب شده باشند، کار با مهلت ۱ به ناچار در زمان ۱ اجرا می شود کار با مهلت ۳ می تواند در زمان های ۲ یا ۳ اجرا شود پس در یکی از زمان ها اجرا می شود و کار جدید با مهلت ۳ می تواند در زمان بعدی اجرا شود. مثال دیگر: اگر ۲ کار قبلا با مهلت های ۱ و ۳ انتخاب شده باشند کار جدید با مهلت ۱ نمی تواند در یکی از زمان ها تا آن مهلت اجرا شود.

| امکان پذیر | مجموعه    | سود | مهلت | کار |
|------------|-----------|-----|------|-----|
| هست        | {3}       | 90  | 4    | 3   |
| هست        | {1,3}     | 85  | 2    | 1   |
| هست        | {3,5,1}   | 66  | 2    | 5   |
| هست        | {5,1,3,2} | 50  | 4    | 2   |
| هست        | {2,5,1,3} | 40  | 3    | 6   |

|   |   |    |           |     |
|---|---|----|-----------|-----|
| 7 | 2 | 35 | {2,5,1,3} | هست |
|---|---|----|-----------|-----|

جمع ارزش های {2,5,1,3} برابر است با:

$$90+85+66+50=291$$

سوال 4: -

سوال 5: پنج فایل مرتب با تعداد رکوردهای زیر موجود است. حداقل تعداد مقایسه ها برای ادغام این 5 فایل چقدر است؟

|   |   |   |   |    |
|---|---|---|---|----|
|   |   |   |   |    |
| 5 | 7 | 8 | 9 | 14 |

راه حل تشریحی:

گر تعداد فایل ها  $k$  باشد و تعداد کل عناصر را  $n$  در نظر بگیریم، برای محاسبه حداقل تعداد مقایسه از روش زیر استفاده می کنیم:

ابتدا بر اساس اولین عنصر هر فایل یک minheap می سازیم.  
 $O(k)$

سپس ریشه minheap (کوچکترین عنصر) را حذف و در خروجی ذخیره می کنیم  $O(n \log k)$

ریشه دوباره حذف و در خروجی می نویسیم عنصر دوم فایل دوم را درج می کنیم و ...

لذا به ازای هر عنصری که در minheap قرار می گیرد  $\lceil \log k \rceil$  مقایسه جهت بازسازی نیاز داریم پس در کل نیاز به  $O(n \log k)$  مقایسه خواهیم داشت.

|اتمام نیم سال|