

פתרון מוצע לבחינת מה"ט באלגוריתמיקה ותכנות ב JAVA

מועד א' תשפ"א, מרץ 2021
מחבר: ד"ר לביא אודי, מכללת אורט רחובות

חלק א'

שאלה מס. 1

```
public static Scanner in = new Scanner(System.in);
public static void main(String[] args)
{
    int num; // מספר נקלט
    int cntEven=0; // כמות מספרים זוגיים
    int sumOdd=0; // סכום מספרים אי זוגיים
    for(int i=0; i<40; i++){
        num = in.nextInt();
        if (num>=100 && num<=999){
            System.out.println("סכום ספרות: "
                               +(num/100+num%100/10+num%10));
        }
        if (num%2==0){
            cntEven++;
        }
        else{
            sumOdd+=num;
        }
    }
    System.out.println("כמות מספרים זוגיים: "+cntEven);
    System.out.println("סכום מספרים אי זוגיים: "+sumOdd);
}
```

שאלה מס. 2

א. על פי הגדרת נתוני השאלה מוגדרת המחלקה Box:

```
public class Box{
    private String color;
    private int length;
    private int width;
    private int height;
```

```
public Box(String color){  
    this.color=color;  
    this.length=10;  
    this.width =10;  
    this.height =10;  
}  
...  
}
```

ב. הפעולה smallBox:

```
public Static String smallBox(Box[]arr){  
    int minIndex=0;  
    int volume=Box[0].getLength()*Box[0].getWidth()*Box[0].getHeight();  
    for(int i=1; i<Box.length; i++){  
        int calc= Box[i].getLength()*Box[i].getWidth()*Box[i].getHeight();  
        int max=Math.max(volume, calc);  
        if(max!=volume){  
            volume=calc;  
            minIndex=i;  
        }  
    }  
    return Box[minIndex].getColor();  
}
```

שאלה מס. 3

א. על פי הגדרת נתוני השאלה מוגדרת המחלקה Tank:

```
public class Tank{  
    int capacity;  
    int amount;  
    public Tank(int capacity){  
        this.capacity=capacity;  
        this.amount=0;  
    }  
}
```

ב.

```
public boolean hasAmount(){  
    return this.amount>0;  
}
```

ג.

```
public boolean isPossible(int num, char op){  
    boolean flag=false;  
    if (op=='+'){  
        if(this.amount+num<=this.capacity)  
        {  
            this.amount+=num;  
            flag=true;  
        }  
    }  
    else{  
        if(this.amount>=num){  
            this.amount-=num;  
            flag=true;  
        }  
    }  
    return flag;  
}
```

ד.

```
public static int fill(Tank t1, Tank t2){
    if(t2.isPossible(t1.getAmount(), '+')
        return t1.getAmount();
    else return t2.getCapacity()-t2.getAmount();
}
```

שאלה מס. 4

א.

```
public static Boolean isPerfect(int [] arr){
    boolean flag=false ;
    if (arr.length%2!=0 && arr[arr.length/2]==0)
    {
        flag=true;
        for(int i=0;i<arr.length/2&&flag; i++)
            if (!(arr[i]>=0 &&arr[i]<=9))
                flag=false;
        for(int i=arr.length/2+1; i<arr.length&&flag; i++)
            if (arr[i]>=0 &&arr[i]<=9)
                flag=false;
    }
    return flag;
}
```

ב.

סיבוכיות הפעולה במקרה הגרוע הינה $O(n)$. במקרה בו המערך מקיים את התנאים הנדרשים מתבצעת סריקה של כל חצי מערך בנפרד ולכן סך הכל $O(n)$.

שאלה מס. 5

א.

ערך מוחזר	if s>num	if s<num	s	p	k	num	arr
						10	2, 4, 7, 12, 18
					0		
				4			
			20				
		False					
	True						
				3			
			14				
		False					
	True						
				2			
			9				
		True					
					1		
			11				
		False					
	True						
				1			
false							

1. הפעולה מחזירה false

2. דוגמה ל num שיחזיר true: 11

ב. הפעולה בודקת האם ישנם שני ערכים במערך שסכומם שווה לערכו של num.

 ג. סיבוכיות הפעולה: $O(n/2)$ שזה למעשה $O(n)$. הפעולה רצה במקביל על המערך המתקבל ולכן זמן הריצה יהיה מחצית מגודלו.

שאלה מס. 6

```
public static boolean isStringOk(String str){
    boolean flag=true;
    if (str.charAt(0)=='.' || str.charAt(str.length()-
        1)=='.' || str.indexOf('.')==-1)
        flag=false;
    else{
        for(int i=str.indexOf('.');
            i<str.lastIndexOf('.')&&flag;i++){
            if
                (str.charAt(i)=='.'&&str.charAt(i+1)=='.')
                    flag=false;
        }
    }
    return flag;
}
```

חלק ב'

שאלה מס. 7

הגדרת המחלקה Battery:

```
public class Battery{
    private String manufacturer;
    private char size;
    private double volt;
    private boolean isNormal;
```

א.1

```
public Battery (String m, char s, double v, boolean isN){
    this.manufacturer = m;
    this.size = s;
    this.volt=v;
    this.isNormal = isN;
}
```

2.א.

```
public Battery(double v, char s){
    this.manufacturer = "Duracell";
    this.size = s;
    this.volt=v;
    this.isNormal = true;
}
```

ב.

הגדרת המחלקה:

```
public class Battery{
    double price;
    Battery [] batteries;
    ...
}
```

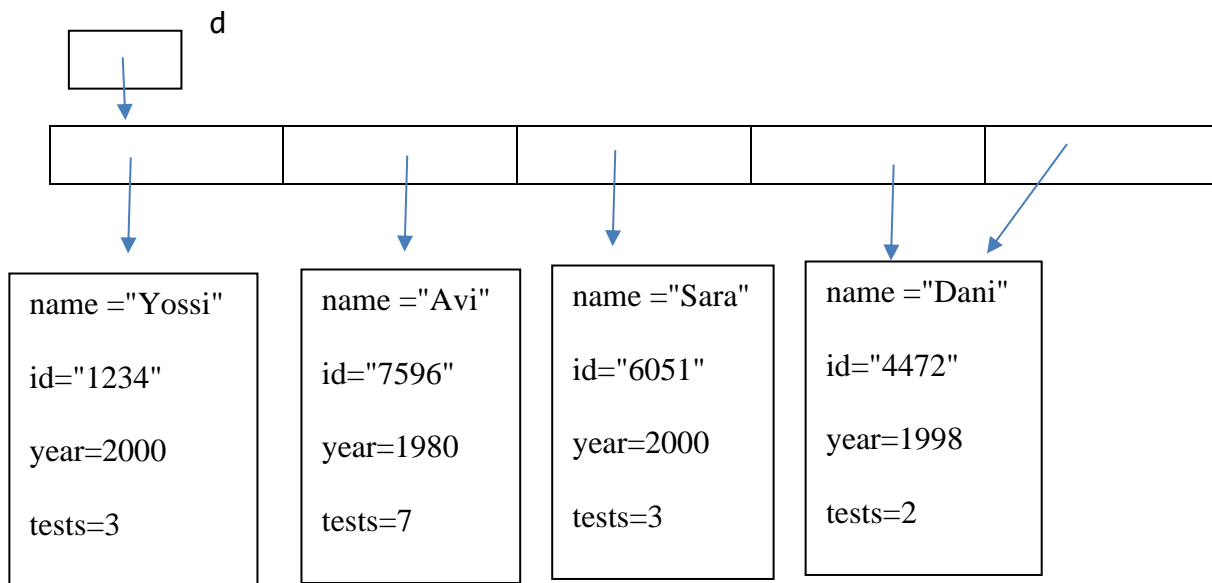
1.ב.

```
public static void checkPrice(BatteryPack b, double sum){
    double total=0;
    for(int i=0; i<b.getBatteries().length; i++)
    {
        if (b.getBatteries()[i].getIsNormal())
            sum+= b.getBatteries()[i].getVolt();
    }
    if (sum<v)
        return false;
    else return true;
}
```

שאלה מס. 8

א.8

```
public class Driver{
    private String name;
    private String id;
    private int year;
    private int tests;
    ...}
Driver d = new Driver[5];
```



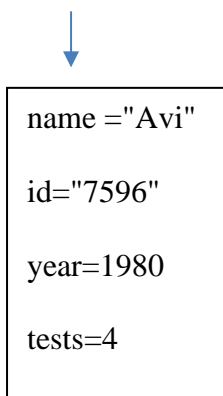
`m=0`

`s=0`

`x=d[1].getTests() =7`

`d[1].setTests(d[3].getTests() +2)`

`d[1]`



d[3].setTests(x)

d[3]



```
name="Dani"
id="4472"
year=1980
tests=7
```

פלט:

3

4

1

7

7

4.4

3

(s – סכום מבחני הנהיגה – 22, m מספר הנהגים שנולדו אחרי 1995 – 3)

8.ב

```
public static String [] func(Driver [] arr, int num){
    int cnt=0, j=0;
    for(int i=0; i<arr.length; i++){
        if (arr[i].getTests() < num)
            cnt++;
    }
    String [] ar = new String [cnt];
    for(int i=0; i<arr.length; i++){
        if (arr[i].getTests() < num)
            ar[j++]=arr[i].getName();
    }
    return ar;
}
```

שאלה מס. 9

9. א

```
public static boolean bigPolindrom(String str){
    int i=0;
    int j=str.length -1;
    while(i<j){
        char tav1=str.charAt(i);
        char tav2 = str.charAt(j);
        if (tav1 >='A' && tav1 <='Z' && tav2>='A' && tav2 <='Z'){
            if (tav1 !=tav2)
                return false;
            else {
                i++;
                j--;
            }
        }
        else{
            if (!(tav1 >='A' && tav1 <='Z'))
                i++;
            if(!(tav2>='A' && tav2 <='Z'))
                j--;
        }
    }
    return true;
}
```

9. ב

```
public static String funcB(String [] arr){
    String s = "";
    for(int i=0;i<arr.length; i++)
        if (bigPolindrom(arr[i]))
            if (arr[i].length >s.length)
                s=arr[i];
    return s;}
}
```

שאלה מס. 10**א.**

```
public static int funcA(int num){
    int cnt=0;
    while(num>0){
        if (num%2 ==0)
            cnt++;
        num= num/10;
    }
    return cnt;
}
```

ב.

```
public static int funcB(int []arr){
    int max=0;
    for(int i=0; i<arr.length; i++)
        max=Math.max(max, funcA(arr[i]));
    return max;
}
```

ג.

```
public static boolean funcC(int []arr){
    int [] monim = new int [funcB(arr)];
    for(int i=0;i<arr.length; i++)
        monim[funcA(arr[i])]++;
    for(int i=0;i<monim.length; i++)
        if (monim[i]!=1)
            return false;
    return true;
}
```

ד.

סיבוכיות זמן הריצה של הפעולות:

סעיף א – בהנחה שאורך המספר המתקבל הוא באורך n , זמן הריצה יהיה $O(n)$. פרוק המספר לספרות.

סעיף ב – בהנחה שאורך המערך המתקבל הוא m , ובעבור כל איבר מהמערך מופעלת הפעולה מסעיף א ולכן יהא זמן הריצה $O(n*m)$.

סעיף ג – הפעולה מפעילה את הפעולה מסעיף ב ($O(n*m)$) וכן מפעילה את הפעולה מסעיף א בעבור כל אחד מאיברי מערך $monim$. בהנחה שאורך מערך $monim$ הינו k , יהא זמן ריצת הלולאה $O(n*k)$.

סה"כ זמן הריצה: $O(n*m) + O(n*k) = O(\max(n*m, n*k))$.

שאלה מס. 11

א. 1 למערך a נשלח מערך b: 10,5,3,16,20,30

ערך מוחזר	If start<stop	stop	start	זימון
$a[0]=a[0]+a[1]=74+10=84$	true	5	0	one(b, 0,5)
$a[1]=a[1]+a[2]=69+5=74$	true	5	1	one(b, 1,5)
$a[2]=a[2]+a[3]=66+3=69$	true	5	2	one(b, 2,5)
$a[3]=a[3]+a[4]=50+16=66$	true	5	3	one(b, 3,5)
$a[4]=a[4]+a[5]=20+30=50$	true	5	4	one(b, 4,5)
	false	5	5	one(b, 5,5)

תוכן מערך a לאחר הפעולה:

50	66	69	74	84
----	----	----	----	----

א.2

מטרת הפעולה: הפעולה מציבה בעבור כל איבר החל מימין לשמאל, את סכום האיבר עם סכום האיברים שמגיעים לפניו במערך. כך שמתקבל באיבר במקום האפס סכום כל האיברים במערך המקורי, באיבר במקום הראשון מתקבל סכום כל האיברים למעט האיבר במקום האפס וכן הלאה.

ב.1+2

מערך c לאחר הזימון: 10, 8, 12, 20, 15, 10

הפעולה מקבלת מערך שלמים כפרמטר ובעבור כל איבר החל מהמקום האפס עד לאיבר אחד לפני אחרון סוכמת את ערך האיבר + ערך האיבר שבא אחריו.

ולכן מערך c במקור היה: 10, 5, 15, -3, 11, -1

שאלה מס. 12

```
public class Task{
    private String title;
    private int minutes;
    private int priority;
    private boolean isComplete;
    ...
}

public class TaskManager{
    private Task [] tasks;
    private int numOfTasks;
    ...
}
```

א.

```
public int doIt(int num){
    if (this.minutes < num){
        this.minutes=0;
        this.isComplete = true;
        return num-this.minutes;
    }
    else return 0;
}
```

ב.

```
public boolean addTak(Task t){
    boolean flag=false;
    if (this.numOfTasks()<this.tasks.length)
    {
        Tasks[this.numOfTasks++]=t;
        flag = true;
    }
    else{
        for(int i=0;i<this.tasks.length &&this.tasks[i].getIsComplete();
            i++);
        if (i<this.tasks.length){
            this.tasks[i]=t;
            flag=true;
        }
    }
    return flag;
}
```

ג.

```
public int importantTask(){
    int index=-1;
    for(int i=0;i<this.tasks.length; i++)
    {
        if (!this.tasks[i].getIsComplete()){
            if (index==-1)
                index=i;
            else
                if (this.tasks[index].getPririty()< this.tasks[i].getPririty())
                    index = i;
        }
    }
    return index;
}
```

1.

```
public void work (int num){
    int index;
    while(num>0){
        index = importantTask();
        if(index !=-1){
            num = this.tasks[index].doIt(num);
        }
        else
            num=0;
    }
}
```

שאלה מס. 13

זימון	S1	S2	If1	If2	place	If3	a	b	c	ערך מוחזר
what("hello", "oeloh")	hello	oeloh	f	f	4	f	oelo	\0	oelo	what("ello", "oelo")
what("ello", "oelo")	ello	oelo	f	f	1	f	o	lo	olo	what("llo", "olo")
what("llo", "olo")	llo	olo	f	f	1	f	o	o	oo	what("lo", "oo")
what("lo", "oo")	lo	oo	f	f	-1	t				false

א. תוצאת הזימון היא false.

ב. דוגמה שתחזיר true: what("hello", "olleh")

ג. הפעולה בודקת האם כל התווים של מחרוזת s1 מופיעים במחרוזת s2. אם כן, מוחזר

true, אחרת מוחזר false.

ד. הפעולה what בצורה איטרטיבית:

```
public static boolean what(String s1, String s2){
    if (s1.length !=s2.length)
        return false;
    for(int i=0;i<s1.length; i++)
        if (s2.indexOf(s1.charAt(i))== -1)
            return false;
    return true;
}
```