

מדעי המחשב - 2 יחידות לימוד  
פתרון בחינת הבגרות

פרק א

שאלה 1:

קלט: 2, 1, 3, 6 ⇒

C# קטע הקוד בשפת

```
k = 1;
num = 0;
for (i = 1 ; i < 5 ; i++)
{
    x = int.Parse(Console.ReadLine());
    if (x % 2 == 0)
        num = (num * 10) + x;
    else
    {
        k = k * 10;
        num = num + (x * k);
    }
}
Console.WriteLine (num);
```

k	num	i	x	x זוגי?	פלט
1	0				3126
	2	1	2	T	
10	12	2	1	F	
100	312	3	3	F	
	3126	4	6	T	

שאלה 2:

	x	y	x > 0	פלט
א.	10	4	T	-2
	6		T	
	2		T	
	-2			
ב.	-10	4	F	-10
ג.	10	4	T	6
	6			

סעיף ב: כל מספר שלילי או 0 מתקבל.  
(אין צורך בטבלת מעקב עבור סעיף זה).

**שאלה 3:**

קטע תכנית : סכום ומספר המספרים התלת ספרתיים במערך a שגודלו 62.

```
sum = 0;
count = 0;
for (i = 0 ; i < 62 ; i++)
    if (a[i] >= 100 && a[i] <= 999) // if (a[i] > 99 && a[i] < 1000) : אפשר גם :
    {
        sum = sum + a[i];
        if (a[i] > 248)
            count ++ ;
    }
Console.WriteLine (sum + " : סכום המספרים התלת ספרתיים");
Console.WriteLine ("היו " + count + " מספרים תלת-ספרתיים גדולים מ- 248");
```

**Java**

```
num = input.nextInt();
sum = 0;
count = 0;
while (num < 0)
{
    sum = sum + num;
    count = count + 1;
    num = input.nextInt();
}
System.out.println (sum / num);
```

**שאלה 4:**

השגיאה : בשורה האחרונה : מודפס סכום המספרים חלקי המספר האחרון. (השגיאה חמורה במיוחד אם הערך האחרון שנקלט ל- num הוא 0). התיקון : יש לחלק במספר המספרים : Console.WriteLine ((double)sum / count); שים ♥ : ללא המרת אחד המשתנים ל- double לא יקבל ממוצע אמיתי.

**שאלה 5:**

**Java**

```
int check (int a, int b)
{
    if (b - a > 5)
        return 1;
    else
        return 0;
}
```

```
I x = check (6, 2);
System.out.println (x);
II a = 2;
b = 8;
System.out.println (check);
III m = 7;
for (i = 1 ; i < 4 ; i++)
    System.out.println (check (i, m));
```

- א. משפט הזימון תקין. הפלט 0
- ב. משפט הזימון אינו תקין. חסרים הפרמטרים שיישלחו לפונקציה.
- ג. משפט הזימון תקין :

תכנית		פונקציה			ערך מוחזר / פלט
m	i	a	b	b-a > 5	
7	1	1	7	T	1
	2	2	7	F	0
	3	3	7	F	0

פרק בלפינה 6:

```

//--- קלט: מספר הכרטיסים להגרלה, למופע ולמזנון שנרכשו ביריד.
//--- פלט: מספר המשתתפים ביריד, כמה כרטיסים מכל סוג נמכרו, ומספר כרטיסי ההגרלה שניתנו חינם.

public class Hgrala {
    public static void main (String [] args){
        int hagT, mufT, mizT ; // מספר הכרטיסים שנרכשו להגרלה, מופע ומזנון
        int sHag = 0 , sMuf = 0 , sMiz = 0; // צוברים לכל סוגי הכרטיסים
        int mone = 0 , freeHagT = 0; // מונה מספר המבקרים, מספר כרטיסי החינם

        System.out.print (" → מספר הכרטיסים להגרלה ? ");
        hagT = int.Parse(Console.ReadLine());
        while (hagT != -1)
        {
            System.out.print (" → מספר הכרטיסים למופע ? ");
            mufT = input.nextInt ();
            System.out.print (" → מספר הכרטיסים למזנון ? ");
            mizT = input.nextInt ();

            mone ++ ; // ספירת מספר המבקרים
            sHag = sHag + hagT ; // ספירת מספר כרטיסי ההגרלה
            sMuf = sMuf + mufT; // ספירת מספר הכרטיסים למופע
            sMiz = sMiz + mizT; // ספירת הכרטיסים למזנון

            if (hagT > 10)
                freeHagT ++ ; // בדיקת זכאות לכרטיס הגרלה חינם

            System.out.print (" → מספר הכרטיסים להגרלה ? ");
            hagT = int.Parse(Console.ReadLine());
        }

        Console.WriteLine ("היו " + mone + " מבקרים ביריד");
        Console.WriteLine ("נמכרו " + sHag + " כרטיסי הגרלה");
        Console.WriteLine ("נמכרו " + sMuf + " כרטיסים למופע");
        Console.WriteLine ("נמכרו " + sMiz + " כרטיסים למזנון");
        Console.WriteLine ("חולקו " + freeHagT + " כרטיסי חינם להגרלה");
    }
}

```

**שאלה 7:**

א. הפונקציה:

```
//-----
// ט.כניסה: מספר דו-ספרתי שלם וחיובי שספרותיו שונות זו מזו.
// ט.יציאה: סכום המספרים החד-ספרתיים שבין ספרות המספר.
public static int bigSum ( int a )
{
    int d1, d10;           { ספרות המספר }
    int i, sum = 0;
    d1 = num % 10;
    d10 = num / 10;
    if ( d1 < d10 )
        for ( i = d1 ; i <= d10 ; i ++ )
            sum = sum + i ;
    else
        for ( i = d1 ; i >= d10 ; i -- )
            sum = sum + i ;
    return sum;
}
```

דן אחרת לחישוב סכום הספרות:

```
if ( d1 < d10 )
    for ( i = d1 ; i <= d10 ; i ++ )
        sum = sum + i ;
else
    for ( i = d10 ; i < d1 ; i ++ )
        sum = sum + i ;
```

ב. התכנית:

```
//--- קלט: מספרים דו-ספרתיים למערך
//--- פלט: מספר המספרים שבהם סכום הספרות שבין ספרות האחדות וספרות העשרות גדול מ-15
public static void main (String [] args)
{
    int [] arr = new int [23] ;
    int sum, i, count;
    // כאן יבוא קטע תכנית המבצע קלט למערך הנתון
    count = 0;
    for ( i = 0 ; i < 23 ; i ++ )
    {
        sum = bigSum (arr[i]);
        if (sum > 15)
            count ++ ;
    }
    Console.WriteLine (" נמצאו " + count + " מספרים המקיימים את התנאי ");
}
```

בשאלה ביקשו רק קטע תכנית, לכן מספיק לכתוב קטע זה.

**שאלה 8:**

```

Java
count = 0;
for (i = 0 ; i < n ; i++)
{
    if (a[i] != (b[i*2]+b[i*2+1]))
        a[i] = -1;
    else
        count = count + 1;
}
System.out.println (n-count);
    
```

b

0	1	2	3	4	5	6	7	8	9
0	2	4	4	5	2	8	8	10	10

a

0	1	2	3	4
2	17	3	6	20
	-1	-1	-1	

n	count	i	a[i] ≠ b[i*2] + b[i*2+1]	קיום התנאי	פלט
5	0				3
	1	0	2 ≠ 0 + 2	F	
		1	17 ≠ 4 + 4	T	
		2	3 ≠ 5 + 2	T	
		3	6 ≠ 8 + 8	T	
	2	4	20 ≠ 10 + 10	F	

ב. הסבר: מכיוון שמודפס הערך של {n - count} והמונה גדל ב-1 עבור כל ערך של מערך a שאינו שונה מסכום 2 האיברים במערך b, הרי שאיברי מערך a צריכים להיות שווים לסכום שני התאים המתאימים במערך b.

a

0	1	2	3	4
2	8	7	16	20

פרק גלסוף 9:

```
//--- קלט: מערך דו-ממדי (מטריצה) שאיבריו הם 0 או 1 (בלבד)
//--- פלט: מספר האיברים ה"מוקפים" במערך.
//--- איבר "מוקף" הוא איבר שערכו 1 ואורך השרשרת בשורה ובעמודה שלו שווים
```

```
public class Matrix {
    final static int N = 10;

    public static void main(String [] args){
        int [][] m = new int [N][N];
        int i, j, count = 0;

        // כאן יבוא קטע תכנית המבצע קלט למערך הנתון
        for (i = 0 ; i < N ; i ++ )
            for (j = 0 ; j < N ; j ++ )
                count = count + surroundedItem (m, i, j);

        System.out.println(count + " : מספר האיברים המוקפים הוא");
    }

    //-----
    //--- מוחזר אורך השרשרת בשורה r במטריצה m ---
    public static int rowChain (int [][] m, int r)
    {
        int j, count = 0;
        for (j = 0 ; j < N ; j ++ )
            count = count + m[r][j];
        return count;
    }

    //-----
    //--- מוחזר אורך השרשרת בעמודה c במטריצה m ---
    public static int colChain (int [][] m, int c)
    {
        int i, count = 0 ;
        for (i = 0 ; i < N ; i ++ )
            count = count + m[i][c];
        return count;
    }
}
```

הנחת התכנית: איברי המערך הם 0 או 1  
ויש רק שרשרת אחת בכל שורה או עמודה,  
ולכן מספיק לסכם את איברי השורה/עמודה.

```

//-----
//---          טענת כניסה: מטריצה m, שורה i ועמודה j ---
//---          טענת יציאה: 1 אם האיבר במקום זה הוא איבר "מוקף" ו-0 אחרת. ---
//---          איבר מוקף = ערכו של האיבר שבתא (i, j) ואורך שרשרת השורה שלו. ---
//---          שווה לאורך שרשרת העמודה שלו ---
public static int surroundedItem (int [][] m, int i, int j)
{
    if (m[i][j] == 1 && rowChain (m,i) == colChain (m,j))
        return 1
    else
        return 0;
}
} //--- end of class ---

```

**לצורה 10:**

**טבלת משתנים:**

שם משתנה	טיפוס משתנה	תפקיד המשתנה
books	מערך בגודל 315 של שלמים	מערך הצוברים לסך הספרים שהוזמנו מכל סוג.
num	שלם	מספר העותקים של הספר שהזמין התלמיד.
cat	שלם (בין 1 ל- 315)	המספר הקטלוגי של הספר שהזמין התלמיד.
sum	שלם	הסכום הכולל שעל התלמיד לשלם עבור הזמנתו.
total	שלם	הסכום הכולל שעל בית הספר לשלם עבור כלל ההזמנה.

**תת-קצ'אות:**

```

public static void studentOrder (int [] books) (books) הזמנת-תלמיד
    טענת כניסה: מערך הספרים.
    טענת יציאה: הפעולה מוסיפה לכל תא במערך הספרים, המתאים למספר הקטלוגי של הספר שהזמין
    התלמיד, את מספר הספרים שהוזמנו.

public static void booksOrdersShow (int [] books) (books) הצג-הזמנות
    טענת כניסה: מערך הספרים.
    טענת יציאה: הפעולה מדפיסה לכל ספר את מספר ההזמנות שבוצעו עבורו.

public static int booksTotalOrder (int [] books) (books) סכום-סך-ההזמנות
    טענת כניסה: מערך הספרים.
    טענת יציאה: מוחזר סכום איברי המערך מוכפל במחיר הספר (28 ₪).

```

```
//--- קלט : הזמנת הספרים לכל תלמיד ---
//--- פלט : פירוט וסך ההזמנות לבית הספר. ---

public class BooksOrder {

    final static int N = 620;
    final static int BOOK_NUM = 315;
    final static int BOOK_PRICE = 28;

    public static void main (String [] args){
        int [] books = new int [BOOK_NUM];
        int total;

        for (int i = 1 ; i <= N ; i ++ )
            studentOrder (books);

        booksOrdersShow (books);

        total = booksTotalOrder (books);
        Console.WriteLine (" סך הזמנת בית הספר : " + total + " שח ");
    }

    //-----
    // ט. כניסה : מערך הספרים.
    // ט. יציאה : הפעולה מוסיפה לכל תא במערך הספרים, המתאים למספר הקטלוגי של הספר
    // שהזמין התלמיד, את מספר הספרים שהוזמנו.

    public static void studentOrder (int [] books)
    {
        int num, cat, sum = 0;

        System.out.print (" (1-" + BOOK_NUM + ") מספר קטלוגי לספר → ");
        cat = input.nextInt ();
        while ( cat != 0 )
        {
            num = input.nextInt (" מספר הספרים → ");
            books [cat-1] = books [cat-1] + num;
            sum = sum + num;

            System.out.print (" (1-" + BOOK_NUM + ") מספר קטלוגי לספר → ");
            cat = input.nextInt ();
        }
        Console.WriteLine ((sum*BOOK_PRICE) + " : סך ההזמנה לתלמיד ");
    }
}
```

```
//-----  
// טענת כניסה: מערך הספרים.  
// טענת יציאה: הפעולה מדפיסה לכל ספר את מספר ההזמנות שבוצעו עבורו.
```

```
public static void booksOrdersShow (int [] books)
```

```
{  
    int i;  
    Console.WriteLine (" פירוט ההזמנות לבית הספר: ");  
    Console.WriteLine "מספר קטלוגי כמות ספרים";  
    for (i = 0 ; i < BOOK_NUM ; i++)  
        Console.WriteLine (books[i] + " " + i);  
}
```

```
//-----  
// טענת כניסה: מערך הספרים.  
// טענת יציאה: מוחזר סכום איברי המערך מוכפל במחיר הספר (28 ש"ח).
```

```
public static int booksTotalOrder (int [] books)
```

```
{  
    int i, sum = 0; // צובר לסכום המערך = sum  
    for (i = 0 ; i < BOOK_NUM ; i++)  
        sum = sum + books [i];  
    return (sum * BOOK_PRICE) ;  
}  
}
```