אפקה המכללה האקדמית להנדסה בתל-אביב AFEKA
TEL-AVIV ACADEMIC COLLEGE OF ENGINEERING

הנדסת תוכנה

Software Engineering

<u>פתרון תרגיל 2 להגשה במבני נתונים (קורס מס' 10117)</u>

מרצה : **ד"ר ראובן חוטובלי**

<u>שאלה 1</u>

	סעיף א'	סעיף ב'	סעיף ג'	סעיף ד'	סעיף ה'	סעיף ו'	סעיף ז'	סעיף ח'	סעיף ט'	סעיף י'	סעיף י"א	י"ב	י"ג
<u>נכון</u>	X	X	X		X			X		X			X
<u>לא נכון</u>				X		X	X		X		X	X	

		תשובה			
שאלה		1	2	3	4
.2	א				X
	ב			X	
	ג			X	
	ד		X		
	ה		X		
	ו	X			
	ז	X			
	ח		X		
	ט			X	
	י			X	
	י"א				X
	י"ב		X		
	י"ג			X	
	י"ד				X
	ט"ו				X
	ט"ז	X			
	י"ז		X		
	י"ח			X	
	י"ט				X
	כ			X	
	כ"א		X		
	כ"ב			X	

AFEKA המכללה האקדמית להנדסה בתל-אביב TEL-AVIV ACADEMIC COLLEGE OF ENGINEERING אפקה

שאלה 3

א.

האם הטענה הבאה נכונה תמיד? אם כן ענה ב-"נכונה" אחרת "לא נכונה".

$$if\ \ g(n)+f(n)\in\Omega(t(n))\ \ and\ \ g(n)-f(n)\in O(t(n))\ \ then\ \ f(n)\in\Omega(t(n))$$

אם לדעתך הטענה נכונה אז הוכח אותה אחרת תנו דוגמא נגדית.

<u>הטענה איננה נכונה</u> כי אם נבחר:

$$f(n)=n\quad and\quad g(n)=n^2\quad and\quad t(n)=n^2$$

אז נקבל: $f(n)+g(n)=n^2+n\ge c_1 n^2\quad \forall n\ge n_1$

לכן מתקיים $f(n)+g(n)\in\Omega(t(n))$ כנדרש.

כמו כן, $g(n)-f(n)=n^2-n\le n^2\le c_2 n^2\quad \forall n\ge n_2$

לכן $g(n)-f(n)\in O(t(n))$ מתקיים כנדרש.

עתה השאלה היא האם $f(n)\in\Omega(t(n))$? כלומר האם $n\in\Omega(n^2)$? והתשובה היא לא.

כי $\lim \frac{n}{n^2}=0$ ולפי המשפט $n^2\in\Omega(n)\ \ and\ \ n\notin\Omega(n^2)$.

א. הטענה אינה נכונה לדוגמא $f(n)=3^{2n}$. כלומר אם הטענה הייתה נכונה אז יתקיים:

$$C_1 * 3^n \le 3^{2n} \le C_2 * 3^n$$

לאחר חילוק כל האגפים ב - 3^n

נקבל: $C_1 \le 3^n \le C_2$ ולא יכול להיות ש: 3^n יהיה חסום מלמעלה ע"י קבוע כלשהו.

מש"ל

<u>ג.</u>

$$2^{O(\lg\lg n)} = O(\lg n)$$

<u>לא נכון.</u>

הוכחה:

$2^{O(\lg\lg n)} \le 2^{c\lg\lg n}$ לפי הגדרה

$= 2^{\lg((\lg n)^c)}$ חוקי לוגים

$= (\lg n)^c$ $\quad 2^{\lg x}=x$

$\ne O(\lg n)$ $\quad \forall c>1$

דוגמה נגדית:

<u>שאלה 4:</u>

סדר עולה (לא יורד) משמאל לימין

$$2^{\lg n},\lg\left(n^2!\right),n^{10},\left\{2^{\lg^2 n},n^{\lg n}\right\},4^{\sqrt{n}},3^n,\left\{2^{2n},4^n\right\},\left(n+1\right)^n,n^{n+1},(4n)!$$

<u>פונקציות תחת סוגריים מסולסלים יש להן אותה התנהגות אסימטוטית</u>

.1 $2^{(\log n)^2} = 2^{\log n \times \log n} = n^{\log n} \to f_4 \in \Theta(f_5)$

טענת עזר :

$$\log(n^2!) = \log(1 \cdot 2 \cdot \ldots \cdot (n^2-1) \cdot n^2) = \log(1) + \log(2) + \cdots + \log(n^2-1) + \log(n^2)$$

$$\underbrace{\log(1) + \log(2) + \cdots + \log(n^2-1) + \log(n^2)}_{n^2\ elements,\ \log(n^2)\ being\ the\ largest} \le n^2 \cdot \log(n^2) = 2n^2\log(n) \Rightarrow \log(n^2!) \in O(n^2\log(n))$$

$$\underbrace{\log(1)}_{0} + \underbrace{\log(2) + \cdots + \log(n^2-1) + \log(n^2)}_{n^2-1\ elements,\ \log(2)\ being\ the\ smallest} \ge (n^2-1)\underbrace{\log(2)}_{1\ assuming\ base\ 2} = n^2 - 1 \Rightarrow \log(n^2!) \in \Omega(n^2)$$

<u>הוכחות עבור כל זוג עוקב :</u>

.1 $\lim_{n\to\infty}\frac{2^{\log n}}{\log((n^2)!)} = \lim_{n\to\infty}\frac{n}{\log((n^2)!)} \le \lim_{n\to\infty}\frac{n}{n^2-1} = 0$

.2 $\lim_{n\to\infty}\frac{\log\left((n^2)!\right)}{n^{10}} \le \lim_{n\to\infty}\frac{n^2\log n^2}{n^{10}} = \lim_{n\to\infty}\frac{2n^2\log n}{n^{10}} = \lim_{n\to\infty}\frac{\log n}{n^8} = 0$

.3 $\lim_{n\to\infty}\frac{n^{10}}{n^{\log n}} = 0\ , 10 < \log n$

.4 $\lim_{n\to\infty}\frac{n^{\log n}}{2^{(\log n)^2}} = \frac{n^{\log n}}{n^{\log n}} = 1$

.5 $\lim_{n\to\infty}\frac{2^{(\log n)^2}}{4^{\sqrt{n}}} = \lim_{n\to\infty}\frac{n^{\log n}}{4^{\sqrt{n}}} = \lim_{n\to\infty}\frac{e^{\ln n^{\log n}}}{e^{\ln 4^{\sqrt{n}}}} \le \lim_{n\to\infty}\frac{e^{\log n*(\ln n)}}{e^{\sqrt{n}\ln(4)}} = 0, \log n * \ln n <$
$\sqrt{n}\ln(4)$

.6 $\lim_{n\to\infty}\frac{4^{\sqrt{n}}}{3^n} < \lim_{n\to\infty}\frac{4^{\frac{n}{2}}}{3^n} = \lim_{n\to\infty}\frac{2^n}{3^n} = \lim_{n\to\infty}(\frac{2}{3})^n = 0$

.7 $\lim_{n\to\infty}\frac{3^n}{4^n} = \lim_{n\to\infty}(\frac{3}{4})^n = 0$

.8 $\lim_{n\to\infty}\frac{4^n}{2^{2n}} = \frac{4^n}{4^n} = 1$

.9 $let\ \ n_0 = 4\ \ and\ \ \forall n \ge n_0\ \ \ 4^n < (n+1)^n$

.10 $\lim_{n\to\infty}\frac{(n+1)^n}{n^{n+1}} = \lim_{n\to\infty}\frac{(n+1)^n}{n\cdot n^n} = \lim_{n\to\infty}\frac{1}{n}\cdot\left(1+\frac{1}{n}\right)^n = \lim_{n\to\infty} 0\cdot e = 0$

$\exists n_0:\ \forall n \ge n_0\ \ \ n^{n+1} > (n+1)^n$

11. נגדיר: $a_n = \frac{(4n)!}{n^{n+1}}$ לכן $a_{n+1} = \frac{(4n+4)!}{(n+1)^{n+2}}$

$$\lim_{n\to\infty} \frac{\frac{(4n+4)!}{(n+1)^{n+2}}}{\frac{(4n)!}{n^{n+1}}} = \lim_{n\to\infty} \frac{n^{n+1}\cdot(4n+4)!}{(4n)!\cdot(n+1)^{n+2}} = \lim_{n\to\infty} \frac{n\cdot n^n\cdot(4n+4)!}{(4n)!\cdot(n+1)^n\cdot(n+1)^2} =$$

$$= \lim_{n\to\infty} \frac{n\cdot n^n\cdot(4n)!\cdot(4n+1)(4n+2)(4n+3)(4n+4)}{(4n)!\cdot(n+1)^n(n+1)^2} =$$

$$\lim_{n\to\infty} \frac{n^n}{(n+1)^n}\cdot\frac{n(4n+1)(4n+2)(4n+3)(4jn+4)}{(n+1)^2}$$

לכן אם נגדיר $b_n = \frac{n^{n+1}}{(4n)!}$ אז בהמשך למה שקיבלנו עד כה נקבל:

$$\lim_{n\to\infty} \frac{(n+1)^n}{n^n}\cdot\frac{(n+1)^2}{n(4n+1)(4n+2)(4n+3)(4jn+4)} =$$

$$= \lim_{n\to\infty}\left(1+\frac{1}{n}\right)^n\cdot\frac{(n+1)^2}{n(4n+1)(4n+2)(4n+3)(4jn+4)} = e\cdot 0 = 0$$

$$\Rightarrow (4n)! > n^{n+1}$$

שאלה 5:

סעיף א

עץ רקורסיה/הצבה/אינדוקציה $\Theta(n)$ – (שימו לב שהסכום $\sum_i (a+b)^i$ חסום עבור a+b<1).

שבונים עץ רקורסיה רואים:

ברמה 0 סכום האיברים החופשיים הוא: $(\alpha+\beta)^0\cdot n$

ברמה 1 סכום האיברים החופשיים הוא: $(\alpha+\beta)^1\cdot n$

ברמה 2 סכום האיברים החופשיים הוא: $(\alpha+\beta)^2\cdot n$

ברמה 3 סכום האיברים החופשיים הוא: $(\alpha+\beta)^3\cdot n$

וכך ממשיכים.

לכן, ברמה i סכום האיברים החופשיים הוא: $(\alpha+\beta)^i\cdot n$

אך מס' הרמות שתפתחנה כמו שראינו בהרצאה הוא מסדר גודל logn .

לכן נקבל: $T(n) = \sum_{i=0}^{\log n}(\alpha+\beta)^i\cdot n = n\cdot\sum_{i=0}^{\log n}(\alpha+\beta)^i$

ובעבור n-יים גדולים נקבל $T(n) = n\cdot\sum_{i=0}^{\infty}(\alpha+\beta)^i = n\cdot c$ כי עבור $\alpha+\beta<1$

הטור $\sum_{i=0}^{\infty}(\alpha+\beta)^i$ מתכנס (סדרה הנדסית אינסופית יורדת).

לכן $T(n)=O(n)$

<u>סעיף ב –</u>

$T(n) = T(n/2+\sqrt{n}) + \sqrt{6044}$.

פתרון. קל לראות כי $\sqrt{n} < n/2$ כמעט לכל n, ולכן, נסתכל על הנוסחא כאילו היא: $T(n) = T(n/2) + \sqrt{6044}$. ראינו בכיתה שנוסחא כזו הולכת ל־ $\Theta(\log n)$. כעת נוכיח שנוסחא המקורית היא אכן כזו.

טענה 1. קיימים שני קבועים, $c, n_0 > 0$ כך שלכל $n > n_0$ מתקיים:

$$T(n) \le c\cdot\log n$$

הוכחה: ההוכחה באינדוקציה על n. נתחיל עם הצעד, נניח שעבור $n/2+\sqrt{n}$ מתקיים:

$$T(n/2+\sqrt{n}) \le c\cdot\log(n/2+\sqrt{n})$$

ולכן:

$$T(n) = T(n/2+\sqrt{n}) + \sqrt{6044} \le c\cdot\log(n/2+\sqrt{n}) + \sqrt{6044}$$

החל מ $n_0 = 16$, נקבל כי: $n/2+\sqrt{n} \le n/2+n/4 = 3n/4$, ולכן נקבל:

$$T(n) \le c\cdot\log(n/2+\sqrt{n}) + \sqrt{6044} \le c\cdot\log(3n/4) + \sqrt{6044}$$

אנו שואלים עבור איזה c הנ"ל קטן מ־ $c\log n$. נקבל:

$$\begin{aligned} c\cdot\log(3n/4) + \sqrt{6044} &\le c\log n \\ \sqrt{6044} &\le c\cdot(\log n - \log(3n/4)) = c\cdot(\log 4/3) \\ \frac{\sqrt{6044}}{\log(4/3)} &\le c \end{aligned}$$

ונקבל כי עבור c כנ"ל, ועבור $n \ge 16$, צעד האינדוקציה מתקיים.

בסיס: השלם לבד. $\blacksquare$

נהנ"ל אנו מסיקים כי $T(n) \in O(\log n)$. כעת נראה חסם תחתון.

טענה 2. קיימים קבועים $c, n_0 > 0$ כך שלכל $n \ge n_0$ מתקיים:

$$T(n) \ge c\cdot\log n$$

הוכחה: צעד האינדוקציה: נניח נכונות עבור $n/2+\sqrt{n}$ שמתקיים:

$$T(n/2+\sqrt{n}) \ge c\cdot\log(n/2+\sqrt{n})$$

נקבל:

$$T(n)=T(n/2+\sqrt{n})+\sqrt{6044}\ge c\cdot\log(n/2+\sqrt{n})+\sqrt{6044}\ge c\cdot\log(n/2)+\sqrt{6044}$$

אנו אושלים מתי הנ"ל גדול מ - $c\cdot\log n$. נקבל:

$$\begin{aligned} c\cdot\log(n/2)+\sqrt{6044} &\ge c\cdot\log n \\ \sqrt{6044} &\ge c\cdot\log 2 \\ \sqrt{6044} &\ge c \end{aligned}$$

וקיבלנו שצעד האינדוקציה נכון לכל c קטן מספיק.

בסיס - השלם לבד. ■

<u>סעיף ג –</u>

$$T(n)=T(\log n)+1$$

נציב את הנוסחא בעצמה פעם אחת ונקבל:

$$T(n)=T(\log^{(2)} n)+2$$

נציב פעם נוספת ונקבל:

$$T(n)=T(\log^{(3)} n)+3$$

ובאופן כללי:

$$T(n)=T(\log^{(i)} n)+i$$

את הנ"ל יש להוכיח באינדוקציה. מתי נגיע ל - $T(1)$? כלומר, מתי:

$$\log^{(i)} n=1$$

הנ"ל קורה כאשר $i=\log^* n$. כאשר $i=\log^* n$, נקבל:

$$T(n)=1+\log^* n$$

ולכן הנוסחא היא ב - $\Theta(\log^* n)$.

<u>סעיף ד –</u>

$$T(n)=\frac{1}{n}+\sum_{i=1}^{n-1}[T(i)-T(i-1)]\ ; \qquad T(0)=0,\quad T(1)=1$$

נפתח את הסכום לעיל, נתחשב בתנאי ההתחלה T(0)=0, ונקבל :

$$\begin{aligned} T(n)=\frac{1}{n}&+T(1)-T(0)+T(2)-T(1)+T(3)-T(2)+\cdots+T(n-2) \\ &-T(n-3)+T(n-1)-T(n-2)=\frac{1}{n}-T(0)+T(n-1) \\ &=\frac{1}{n}+T(n-1) \end{aligned}$$

בסה"כ קיבלנו : $T(n) = T(n-1) + \frac{1}{n}$

כעת נבצע לנוסחא זו n-1 איטרציות ונקבל :

$$T(n) = T(n-1) + \frac{1}{n} = T(n-2) + \frac{1}{n-1} + \frac{1}{n} = T(n-3) + \frac{1}{n-2} + \frac{1}{n-1} + \frac{1}{n}$$
$$= \cdots = T(1) + \sum_{i=n}^{1} \frac{1}{i} = 1 + \sum_{i=1}^{n} \frac{1}{i} = 1 + \emptyset(\ln(n)) + O(1) = \emptyset(logn)$$

כלומר, קיבלנו כי סדר הגודל של T(n) הוא $\emptyset(logn)$.

<u>שאלה 6:</u>

א. נתאר אלגוריתם בשם SimpleCheck המקבל כפרמטרים מערך A ושני מספרים טבעיים r ו q,
באשר r<q כדלקמן:

```
SimpleCheck(A,r,q)
{
    return (A[q] – A[r] > q-r)
}
```

האלגוריתם המבוקש בשאלה אם כן הינו SimpleCheck(A,1,n).

ב.

נתאר אלגוריתם רקורסיבי בשם SolveB המקבל כפרמטרים מערך A ושני מספרים טבעיים r ו q,
באשר r<q כדלקמן:

```
SolveB(A,r,q)
{
    If (r+1 = q) return A[r]+1;
    m = (r+q) div 2;
    if (SimpleCheck(A,r,m) == true)
        return SolveB(A,r,m);
    else
        return SolveB(A,m,q);
}
```

האלגוריתם המבוקש בשאלה אם כן הינו

```
SolB(A,1,n)
{
    if (SimpleCheck(A,1,n) == false)
        Print "There is no such element in A";
    else
        return SolveB(A,1,n);
}
```

ג. נתאר אלגוריתם רקורסיבי בשם SolveC המקבל כפרמטרים מערך A ושני מספרים טבעיים r ו q,
באשר r<q כדלקמן:

```
SolveC(A,r,q)
{
```

```
   if (q < r) return false;
   m = (q + r) div 2;
   if (A[m] = m) return true;
   if (A[m] > m) return SolveC(A,r,m-1)
   else return SolveC(A,m+1,q);
}
```

האלגוריתם המבוקש בשאלה אם כן הינו SolveC(A,1,n).

<u>שאלה 7</u>

A is a Boolean (1 or 0 values) matrix of size n*n.

Find a data structure that supports the following operations in the given time:

	<u>Operation</u>	<u>Time</u>
init(n)	Initialize A with the value 1	$O(n^2)$
flip(i,j)	A[i,j]=!A[i,j]	O(1)
hasRowOf1	Return true iff A has a row that contains only <u>1-s</u>	O(1)
hasRowOf0	Return true iff A has a row that contains only <u>0-s</u>	O(1)

Solution:

We will use the uses the following data structures:

- A[n][n] – the matrix
- Sum[n] – An array containing sums of the rows in A. Sum[i] = the sum of the row i in A.
- count1 = how many cells in Sum contains n
- count0 = how many cells in Sum contains 0

<u>flip(i,j)</u>

```
    if (A[i][j]==0)
    {
            if (Sum[i]=0)   count0- - ;
            Sum[i]++ ;
            if (Sum[i]=n)   count1++ ;
            A[i][j]=1;
    }
    else {
            if (Sum[i]=n) count1- - ;
            Sum[i]--   ;
            if (Sum[i]=0)  count0++;
            A[i][j]=0;
        }
```

```
    hasRowOf1()
            return count1>0;

    hasRowOf0()
            return count0>0;
```

<u>init()</u>

```
    fill A with 1's
    fill Sum with n.
    count1=n
    count0=0
```

<u>שאלה 8</u>

	1	**2**	**3**	**4**
a				X
b				X
c			X	
d			X	
e		X		
f		X		