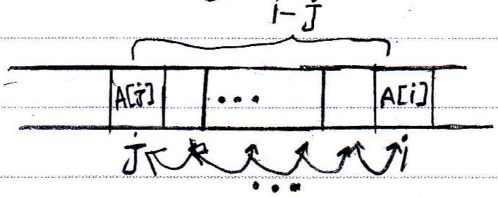
س۱: الف) وقتی یک نابه جایی داشته باشیم یعنی یکی از خانه های آرایه در جای درستش نمی باشد. پس تا زمانی که در جای درستش قرار بگیرد نابه جایی داریم و در insertion تعداد جابه جایی ها با تعداد نابه جایی ها برابر است. پس تعداد نابه جایی ها با تعداد اجرای حلقه while رابطه مستقیم دارد و هرچه نابه جایی بیشتر شود تکرار دستورات داخل while نیز بیشتر می شود و زمان اجرا بالا می رود.

ب) با توجه به قسمت قبل فرض کنیم آرایه ی مرتب شده فقط یک نابه جایی دارد پس برای مرتب شدن باید A[j] تک تک با خانه بعدی جابه جا شود تا به خانه A[i] برسد و این مقدار جابه جایی برابر i-j است که در واقع i-j تا نابه جایی هم رخ می دهد تا به جای درست خود برسد.



س۲:

1. x=0;	$C_1 \times (1)$
2. for(i=1;i<=n;i++)	$C_2(n+1)$
3. for(j=1;j<=n;j++) x++;	$C_3(n)(n+1)+C_4(n)(n)$
4. j=1;	$C_5(n)(1)$
5. while(j<n){	$C_6(n)(\lceil\log_2 n\rceil+1)$
6. x++; j=j*2;	$C_7(n)(\lceil\log_2 n\rceil)(1)+C_8(n)(\lceil\log_2 n\rceil)(1)$
7. }	$C_9 \times 0$
8. }	$C_{10} \times 0$

$$T(n)=C_1+C_2(n+1)+C_3(n^2+n)+C_4(n^2)+C_5(n)+C_6(n\lceil\log_2 n\rceil+n)+C_7(n\lceil\log_2 n\rceil)+C_8(n\lceil\log_2 n\rceil)=(C_3+C_4)n^2+(C_6+C_7+C_8)n\lceil\log_2 n\rceil+(C_2+C_3+C_5+C_6)n+(C_1+C_2)=A n^2+A_1 n\lceil\log_2 n\rceil+A_2 n+A_3=O(n^2)$$

س۳:

```
for(i=1; i<=n; i++)
    if(A[i]==a)
        return i;
```

بهترین حالت a در خانه اول است. پس حلقه 1 بار اجرا می‌شود. $T(n)=O(1)$

بدترین حالت a در خانه آخر است. پس حلقه n بار اجرا می‌شود. $T(n)=O(n)$

حالت متوسط: $x+3x+6x=1 \Rightarrow x=\frac{1}{10}$

$$E(n)=\sum_{R_x} xP(X=x)=\sum_{i=1}^{\frac{n}{3}} i\times\frac{1}{(\frac{n}{3})}\times\frac{1}{10}+\sum_{i=\frac{n}{3}+1}^{\frac{2n}{3}} i\times\frac{1}{(\frac{n}{3})}\times\frac{3}{10}+\sum_{i=\frac{2n}{3}+1}^{n} i\times\frac{1}{(\frac{n}{3})}\times\frac{6}{10}$$

$$\frac{3}{n}\left(\frac{1}{10}\sum_{i=1}^{\frac{n}{3}} i+\frac{3}{10}\sum_{i=\frac{n}{3}+1}^{\frac{2n}{3}} i+\frac{6}{10}\sum_{i=\frac{2n}{3}+1}^{n} i\right)=$$

$$\frac{3}{n}\left(\frac{1}{10}\frac{(\frac{n}{3}+1)(\frac{n}{3})}{2}+\frac{3}{10}\frac{(n+1)(\frac{n}{3})}{2}+\frac{6}{10}\frac{(\frac{5n}{3}+1)(\frac{n}{3})}{2}\right)=$$

$$\left(\frac{1}{20}(\frac{n}{3}+1)+\frac{3}{20}(n+1)+\frac{6}{20}(\frac{5n}{3}+1)\right)=$$

$$\frac{1}{20}\left(\frac{n}{3}+1+3n+3+\frac{30n}{3}+6\right)=\frac{1}{20}\left(10+\frac{40n}{3}\right)=\frac{3+4n}{6}$$ بار اجرا می‌شود.

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

س۴: برای آرایه‌ی نامرتب $A[1\ldots n]$:

```
for(i=1; i<=n; i++){                  c1(n)
    for(j=1; j<=i-1; j++){            c2 Σ(i=1..n) tj
        if(A[i]+A[j]==K){
            return 1;
        }
    }
}return 0;
```

توضیح: در این الگوریتم زوج‌های مرتب تک به تک بررسی می‌شود یک حلقه عدد اول را پیمایش می‌کند و یک حلقه عدد دوم و چک کردن جمع آن دو. اگر هم پیدا نشد (return 0) حلقه‌ها تمام می‌شوند.

بدترین حالت

$$T(n)=c_1(n)+c_2\sum_{i=1}^{n}\overbrace{t_j}^{(i-1)}=c_1(n)+c_2\left(\frac{(n+1)(n)}{2}-n\right)$$

$$=\left(c_1-\frac{c_2}{2}\right)n+\left(\frac{c_2}{2}\right)n^2=An^2+Bn=O(n^2)$$

Date: .... Subject: ....

حالت دوم برای آرایه مرتب شده ی $A[1 \ldots n]$:

```
P1 = 1;                         C1(1)
P2 = n;                         C2(1)
While(P1 < P2){                 C3 tm
    if(A[P1]+A[P2] == K){
        return 1;
    }
    if(A[P1]+A[P2] > K){
        P2 --;
    }
    else{
        P1++;
    }
}
return 0;
```

توضیح: در این الگوریتم آرایه مرتب شده را می گیریم و دو اشاره گر را تعریف می کنیم که اول ابتدای بازه و آخری انتهای بازه می باشد. در اول کار ابتدای بازه برابر اولین خانه آرایه و در انتها برابر آخرین خانه است. سپس درون حلقه تا موقعی که ابتدا از انتها کوچک تر است عملیات جست وجو را انجام می دهیم. اگر مقدار خانه ابتدا به علاوه مقدار خانه انتها برابر K شود که دو عدد مقادیر این دو خانه هستند. (return 1) و کار تمام می شود. اگر جمع مقادیر این دو خانه از K عبور کرد انتهای بازه را یکی به عقب میاریم و اگر از مقدار K کمتر بود ابتدای بازه را یکی به جلو می بریم. و دوباره این عمل تکرار می شود تا وقتی که به K برسیم. اگر هم که K پیدا نشد، ابتدا با انتهای بازه برابر می شود و حلقه تمام می شود. (return 0).

تعداد اجرای دستورات داخل حلقه حداکثر $(n-1)$ است و حداقل 1 بار.

$$T(n) = C_1(1) + C_2(1) + C_3 t_m$$

t_m را بدترین حالت می گیریم:

$$T(n) = (C_1 + C_2) + C_3(n-1) = An + B = O(n)$$

پس در حالت مرتب شده زمان اجرا کم تر از حالت مرتب نشده است.

سوال ۵:

a)

$\log(n!) = \log(n(n-1)(n-2)(n-3)\dots(2)(1)) = \log n + \log(n-1) + \log(n-2) + \dots$
$+ \log 2 + \log 1 \leq \underbrace{\log n + \log n + \dots + \log n + \log n}_{\text{تا } n} = \log(\underbrace{n \times n \times n \times \dots \times n}_{\text{تا } n})$
$= \log(n^n)$

$\Rightarrow \exists c, n_0 > 0 \mid \forall n \geq n_0 \quad \log(n!) \leq c \log(n^n) \iff \lg(n!) = O(\lg(n^n))$ ①
$c = 1$

$\log(n!) = \log(n(n-1)(n-2)\dots(2)(1)) = \log n + \log(n-1) + \log(n-2) + \dots + \lg 1$
$\geq \underbrace{\log n + \log(n-1) + \dots + \log(\frac{n}{2})}_{\text{تا } \frac{n}{2}} \geq \underbrace{\log \frac{n}{2} + \log \frac{n}{2} + \dots + \log \frac{n}{2}}_{\text{تا } \frac{n}{2}} = \frac{n}{2} \log(\frac{n}{2})$ (I)

$\forall n \geq 4 \quad \log (\frac{n}{2})^{2n} \geq \log n^n \Rightarrow 2n \log \frac{n}{2} > \log n^{n \times \frac{1}{4}} \Rightarrow \frac{n}{2} \log \frac{n}{2} > \frac{1}{4} \log n^n$ (II)

$\xrightarrow{\text{(I)(II)}} \log(n!) > \frac{1}{4} \log n^n \Rightarrow \exists c, n_0 > 0 \; \forall n > n_0 \; \log(n!) \geq c \log(n^n) \Rightarrow \log(n!) = \Omega(\log n^n)$
$c = \frac{1}{4}$ ②

$\xrightarrow{①, ②} \lg(n!) = \Theta(\lg n^n)$

b) $\frac{1}{\log_2 n} = \log_n 2 \xrightarrow{\text{به توان}} n^{\frac{1}{\log n}} = n^{\log_n 2} = 2 \Rightarrow n^{\frac{1}{\log n}} = 2$ ①

$\exists c_1, c_2, n_0 > 0 \mid \forall n \geq n_0 \quad c_1 \times 2 \leq 1 \leq c_2 \times 2 \iff 2 = \Theta(1)$ ②
$\Rightarrow c_1 = \frac{1}{2}, \; c_2 = 1$

$\xrightarrow{①, ②} n^{\frac{1}{\log n}} = \Theta(1)$

c) $\lim_{n \to \infty} \frac{n!}{2^n} = \lim_{n \to \infty} \frac{(\frac{n}{e})^n \sqrt{2\pi n}(1 + \Theta(\frac{1}{n}))}{2^n} = \lim_{n \to \infty} \frac{n^n}{2^n} = \infty \Rightarrow n! = \omega(2^n)$

d) $\lim_{n \to \infty} \frac{n!}{n^n} = \lim_{n \to \infty} \frac{(\frac{n}{e})^n \sqrt{2\pi n}(1 + \Theta(\frac{1}{n}))}{n^n} = \lim_{n \to \infty} \frac{n^n e^{-n}}{n^n} = \lim_{n \to \infty} e^{-n} = 0$

$\Rightarrow n! = o(n^n)$

س ۶:

a) غلط: $f(n)=n \quad g(n)=n^2$

$$n = O(n^2) \Longrightarrow n^2 = O(n) \ \times$$

b) غلط: $f(n)=n \quad g(n)=n^2$

$$n+n^2 = \Theta(n^2) = \Theta(\min\{n, n^2\}) = \Theta(n)$$

$$\Longrightarrow \Theta(n^2) = \Theta(n) \ \times$$

c) درست: $f(n) \in O(g(n)) \Longleftrightarrow \exists c, n_0 > 0 \mid \forall n \geq n_0 \ f(n) \leq c\,g(n)$ ①

طبق فرض: $1 \leq f(n) \xRightarrow{①} 1 \leq f(n) \leq c\,g(n)$

از دو طرف log $\Longrightarrow \log_2 1 \leq \log_2(f(n)) \leq \log_2(c\,g(n))$

$$0 \leq \lg(f(n)) \leq \lg(c\,g(n)) = \lg c + \lg(g(n))$$

$$\Longrightarrow = \lg g(n)\left(\frac{\lg c}{\lg g(n)} + 1\right) \leq \lg g(n)\underbrace{(\lg c + 1)}_{c'} = c' \lg(g(n))$$

$$\Longrightarrow 0 \leq f(n) \leq c' \lg g(n) \Longleftrightarrow \lg f(n) \in O(\lg(f(n))$$

$$\Longrightarrow \lg(f(n)) = O(\lg(f(n)))$$

d) غلط: $f(n) = 2n \quad g(n) = n$

$$\left.\begin{aligned} 2^{f(n)} &= 2^{2n} = 4^n \\ 2^{g(n)} &= 2^n \end{aligned}\right\} \Longrightarrow 4^n = O(2^n)$$

$\times$

e) غلط: $f(n) = \frac{1}{n}, \quad f(n) = \frac{1}{n^2} \qquad \frac{1}{n} = O\left(\frac{1}{n^2}\right)$

$\times$

f) درست: $f(n) \in O(g(n)) \Longleftrightarrow \exists c, n_0 > 0 \mid \forall n \geq n_0 \ f(n) \leq c\,g(n)$

$$\Longrightarrow \underbrace{\frac{1}{c}}_{c'} f(n) \leq g(n) \Longrightarrow c' f(n) \leq g(n)$$

$$\Longrightarrow g(n) \geq c' f(n) \Longleftrightarrow g(n) \in \Omega(f(n))$$

$$\Longrightarrow g(n) = \Omega(f(n)) \ \checkmark$$

Date: Subject:

g) غلط: $f(n)=2^n \Rightarrow f(\frac{n}{2})=2^{\frac{n}{2}}$

$$2^n \underset{\times}{=} \Theta(2^{\frac{n}{2}}) \Longleftrightarrow \begin{cases} 2^n=\Omega(2^{\frac{n}{2}}) \checkmark \\ 2^n=O(2^{\frac{n}{2}}) \times \end{cases}$$

h) درست: $g(n)=O(f(n))$

$$f(n)+g(n)=\Theta(f(n)) \Longleftrightarrow \exists c_1,c_2,n_0>0 \mid \forall n>n_0$$
$$c_1 f(n) \le f(n)+g(n) \le c_2 f(n)$$

برای اثبات عبارت بالا باید c_1 و c_2 ای پیدا شوند که شرایط صدق کنند.

$$0 \le g(n) \Rightarrow f(n) \le f(n)+g(n) \Rightarrow c_1=1$$

$$g(n)=O(f(n)) \Rightarrow g(n)\in O(f(n)) \Rightarrow \exists c,n_0'>0 \mid \forall n>n_0' \quad g(n)\le c f(n) \quad (1)$$
$$\Rightarrow f(n)\le f(n) \xrightarrow{+(1)} f(n)+g(n) \le 2f(n)$$
$$\Rightarrow c_2=2$$

$$\Rightarrow f(n)+g(n)=\Theta(n)$$
$$\Rightarrow f(n)+O(f(n))=\Theta(n) \checkmark$$

Date: .... Subject: ....

سوال ۷:

A	B	O	o	Ω	ω	Θ
$n^{۲}$	$n^{۳}$	Yes	yes	No	No	No
$\log^{k} n$	n^{ε}	yes	yes	No	No	No
n^{k}	c^{n}	yes	Yes	No	No	No
۲^{n}	$۲^{n/۲}$	No	No	Yes	yes	No
$n^{\log c}$	$c^{\log n}$	yes	No	Yes	No	Yes
$۴^{\log n}$	$n^{۲}$	yes	No	yes	No	yes
$n!$	$n۲^{n}$	No	No	yes	yes	No
$\sqrt{۲}^{\log n}$	$۲^{\sqrt{۲\log n}}$	No	No	yes	yes	No
$(\log n)!$	$۲^{۲n}$	Yes	Yes	No	No	No
$n^{\log(\log n)}$	$(\log n)^{\log n}$	Yes	No	Yes	No	Yes