

وحید علافر - ۹۱۷۸۳۹۰۲۲ - سوالات زوج ترم تابستان ۱۳۹۴

```
for (i=1; i <= n; i++) for (j= n-1; j <= n+i ;j++)
    S = S + 2;
```

$$n^2+n-3 \quad .^{\text{f}} \quad n^2+2n+2 \quad .^{\text{r}} \quad 2n^2+n+5 \quad .^{\text{r}} \quad n^2+3n-2 \quad .^{\text{f}}$$

```
int F(int m, int n){ if (m==1 || n==1)
```

```
return 1; else if (m==n)
```

```
return F(m-1, n-1) + 2; else
```

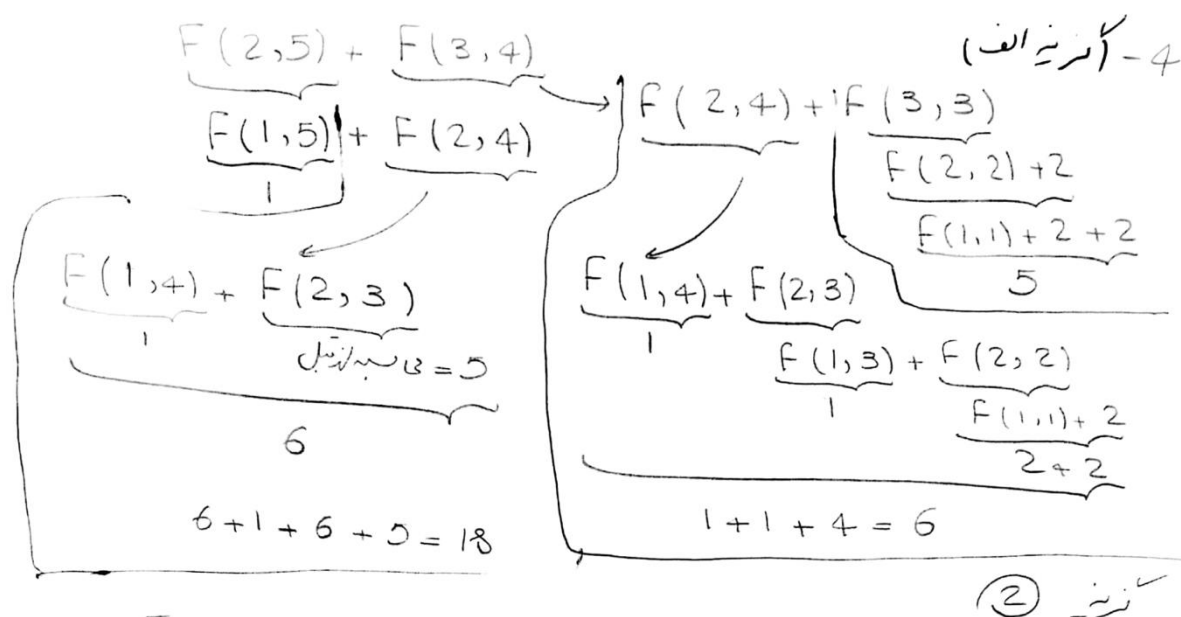
```
return F(m-1, n) + F(m, n-1);
```

14.4

۱۲.۳

18.2

١٤٠١



۸- در الگوریتم merge sort برای مرتب کردن یک آرایه n عنصری، تابع merge (ادغام) چند بار فراخوانی می شود؟

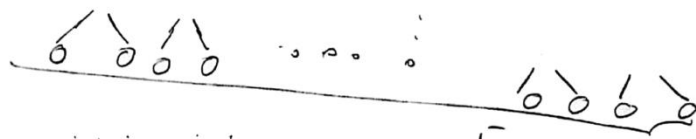
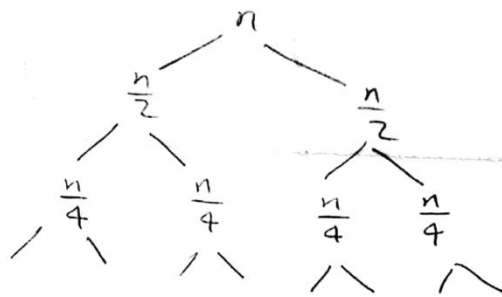
۴. $\log n$

۳. $\frac{n}{2}$

۲. n

۱. n-1

⑧ صحیح ترین گزینه ① است. اثبات کنیم:
 الگوریتم merge sort ابتدا کل آرایه را به دو بخش تقسیم می کند، پس حدود
 نیمی از آرایه در هر بخش به همین ترتیب، این فرآیند تا آنجا که به
 ترتیب یوار merge می کند.
 پس تعداد فراخوانی تابع merge برابر با تعداد فراخوانی تابع است که فراخوانی
 می کند. به عبارت دیگر، در آن حالت خطا می کند.
 بنابراین فرض کنیم یک آرایه n به ۲ بخش تقسیم می کنیم:



این n تا عنصر آرایه، ۲-۲، merge می شوند. بنابراین $\frac{n}{2}$ بار تابع
 merge فراخوانی می شود. حال این $\frac{n}{2}$ عنصر دوباره به ۲-۲، merge می شوند
 پس $\frac{n}{4}$ بار تابع merge فراخوانی می شود. به همین ترتیب ادامه پیدا خواهد
 کرد. merge تابع به همین روش، آرایه n به ۲-۲، sort شده و خواهد یافت.

$$\frac{n}{2} + \frac{n}{4} + \frac{n}{8} + \dots + 1 = n - 1$$

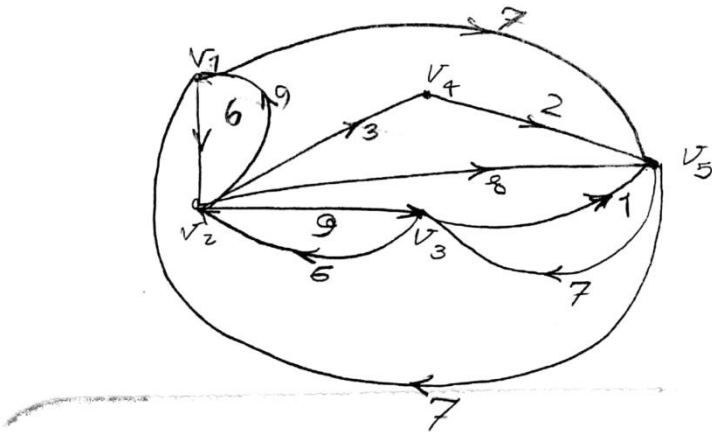
۱۰- کدام یک از عبارات های زیر در مورد الگوریتم کراسکال درست است؟

۱. برای پیاده سازی این الگوریتم از ساختار هرم Heap استفاده می شود.
۲. اگر یک گراف کامل n راسی داشته باشیم، زمان اجرای این الگوریتم از مرتبه $\theta(n \log n)$ خواهد بود.
۳. برای هر گراف درخت حاصل از این الگوریتم قطعاً با درخت حاصل از الگوریتم پریم یکسان خواهد بود.
۴. استفاده از این الگوریتم در گرافهای متراکم نسبت به الگوریتم پریم زمان اجرای بیشتری خواهد داشت.

۱- گزینه ۱

گزینه ۳، ۴ خط هستند، زیرا الگوریتم پریم در گراف های dense بهتر است
 گزینه ۲، الگوریتم کراسکال در گراف های sparse بهتر است.
 پس برای گراف زمان اجرای این دو الگوریتم یکسان نیست (گزینه ۳ اشتباه است).
 الگوریتم پریم نسبت به الگوریتم کراسکال در گراف های dense بهتر است (گزینه ۴ خط است).
 گزینه ۲ خط است چون complexity این الگوریتم در حالت گسترده
 $O(E \log V)$ است. E تعداد یال ها و V تعداد رئوس گراف
 است. برای گراف های کامل $n \geq 4$ را $E = \frac{n(n-1)}{2}$ ، تعداد یال ها از تعداد رئوس بیشتر
 خواهد بود.

- 12



Priority queue :

v_1
 $v_2 \quad v_5$
 $v_5 \quad v_4 \quad v_3$
 $v_4 \quad v_3$
 v_3
 \emptyset

distances: $v_2 \quad v_3 \quad v_4 \quad v_5$

∞	∞	∞	∞
6	∞	∞	7
6	15	9	7
<u>6</u>	14	9	<u>7</u>
<u>6</u>	13	<u>9</u>	<u>7</u>
<u>6</u>	<u>13</u>	<u>9</u>	<u>7</u>

جواب

۱۲- اگر ماتریس زیر نشان دهنده ماتریس مجاورت یک گراف جهت دار شامل پنج راس (V_1, \dots, V_5) باشد، پس از اجرای الگوریتم دیکسترا طول کوتاهترین مسیر از V_1 به V_3 چقدر است؟

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

۱۴ .۴

۱۸ .۳

۱۳ .۲

۱۵ .۱

۱۴ - جدول زیر اطلاعات مربوط به مهلت و سود هشت کار را نشان می دهد. شخصی در مواجهه با این کارها و بدون بررسی آنها

همه آنها را انتخاب می کند و ادعا می کند چنانچه نتواند کاری را حداکثر تا پایان مهلتش اجرا کند ۲ برابر سود آن کار جریمه پرداخت کند. حداقل جریمه ای که این شخص باید بپردازد چه خواهد بود؟

کار	A	B	C	D	E	F	G	H
مهلت	2	3	5	1	3	1	2	1
سود	20	35	12	8	28	30	5	50

۱۲۲ .۴

۹۰ .۳

۲۵۰ .۲

۱۲۶ .۱

۱۴ - سوال خطا است برای محبت کل در مسئله درجین کرده آیا در کار محبت کوک را استناد میکند یا خیر.

در مسئله ضریب دو جمله ای برای محاسبه $\binom{n}{k}$ با استفاده از راهبرد برنامه نویسی پویا، تعداد اعمال جمع برابر است با:

$$\frac{2n-k}{2} \quad \frac{k(2n+k+1)}{2} \quad \frac{n(2k-n-1)}{2} \quad \frac{k(2n-k-1)}{2}$$

$$C(n, k) = C(n-1, k-1) + C(n-1, k) \quad -16$$

$$C(n-2, k-2) + C(n-2, k-1)$$

و این روش به تدریج تا آنجا ادامه پیدا می کند که $k=1$.
 پس به وضوح تعداد جمع ها برابر است با

$$\frac{k(2n-k-1)}{2}$$
 (گزینه الف)

۱۸- در مسئله کوله پشتی صفر و یک با استفاده از راهبرد پویا، کدام رابطه صحیح است؟

$$p[i][w] = \begin{cases} \max(p[i][w], p_i + p[i][w-w_i]) & w_i \leq w \\ p[i][w] & w_i > w \end{cases} \quad ۱$$

$$p[i][w] = \begin{cases} \max(p[i-1][w], p_i + p[i-1][w-w_i]) & w_i \leq w \\ p[i-1][w] & w_i > w \end{cases} \quad ۲$$

$$p[i][w] = \begin{cases} \max(p[i+1][w], p_i + p[i+1][w-w_i]) & w_i \leq w \\ p[i+1][w] & w_i > w \end{cases} \quad ۳$$

$$p[i][w] = \begin{cases} \max(p[i-1][w], p_i + p[i-1][w-w_i]) & w_i \leq w \\ p[i-1][w] & w_i > w \end{cases} \quad ۴$$

۲۰- چنانچه سه کلید متمایز $key_1 < key_2 < key_3$ هر يك با احتمال جستجوی $P_1 = \frac{2}{6}$ ، $P_2 = \frac{1}{6}$ و $P_3 = \frac{3}{6}$

داشته باشیم، حداقل میانگین زمان جستجو در درخت جستجوی دودویی بهینه کدام است؟

۴. $\frac{10}{6}$

۳. $\frac{12}{6}$

۲. $\frac{9}{6}$

۱. $\frac{11}{6}$

۲۲- برای حل مسئله رنگ آمیزی گراف با استفاده از راهبرد عقبگرد، تعداد کل گره های درخت فضای حالت برای یک گراف n

راسی با عدد رنگی m کدام است؟

۴. $\frac{m^{n-1} - 1}{m-1}$

۳. $\frac{m^n + 1}{m-1}$

۲. $\frac{m^{n+1} - 1}{m-1}$

۱. $\frac{m^{n-1} + 1}{m+1}$

۲۴- الگوی جستجو در درخت فضای حالت برای روش بازگشت به عقب و روش انشعاب و تحدید به ترتیب از راست به چپ به چه صورت است؟

۱. جستجوی ردیفی - جستجوی عمقی

۲. در هر دو حات جستجوی عمقی

۳. در هر دو حات جستجوی ردیفی

۴. جستجوی عمقی - جستجوی ردیفی

24- روش حالت اول - جستجوی عمقی و روش انشعاب و تحدید
 روش دوم - جستجوی عمقی و روش انشعاب و تحدید
 است.
 (کدام است؟)

۲ - الگوریتم **Quick Sort** برای مرتب سازی آرایه ها را نوشته و پیچیدگی زمانی آن را در بدترین حالت تحلیل نمایید.
(به همراه تابع **Partition**)

الف) باید الگوریتم دلتاه $Pivot$ را انتخاب می کنیم .
 ب) $Pivot$ را با عنصر آرایه جابجایی می کنیم .
 ج) اعداد راست و چپ $Pivot$ را با هم مقایسه می کنیم و در صورت کوچکتر بودن عدد سمت راست ، آن را با عدد چپ تعویض می کنیم . تا جایی که تمام اعداد سمت راست از $Pivot$ بزرگتر و اعداد سمت چپ از $Pivot$ کوچکتر باشند . حال مراحل را برای دو آرایه راست و چپ تکرار می کنیم .
 complexity : $O(n \log n)$
 worst case : $O(n^2)$

۴ - اگر ماتریس زیر، ماتریس مجاورت یک گراف جهت دار دارای چهار راس باشد، با اجرای الگوریتم فروشنده دوره گرد در راهبرد پویا، طول تور بهینه را بدست آورید. (عملیات را مرحله به مرحله نشان دهید.)

0	10	15	20
5	0	9	10
6	13	0	12
8	8	9	0