

פתרונות מוצע לבחינות מה"ט ב אלגוריתמיקה ותכנות (java)

מועד א' קיץ תש"ף, יולי 2020
מחברת: גב' זהבה לביא, מכללת אורת רחובות

שאלה מס' 1

```

import java.util.Scanner;

public class mainT
{
    /**
     * פתרון שאלה 1
     */
    public static Scanner in = new Scanner (System.in);
    public static void main(String[] args)
    {

        int cnt=0; //נקודות
        int numZugi=0;//כמה זוגיים
        int cntPos=0;//כמה הזוגיים
        int totPos=0;//סכום הזוגיים
        int num= in.nextInt();
        while(num!=0) {
            cnt++;
            if(num%2==0)
                numZugi++;
            if(num>0) {
                cntPos++;
                totPos+=num;
            }
            num= in.nextInt();
        }
        System.out.println("כמה מספרים שנקלו? "+cnt);
        System.out.println("כמה מספרים זוגיים? "+numZugi);
        if(cntPos>0)
            System.out.println("ממוצע מספרים זוגיים: "+((double)totPos)/cntPos);
        else
            System.out.println("לא נקלטו מספרים זוגיים");
    }
}

```

שאלה מס' 2
א.

```

public int getInteval(){
    return (23-hour)*60+(60-min);
}

public class mainT2
{
    /**
     * פתרון שאלה 2 - שיעיר ב
     */
    public static Scanner in = new Scanner (System.in);
    public static void main(String[] args)
    {
        Clock ck= new Clock(in.nextInt(), in.nextInt());
        Clock ck1= new Clock(in.nextInt(), in.nextInt());
        if(ck1.getInteval()>ck.getInteval())
            System.out.println("טישה ראשונה נזחתת מאוחר יותר");
        else if(ck1.getInteval()<ck.getInteval())
            System.out.println("טישה שנייה נזחתת מאוחר יותר");
        else
            System.out.println("הטישות נזחותת באותו הזמן");
    }
}

```

שאלה מס' 3

ב.

ערך מוחזר	num	c
	12345	0
	1234	1
	123	1
	12	2
	1	2
3	0	3

1. הפעולה מחזירה את הערך 3
2. דוגמה למספר שלם וחובי שעבורו הפעולה מחזירה 0: 2468
3. הפעולה סופרת את כמה הספרות האי זוגיות לתוך משתנה c וממחזירה את ערכו.

.ב.

i	res	what(arr[i])	what(res)	if(what(arr[i]<what(res))	פלט
	1245				
1					
		3	2	false	
2					
	23	1	2	true	
3					
	468	0	1	true	
4		2	0	false	
5					
					468

הפעולה **why** ממחזירה את המספר ממערך המספרים, בו כמות הספרות האי זוגיות היא הנמוכה ביותר.

שאלה מספר 4

.א.

```
public static boolean isBalanced(int []arr) {
    if(arr.length>3 && arr.length%2==1) {
        int midNumm=arr[arr.length/2];
        for(int i=0;i<arr.length/2;i++) {
            if(arr[i]<=midNumm)
                return false;
        }
        for(int i=arr.length/2+1;i<arr.length;i++) {
            if(arr[i]>=midNumm)
                return false;
        }
    }
    else return false;
    return true;
}
```

ב. סיבוכיות הפעולה מסעיף ב: **(n)**O. במקרה הגורע יבוצע מעבר על כל המערך לבדיקת האיברים מול האיבר האמצעי.

שאלה מס' 5

- א. הפעולה `תוחזר true`
- ב. הפעולה `one` בודקת האם יש ערך אחד לפחות במערך שהזור על עצמו. אם כן, מוחזר `true`, אחרת `false`.
- ג. הסיבוכיות היא $O(n^2)$. סכום מצטבר של סריקת המערך בעבר כל ערך, החל מהאיבר העוקב לו.
- ד. כן, ניתן לכתוב בצורה ייעילה יותר :

```
for(int i=0; i<arr.length-1;i++)
{
    if(arr[i]==arr[i+1])
        return true;
}
return false;
}
```

אין צורך לסרוק את כל המערך בחיפוש ערך זהה. אם קיים כזה, הוא יופיע מיד אחרי התא הנבדק.
ה. לא, זה לא משנה כיצד המערך ממויין.

שאלה מס' 6

.g.

1. דוגמה למחוזות שתוחזר `:true` **ABCDEFG**
 2. דוגמה למחוזות שתוחזר `:false` **EFG3ABC**
 3. הפעולה `why` בודקת אם ישנותו אחד לפחות במחוזות המתקבלת שאינו אותן אングליית גדולה. אםתו כזה, מוחזר `true`, אחרת מוחזר `false`.
 4. במקרה הגרוע מתבצעת סריקה של כל המחרוזות ולכון הסיבוכיות הינה: $(O(n))$.
- ב.
1. הפעולה רצה במקביל על מערך המחרוזות, איבר מהצד השמאלי של המערך ואיבר מהצד הימני ובודקת אם המחרוזות הנבדקות במקביל שתיهن מכילות רק תווים שנןאותיות אנגליות גדולות או גם תווים אחרים.
- אם הבדיקה מ>Returns `true` או `false` בעבר צמד מחרוזות אחד לפחות – יוחזר `false`.

.2. דוגמה למערך שיזיר **true**:

aBc	ZEH	ABC	Ds21	W3EQ	FCV
-----	-----	-----	------	------	-----

הערכים המוחזרים על ידי בדיקת כל אחת מהמחרוזות במקביל תחזיר ערך שונה.

.2. דוגמה למערך שיזיר **false**:

aBc	ZEH	ABC	Ds21	WQEQ	3fC
-----	-----	-----	------	------	-----

הבדיקה של המחרוזות **aBc** ו- **3fC** תחזיר **false** ולכנו יוחזר **false** על ידי הפעולה **mystery**.

.4. סיבוכיות הפעולה: במקרה הגרוע מתבצע ריצה במקביל על המערך – $n/2$ איברים. הפעולה **why**

ריצה ($O(n)$) ומאהר ומתבצעת לולאה (כפל זמן ריצה) – $O(n^2)$

שאלה מס' 7

.א.

```
public class Truck
{
    private String truckId;
    private String driverName;
    private int numStorage;
    private boolean isFree;
    public Truck(String truckId, String driverName, int numStorage)
    {
        this.truckId = truckId;
        this.driverName = driverName;
        this.numStorage = numStorage;
        this.isFree = true;
    }
}
```

.ב.

```
public static void printNames(Truck [] vec){
    for(int i=0;i<vec.length; i++){
        if(vec[i].isFree() && vec[i].getNumStorage()>=7)
            System.out.println(vec[i].getDriverName());
    }
}
```

ג.

```

public static String getIdFree(Truck [] vec) {
    String id="unable to service";
    int maxNumStorage=-1;
    for(int i=0;i<vec.length; i++) {
        if(vec[i].isFree())
            if(vec[i].getNumStorage()>maxNumStorage) {
                id=vec[i].getTruckId();
                maxNumStorage = vec[i].getNumStorage();
            }
    }
    return id;
}

```

שאלה מס' 8

- א. הפעולה **secret** תחזיר **false**, לאחר והתנאי מתקיים במקרה שבו נבדק הערך האחרון בערך **a** עם הערך השני בערך **b**.
- ב. דוגמה למערכות שיחזרו תוצאה שונה: **a={3,6,9,5} b={10,13,15}**
- ג. הפעולה **secret** בודקת האם האיברים בערך **a** קיימים כלם מהאיברים בערך **b**. אם כן, יוחזר **.false**, אחרת **.true**
- ד. סיבוכיות הפעולה היא : $O(n^2)$. בעבור כל איבר מהערך הראשון נבדק הערך השני.

שאלה מס' 9:

ג.

:res מערך

<u>0</u>	<u>1</u>	<u>2</u>	<u>3</u>	<u>4</u>	<u>5</u>	<u>6</u>	<u>7</u>	<u>8</u>	<u>9</u>	<u>10</u>	<u>11</u>	<u>12</u>	<u>13</u>	<u>14</u>	<u>15</u>	<u>16</u>	<u>17</u>	<u>18</u>	<u>19</u>	<u>20</u>	<u>21</u>	<u>22</u>	<u>23</u>	<u>24</u>	<u>25</u>
<u>3</u>	<u>2</u>	<u>0</u>	<u>1</u>	<u>0</u>	<u>3</u>	<u>0</u>	<u>0</u>	<u>0</u>	<u>0</u>	<u>1</u>	<u>0</u>														

פלט:**A A A****B B****D****F F F****K**

ב. הפעולה סופרת את כמות המופעים של כל אחד מהתווים במאזעות מערך מונימ בגודל 26 ומדפיסה כל אותן, שמוינעה לפחות פעם אחת, בצורה רציפה לפי מספר המופעים שלה, בשורות נפרדות.

שאלה מספר 10:

א. פלט של המחלקה:

0 1**1 -1****2 1****3 -1****4 1****5 -1****6 1****7 -1**

0 5**1 -3****2 8****3 3****4 6****5 7****6 1****7 -1**

0 -1
1 1
2 7
3 6
4 3
5 8
6 -3
7 5

ב. הערה: לאחר ולא ניתן להזכיר יותר מערך אחד על ידי פעולה, תוחזר מהירות, המכילה את שני הערכים (ערך מינימלי וערך מקסימלי).

```
public String minMaxValues(){
    int max, min;
    max=min=this.arr[0];
    for(int i=0;i<arr.length; i++){
        if( arr[i]<min)
            min=arr[i];
        else if (arr[i]>max)
            max = arr[i];
    }
    return min+" ,"+max;
}
```

שאלה מס' 11

.א.

1. תוכן המערך לאחר זימון הפעולה: { 12,678,45,123,345 }

2. דוגמה לערך בגודל של 5 תאים, כך שהתוכן לא ישנה לאחר ביצוע הפעולה: 22,78,36,129, 45

3. הפעולה מסדרת את המערך המתkeletal כך שבחלקו השמאלי יופיעו המספרים הזוגיים ואח"כ המספרים האי זוגיים (לא לפי הסדר המקורי).

ב.

- .1. תוכן המערך לאחר זימון הפעולה: **{12,34,46,55,67}**
- .2. הפעולה מסדרת את המערך כך ש: המספרים הזוגיים יופיעו בתחילת (משמאל לימין) ואח"כ האי זוגיים (תווך הקפדה על הסדר המקורי)
- .3. כאשר המערך מסודר מתחילת (כל הערכים הזוגיים משמאל ואח"כ הערכים האי זוגיים) זימון שתי הפעולות יביא לאותה התוצאה.
- .4. סיבוכיות זמן הריצה של הפעולה **order** : **O(n)** . סכום סריקות של המערך, כאשר כל סריקה היא **O(n)** ולכון סיבוכיות זמן הריצה **(n)O** . סריקה של המערך פעם אחת. סיבוכיות זמן הריצה של הפעולה **arrange O(n)** : **arrange**

שאלה מספר 12

ג.

```
public class Familiy
{
    private String name;
    private int num;
    private double totalSum;
    public Familiy(int num)
    {
        this.num = num;
        this.totalSum = 0;
    }
    /**
     * @return the totalSum
     */
    public double getTotalSum()
    {
        return this.totalSum;
    }
    public void input()
    {
        int age;
        double tot=100;
        for(int i=0;i<this.num;i++)
        {
            age=in.nextInt();
            if(age<=3)
                tot+=20.5;
            else if (age >3 && age <=12)
                tot+=30;
            else tot+=40.5;
        }
        this.totalSum = tot;
    }
}
```

.ב.

```

public class mainT12
{
    /**
     * פתרון שאלה 12
     */
    public static Scanner in = new Scanner(System.in);
    public static void main(String[] args)
    {
        String name = in.next();
        int num;
        while (!name.equals("STOP")){
            num=in.nextInt();
            Familiy f = new Familiy(num, name);
            f.input();
            System.out.println("family name:"+name+
totalSum:+f.getTotalSum());
            name = in.next();
        }
    }
}
    
```

שאלה מספר 13

- א. תוצאת הימון : **"ABCDAbe"**
- ב. דוגמה למחוזות :**s2** :**"A4B23C7"**
- ג. דוגמה למחוזות :**s3** :**"A4B2c3Cb7a"**
- ד. כן, תיתכן מחרוזת כזו. היא תורכב מאותיות גדולות, קטנות ותוויות אחרים, לדוגמה:
"A!B@2c37Cb7*a"
- ה. לא קיימת מחרוזת כזו מאחר וכאשר הפעולה נתקלה בטו שהוא גודלה הוא יופיע בתחילת המחרוזת המוחזרת.
- ו. הפעולה יוצרת מחרוזת חדשה מהמחוזות המתקבלת, המכילה בתחילתה את האותיות הגדולות לפי הסדר שלן במחרוזת המקורית (אם ישן כאלה) ואח"כ אותיות קטנות בספר הפוך לסדר שלן במחרוזת המקורית (אם ישן כאלה).
- ז. לא כל ערך שלם מתאים לזמן הפעולה, במקרה של ערך שלילי או ערך גדול מאורך המחרוזות המתקבלת נקבל שגיאת זמן ריצה. יש לבדוק שערכו של **k** אינו שלילי או גדול מאורך המחרוזות בתנאי העצירה (במקרה זה תוחזר מחרוזת ריקה).