

## פתרונות לשאלון 97104 – אלגוריתמיקה ותוכנות – מועד א' אביב 23

**שאלה 1 : (12 נקודות)**

```
public class Main {
    public static void main(String[] args) {
        Scanner in = new Scanner((System.in));

        int sumDigits = 0;
        int num;
        for (int i=0;i<100;i++) {
            num = in.nextInt();
            if (num >=100 && num <1000) {
                sumDigits = 0;
                for (int j=0;j<3;j++) {
                    sumDigits += num % 10;
                    num /= 10;
                }
                System.out.println(" sum digits: " + sumDigits);
            }
        }
    }
}
```

**שאלה 2 : (12 נקודות)**

```
import java.util.Scanner;

public class Main {

    public static void main(String[] args) {
        Scanner in = new Scanner((System.in));

        int stringCounter = 0;
        String str;
        str = in.nextLine();
        while (str.length()<=13) {
            if (str.indexOf("X") != -1 && str.indexOf('Y') == -1)
                stringCounter++;
            str = in.nextLine();
        }
        System.out.println("number of strings: " + stringCounter);
    }
}
```

### שאלה 3 : (12 נקודות)

א. (6 נקודות)

```
public static boolean isInArray(int[] arr,int ind,int value) {
    for (int i=ind+1;i<arr.length;i++)
        if (arr[i] == value)
            return true;
    return false;
}
```

ב. (6 נקודות)

```
public static boolean isUniqueArray(int[] arr) {

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

### שאלה 4 : (12 נקודות)

א. (6 נקודות)

```
b[0] = ("Alex", "1234", "14", 18);
b[1] = ("Benny", "7596", "26A", 17);
b[2] = ("Charly", "6051", "26B", 16);
b[3] = ("Dani", "4472", "778", 21);
b[4] = ("Dani", "4472", "778", 21);
b[0] = ("Alex", "1234", "14", 18);
b[1] = ("Benny", "7596", "26A", 26);
b[2] = ("Charly", "6051", "26B", 16);
b[3] = ("Dani", "4472", "778", 17);
b[4] = ("Dani", "4472", "778", 17);
Alex 18
Benny 26
Charly 16
Dani 17
Dani 17
18.8 // passengers average
1 // number of bus that passengers bigger than 20
```

ב. (6 נקודות)

```
public static String[] notPopular(Bus[] b, int num) {
    int counter = 0;
    for (int i = 0; i < b.length;i++)
        if (b[i].getPassengers < num)
            counter++;

    String[] strArr = new String[counter];
    int index =0;
    for (int i=0;i<b.length;i++)
        if (b[i].getPassengers < num)
            strArr[index++] = b[i].getPlateNumber();
    return strArr;
}
```

**שאלה 5 : (12 נקודות)**

א. (4 נקודות)

```
public static boolean brothers(int num1, int num2) {  
    if (num1<=0||num2<=0)  
        return false;  
    int temp1=num1;  
    int temp2=num2;  
    int MSD1=0,MSD2=0;  
    while (temp1>0) {  
        MSD1 = temp1;  
        temp1 /= 10;  
    }  
    while (temp2>0) {  
        MSD2 = temp2;  
        temp2 /= 10;  
    }  
    return (num1%10==num2%10 && MSD1==MSD2);  
}
```

ב. (4 נקודות)

עבור :

123	78002	591	13	7	25
-----	-------	-----	----	---	----

int[] b = {51,72,13,12343,7,2345,71}

ג. (2 נקודות)  
לא, למשל הדוגמא b שנותתי, אם נהפוך את המערכיהם או אין איבר מותאים למספר 71 במערך השני.

ד. (2 נקודות)  
הפעולה בודקת עבור כל איבר במערך a יש אח במערך b.

**שאלה 6 :** (12 נקודות)

א. (6 נקודות)

1. א

```

str = "AxyBZ3"
c = 0 i=0 str[i] = 'A' %
c = 1 i=1 str [i] = 'x' % 1
c = 1 i=2 str[i] = 'y' % 1 2
c = 1 i=3 str[i] = 'B' % 1 2
c = 2 i=4 str[i] = 'Z' % 1 2
c = 2 i=5 str[i] = '3' % 1 2 5
c = 3

```

% 1 2 5 %
3

2. א

"AxByC6"

א. 3. מחזירה את כמות התווים במחזוריות שאינם אותיות גדולות , מדפסה את המיקומים של התווים השונות  
מאותיות גדולות

ב. (6 נקודות)

1. ב

{"aa","Aa","aBBd","AACbcd","ABCDqw","1ABCDE3456"};

2. ב

```

%0 1 %
0
%1 %
1
%0 3 %
2
%3 4 5 %
3
%4 5 %
4
%0 6 7 8 9 %
5

```

**שאלה 7 : (15 נקודות)**

א. (5 נקודות)

```
public Box (String color,double weight) {  
    this.width = 1 + (int)(Math.random()*(10));  
    this.length = 1 + (int)(Math.random()*(10));  
    this.height = 1 + (int)(Math.random()*(10));  
    this.color = color;  
    this.weight = weight;  
}
```

ב. (5 נקודות)

```
public static String heaviestBox(Box[] arr) {  
    double maxWeight = 0;  
    String maxColor = "";  
    for (int i=0;i<arr.length;i++) {  
        if (arr[i].getWeight()>maxWeight) {  
            maxWeight = arr[i].getWeight();  
            maxColor = arr[i].getColor();  
        }  
    }  
    return maxColor;  
}
```

ג. (5 נקודות)

```
public static Box heaviestColorBox (Box[] arr, String color) {  
    double maxWeight = -1;  
    int maxIndex=0;  
  
    for (int i=0;i<arr.length;i++)  
        if (arr[i].getColor().equals(color))  
            if (arr[i].getWeight() > maxWeight) {  
                maxWeight = arr[i].getWeight();  
                maxIndex = i;  
            }  
    if (maxWeight == -1)  
        return null;  
    return arr[maxIndex];  
}
```

## **שאלה 8 : (15 נקודות)**

א. (4 נקודות)

```

int[] a = {8,15,7,10}
int[] b = {6,7,2}

i=0 k=0 a[i]=8 b[k]=6
i=0 k=1 a[i]=8 b[k]=7
i=0 k=2 a[i]=8 b[k]=2
i=1 k=0 a[i]=15 b[k]=6
i=1 k=1
i=2 k=2
i=3 k=0
i=3 k=1 a[i]=7 b[k]7 false

```

ב. (3 נקודות)

```

int[] a = {8,15,7,10}
int[] b = {6,4,2,3}

```

ג. (2 נקודות)

הפעולה תחזיר true אם כל ערך במערך a גדול ממש מכל האיברים במערך b

ד. (1 נקודה)

$O(n*m)$

ה. (4 נקודות)

```

public static boolean what1(int[] a, int[] b)
{
    int min = a[0];
    for (int i = 1; i < a.length; i++) {
        if (a[i] < min)
            min = a[i];
    }
    int max = b[0];
    for (int i = 1; i < b.length; i++) {
        if (b[i] > max)
            max = b[i];
    }

    return (min > max);
}

```

ו. (1 נקודה)

$O(n+m)$

**שאלה 9 : (15 נקודות)**

א. (5 נקודות)

```
public static boolean isPalindrom(String str) {  
    int len = str.length();  
    for (int i=0;i<len/2;i++) {  
        if (str.charAt(i)!=str.charAt(len-1-i))  
            return false;  
    }  
    return true;  
}
```

ב. (5 נקודות)

```
public static boolean isLetterPalindrom(String str) {  
    int len = str.length();  
    int left = 0;  
    int right = len-1;  
    boolean areLetters = false;  
    while (left<right) {  
        areLetters = true;  
        if (!isLetter(str.charAt(left))) {  
            areLetters = false;  
            left++;  
        }  
        if (!isLetter(str.charAt(right))) {  
            areLetters = false;  
            left++;  
        }  
        if (areLetters) {  
            if (left <= right) {  
                if (str.charAt(left) != str.charAt(right))  
                    return false;  
                else {  
                    left++;  
                    right--;  
                }  
            }  
        }  
    }  
    return true;  
}
```

ג. (5 נקודות)

```
public static String[] getPalArr(String[] arr) {  
    int count=0;  
    for (int i=0 ; i<arr.length;i++)  
        if (isLetterPalindrom(arr[i]))  
            count++;  
    String[] res = new String[count];  
    int index=0;  
    for (int i=0 ; i<arr.length;i++)  
        if (isLetterPalindrom(arr[i]))  
            res[index++]=arr[i];  
    return res;  
}
```

**שאלה 10 : (11 נקודות)**

א. (3 נקודות)

```
public boolean better (Car other) {  
    if (this.type>other.type)  
        return true;  
    if (this.type==other.type)  
        if (!this.isManual&&other.isManual)  
            return true;  
    return false;  
}
```

ב. (3 נקודות)

```
public int getPrice () {  
    int days = pickDate.difference(returnDate);  
    if (this.car.type=='A')  
        return (days*100);  
    if (this.car.type=='B')  
        return (days*150);  
    if (this.car.type=='C')  
        return (days*180);  
    if (this.car.type=='D')  
        return (days*240);  
    else  
        return 0;  
}
```

ג. (5 נקודות)

```
public static void clientsReport(Rent[] rents, String[] names) {  
    for (int i=0; i<names.length;i++) {  
        int priceSum =0;  
        for (int j=0;j<rents.length;j++) {  
            if (names[i].equals(rents[j].name)) {  
                priceSum += rents[j].getPrice();  
                System.out.println(rents[j].toString());  
            }  
        }  
        System.out.println(priceSum);  
    }  
}
```

**שאלה 11: 11 נקודות**

א. 5 נקודות

```
public static boolean isLone (int[][] mat, int row, int col) {  
    for (int i=0;i<mat.length;i++) {  
  
        if (mat[i][col] == mat[row][col]&&i!=row)  
            return false;  
    }  
    for (int i=0;i<mat[0].length;i++) {  
        if (mat[row][i] == mat[row][col]&&i!=col)  
            return false;  
    }  
    return true;  
}
```

ב. 4 נקודות

```
public static int maxLone (int[][] mat) {  
    int maxLoan = -1;  
    for (int i=0;i<mat.length;i++)  
        for (int j=0;j<mat[i].length;j++) {  
            if (isLone(mat,i,j))  
                if (mat[i][j] > maxLoan)  
                    maxLoan = mat[i][j];  
        }  
    return maxLoan;  
}
```

ג. 2 נקודות

$O(n+m)$

$O(n*m*(n+m))$

**שאלה 12 : (11 נקודות)**

א. (3 נקודות)

mystry("HELLO","HELP")

s1 = "HELLO" s2 = "HELP"  
s1 = "ELLO" s2 = "LP"  
s1 = "LLO" s2 = "P"  
s1 = "LO" s2 = ""  
false.

ב. (1 נקודה) בודקת האם s2 היא תת מחרוזת ב s1 שמתהילה בתחילת s1

ג. (3 נקודות)

secret ("MYJOBTEST", "JOB")

s1 = "MYJOBTEST" s2 = "JOB"  
mystry(s1,s2)= false;  
  
s1 = "YJOBTEST" s2="JOB"  
mystry(s1,s2)= true;

true.

ד. (1 נקודה) בודקת האם s2 תת מחרוזת במקום כלשהו בס1

ה. (3 נקודות)

```
public static boolean secret2 (String s1, String s2) {  
    return (s1.contains(s2));  
}
```