

מדעי המחשב - 2 יחידות לימוד
פתרון בחינת הבטחות

פרק א

שאלה 1:

```
a = input.nextInt ();
b = input.nextInt ();
c = input.nextInt ();
if (a + b == 200 || a + c == 200 || b + c == 200)
    System.out.println (" yes ");
else
    System.out.println (" no ");
```

שאלה 2:

m1	m2	n	k	k≤4	s	t	s*t>n	s-t>0	פלט
0	0	8							2 3
1	1		1	T	6	4	T	T	
			2	T	2	3	F	F	
2	2		3	T	5	2	T	T	
	3		4	T	7	0	F	T	

שאלה 3:

קטע תכנית: בכל תא במערך נמצא המספר 1 או המספר 7.

```
int count1 = 0, count2 = 0;
for (int i = 0 ; i < 51 ; i++)
    if (arr[i] == 1)
        count1 ++ ;
    else
        count2 ++ ;
if (count1 > count2)
    System.out.println ( 1 );
else
    System.out.println ( 7 );
```

אפשרות נוספת:
ניתן לספור רק את מספר המופעים של 1 ל- count1
ובסיום הלולאה לחשב: count2 = 51 - count1;

שאלה 4:

a	b	a ≠ b	num1	num2	פלט
2	3	T	23	32	23
					32
					55

ב. 3, 3 (כל שני מספרים זהים).

ג. (i) 3, 1

(ii) 4, 4

שאלה 5:

```

int a, b;
for (int i = 0; i < 38; i++)
{
    a = input.nextInt ();
    b = input.nextInt ();
    System.out.println ( a + b );
    if (b == 0)
        System.out.println ("indivisible");
    else
        System.out.println( a / b);
}

```

הערה: בשפת Java: שלם / שלם ← שלם ולכן אין צורך בהמרה לשלמים.

פרק ב

שאלה 6:

```
// קלט : מחיר של כל אחד מפריטי הלבוש שנקנו בחנות.  
// פלט : גובה ההנחה (אם קיימת) והמחיר לאחר ההנחה.  
// האם מגיע לקונה שובר הנחה לקניה הבאה
```

```
import java.util.Scanner;  
  
public class Targil6 {  
    public static void main (String [] args) {  
        Scanner input = new Scanner (System.in);  
        int count = 0, price, sum = 0, n, discount;  
        System.out.print ("? מחיר הפריט הראשון → ");  
        price = input.nextInt ();  
        while (price != 0)  
        {  
            sum = sum + price;  
            count ++ ;  
            System.out.print ("? מחיר הפריט הבא → ");  
            price = input.nextInt ();  
        }  
        n = sum / 800 ;  
        if (n != 0)  
            discount = 50 * n ;  
        else  
            discount = 0;  
        System.out.print ( discount + " : גובה ההנחה");  
        sum = sum - discount ;  
        System.out.println (sum + " : מחיר הקניה");  
        if (count >= 4)  
            System.out.println (" זכאי לשובר הנחה");  
    }  
}
```

שאלה 7:

	0	1	2	3	4	5	6	7	8	9	10	
arr	6	7	8	11	2	0	2	3	17	7	6	א.

sum	g	k	i	$i \geq 0$	I arr[i]	II arr[g-i]	I == II
0	10	5					
1			4	T	2	2	T
			3	T	11	3	F
			2	T	8	17	F
2			1	T	7	7	T
3			0	T	6	6	T
			-1	F			

בסיום הביצוע ערכו של sum יהיה: 3

	0	1	2	3	4	5	6	7	8	9	10	
a	1	2	3	4	5	6	1	2	3	4	5	ב.

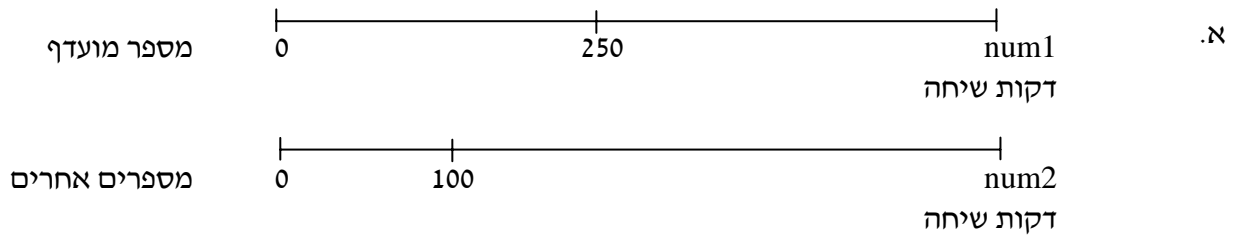
כל מערך שאין בו סימטריה.

ג.

sum	g	k	i	$i \leq g$	I arr[i]	II arr[g-i]	I == II
0	10	5					
1			6	T	2	2	T
			7	T	3	11	F
			8	T	17	8	F
2			9	T	7	7	T
3			10	T	6	6	T
			11	F			

ערכו של sum יהיה: 3

שאלה 8:



```
// טענת כניסה: מספר דקות השיחה למספר מועדף num1
// מספר דקות שיחה למספרים אחרים num2
// טענת יציאה: הסכום בשקלים שיש לשלם
```

```
public static double pay (int num1, int num2)
{
    double sum = 0;
    if (num1 > 250)
        sum = sum + (num1 - 250) * 0.2;
    if (num2 > 100)
        sum = sum + (num2 - 100) * 0.3;
    return sum;
}
```

ב. קטע התכנית:

```
int num1, num2;
double toPay;
System.out.print (" → מספר דקות שיחה למספר מועדף ? ");
num1 = input.nextInt ();
while (num1 >= 0)
{
    System.out.print (" → מספר דקות שיחה למספרים אחרים ? ");
    num2 = input.nextInt ();
    toPay = pay (num1 , num2);
    System.out.println (toPay + " : הסכום לתשלום");
    System.out.print (" → מספר דקות שיחה למספר מועדף ? ");
    num1 = input.nextInt ();
}
```

פרק ג

שאלה 9:

```
public class Targil9{
    // טענת כניסה: מערך דו-ממדי m, מספר שורה line, ומספר עמודה col
    // טענת יציאה: מוחזר 1 אם האיבר שבמקום זה הוא איבר צומת.
    static int nodeItem (int [][] m, int line, int col)
    {
        if (sumOfLine (m, line) == sumOfColumn (m, col))
            return 1;
        else
            return 0;
    }
    // טענת כניסה: מטריצה m ושורה line
    // טענת יציאה: מוחזר סכום השורה
    static int sumOfLine (int [][]m, int line)
    {
        int sum = 0;
        for (int i = 0 ; i < 17 ; i++)
            sum = sum + m[line][i];
        return sum;
    }
    // טענת כניסה: מטריצה m ועמודה col
    // טענת יציאה: מוחזר סכום העמודה.
    static int sumOfColumn (int [][]m, int col)
    {
        int sum = 0;
        for (int i = 0 ; i < 17 ; i++)
            sum = sum + m[i][col];
        return sum;
    }
    public static void main (String [] args){
        int count = 0;
        int [][] mat = new int [17][17];
        for (int i = 0 ; i < 17 ; i ++ )
            for (int j = 0 ; j < 17 ; j ++ )
                if (nodeItem (mat, i, j) == 1)
                    count ++ ;
        System.out.println (count + " : מספר איברי הצומת שנמצאו ");
    }
}
```

שאלה 10:

```
import java.util.Scanner;
```

```
public class Targil10{
```

```
// טענת כניסה: שני מספרים שלמים first ו-second בני 4 ספרות כל אחד, המייצגים שעות.
// טענת יציאה: מספר הדקות שעברו בין first ל-second.
// הנחה:  $1200 \leq \text{first} < \text{second} \leq 2359$ 
```

```
static int elapsedTime (int first, int second)
```

```
{
```

```
    int firstMinute = first % 100;           // מספר הדקות
```

```
    int firstHour = first / 100;             // מספר השעות
```

```
    int secondMinute = second % 100;        // מספר הדקות
```

```
    int secondHour = second / 100;         // מספר השעות
```

```
    int hours = secondHour - firstHour;
```

```
    int minutes;
```

```
(* if (secondMinute >= firstMinute)
    minutes = secondMinute - firstMinute;
```

```
else
```

```
{
```

```
    minutes = (60 + secondMinute) - firstMinute;
```

```
    hours = hours - 1;
```

```
}
```

```
return (hours * 60 + minutes);
```

```
}
```

```
public static void main (String [] args){
```

```
    Scanner input = new Scanner (System.in);
```

```
    int min = 12*60;
```

```
    int first, second, time;
```

```
    for (int i = 0 ; i < 127 ; i++)
```

```
    {
```

```
        System.out.print ("שעת ההתחלה → ");
```

```
        first = input.nextInt();
```

```
        System.out.print ("שעת הסיום → ");
```

```
        second = input.nextInt();
```

```
        time = elapsedTime (first, second);
```

```
        System.out.println (time + " : משך זמן הריצה");
```

```
        if (time < min)
```

```
            min = time;
```

```
    }
```

```
    System.out.println (min + " : זמן מינימלי");
```

```
}
```

```
}
```

דרך פשוטה יותר לחשוב מספר הדקות
(לפי הצעתה של עפרה בן-ארי):

```
return
```

```
(secondHour*60 + secondMinute) -
```

```
(firstHour*60 + firstMinute);
```

(* למעשה, כדי לחשב את מספר הדקות, מספיק לחשב:

```
minutes := secondMinute - firstMinute;
```

```
return (hours * 60 + minutes);
```

(אם דקות ההתחלה קטן מדקות הסיום יהיה minutes מספר שלילי)