

תאריך: \_\_\_\_ . \_\_\_\_ . \_\_\_\_

מקצוע: מדעי המחשב ב' - JAVA

שם: \_\_\_\_\_

נושא: רקורסיה



## דף עבודה מס' 1 – רקורסיה – פעולות שלא מחזירות ערך

\*\*\*\*\*

1. לפניך פעולה בשפת JAVA:

```
public static void series1 ( int n )
{
    //
    //
    if ( n >= 10 )
        System.out.print ( n ) ;
    else
    {
        System.out.print ( n + " " ) ;
        series1 ( n + 1 ) ;
    }
}
```

הפעולה מקבלת מספר טבעי n .  
הפעולה מחשבת .....

שורת הדפסה ↩

לפניך תכנית ראשית בשפת JAVA:

```
public static void main ( String [ ] args )
{
    series1 ( 1 ) ;
}
```

א. נסה /י לעקוב אחר ביצוע התכנית הראשית, רשום/י מהו הפלט שיתקבל לדעתך:

ב. כעת בדק/י בעזרת הרצה במחשב מהו הפלט ? \_\_\_\_\_

ג. הרץ /י את התכנית עבור הקלטים הבאים, ורשום /י את הפלט בטבלה הבאה:

mis2	פלט
<b>series1</b> ( 2 ) ;	
<b>series1</b> ( 7 ) ;	
<b>series1</b> ( 10 ) ;	
<b>series1</b> ( 11 ) ;	

ד. בתכנית שלעיל מופיעה דרך חדשה.  
נסה/י להסביר במילים את הדרך וכיצד היא מתבצעת.



## 2. לפניך פעולה `series2` בשפת JAVA :

שים לב! לשינוי ...

```
public static void series2 ( int n )
{
    if ( n >= 10 )
        System.out.print ( n ) ;
    else
    {
        series2 ( n + 1 ) ;
        System.out.print ( n + " " ) ;
    }
}
```

שורת הדפסה ↩

א. נסה /י לעקוב אחר ביצוע התכנית הראשית מתרגיל 1, רשום/י מהו הפלט שיתקבל לדעתך :

ב. כעת בדק/י בעזרת הרצה במחשב מהו הפלט ?

ג. מדוע, לדעתך, הגיב המחשב בצורה זו ?

ד. הרץ /י את התכנית עבור הקלטים הבאים, ורשום /י את הפלט בטבלה הבאה :

mis2	פלט
<b>series2</b> ( 2 ) ;	
<b>series2</b> ( 7 ) ;	
<b>series2</b> ( 10 ) ;	
<b>series2</b> ( 11 ) ;	

ה. מה יקרה אם נשנה את התנאי בפעולה שלפניך ל-  $( n = 10 )$  ונריץ את התכנית הראשית עם הזימון : **series2** ( 11 ) ?

הסבר/י :



## הגדרה:



\*\*\*\*\*  
\* רקורסיה היא דרך חדשה לפתרון בעיות. משימה שקוראת לעצמה. \*  
\*\*\*\*\*

### תנאים לקיום רקורסיה:

1. הליך **רקורסיבי** - תת-משימה המזמנת אותה תת-משימה, עבור ערך קטן/גדול יותר.
2. **תנאי עצירה** – עוצר את פעולת הרקורסיה (היעדרו של תנאי עצירה יוביל לחזרה אינסופית של הרקורסיה).

הרקורסיה מייצרת שני מצבים:

1. שלב **פתיחת** הרקורסיה – הפעולות המתבצעות **לפני** הזימון הרקורסיבי (עד תנאי העצירה).  
הערכים מעובדים בסדר רגיל.
2. שלב **סגירת** הרקורסיה – הפעולות המתבצעות **לאחר** הזימון הרקורסיבי.  
הערכים מעובדים בסדר הפוך.

### דוגמה לתרגיל:

```
public static void series ( int n )  
{  
    if ( n >= 7 )           ⇐ תנאי עצירה  
        System.out.print ( n ) ;  
    else  
    {  
        System.out.print ( n + " " ) ;           ⇐ שלב פתיחת הרקורסיה  
                                                    הערכים מעובדים בסדר רגיל  
        series ( n + 1 ) ;           ⇐ זימון/קריאה רקורסיבית  
        System.out.print ( n + " " ) ;           ⇐ שלב סגירת הרקורסיה  
                                                    הערכים מעובדים בסדר הפוך  
    }  
}
```

⇨ עבור הקלט:  $n = 1$ , ייתקבל הפלט:   
1 2 3 4 5 6 7 6 5 4 3 2 1  
          פתיחה          סגירה



3. לפניך פעולה בשפת JAVA :

```
public static void printSeries1 ( int n )
{
    //
    //
    if ( n >= 50 )
        System.out.print ( n ) ;
    else
    {
        System.out.print ( n + " " ) ;
        printSeries1 ( n + 5 ) ;
        System.out.print ( n * 2 + " " ) ;
    }
}
```

הפעולה מקבלת מספר שלם  $n$  .  
הפעולה מחשבת .....

לפניך זימון הפעולה בתכנית ראשית : **printSeries1** ( 10 ) ;

א. עקוב/י אחר ביצוע הפעולה, רשום/י מהו הפלט שיתקבל לדעתך :

ב. כעת בדק/י בעזרת הרצה במחשב מהו הפלט ?

4. לפניך פעולה בשפת JAVA :

```
public static void printSeries2 ( int n )
{
    //
    //
    if ( n > 0 )
    {
        System.out.print ( n + " " ) ;
        printSeries2 ( n - 10 ) ;
        System.out.print ( ( - n ) + " " ) ;
    }
}
```

הפעולה מקבלת מספר שלם  $n$  .  
הפעולה מחשבת .....

לפניך זימון הפעולה בתכנית ראשית : **printSeries2** ( 40 ) ;

א. עקוב/י אחר ביצוע הפעולה, רשום/י מהו הפלט שיתקבל לדעתך :

ב. כעת בדק/י בעזרת הרצה במחשב מהו הפלט ?

ג. מה ההבדל בין שתי הסדרות שהודפסו בתרגילים 3 ו-4 ? (ציין/י שני הבדלים)





5. לפניך פעולה בשפת JAVA :

```
public static void printSeries3( int n )
{
    // הפעולה מקבלת מספר שלם n .
    // הפעולה מחשבת ומדפיסה את הסדרה : 3 , 6 , 9 , 12 , 120 , 90 , 60 , 30
    if ( n < 5 )
    {
        System.out.print ( _____ ) ;
        printSeries3 ( n + 1 ) ;
        System.out.print ( _____ ) ;
    }
}
```

לפניך זימון הפעולה בתכנית ראשית : **printSeries3** ( 1 ) ;  
השלם/י את הקווים החסרים כך שהפעולה תבצע את משימתה.

6. לפניך פעולה בשפת JAVA :

```
public static void printSeries4( int n )
{
    // הפעולה מקבלת מספר שלם n .
    // הפעולה מחשבת ומדפיסה את הסדרה : 20 , 40 , 60 , 80 , 100 , 90 , 70 , 50 , 30
    if ( _____ )
        System.out.print ( n ) ;
    else
    {
        System.out.print ( ( n ) + " " ) ;
        _____ ;
        System.out.print ( ( n + 10 ) + " " ) ;
    }
}
```

לפניך זימון הפעולה בתכנית ראשית : **printSeries4** ( 20 ) ;  
השלם/י את הקווים החסרים כך שהפעולה תבצע את משימתה.

7. כתב/י פעולה רקורסיבית לכל אחת מהסדרות הבאות :

- א. 1 , 4 , 7 , 10 , 13 , 16 , 19 , 22 (קיימים 2 פתרונות שונים)  
ב. -10 , -9 , -8 , -7 , -6 , -5 , -4 , -3 , -2 , -1 (קיימים 2 פתרונות שונים)  
ג. 1 , 1 , 2 , 2 , 3 , 3 , 4 , 4 , 3 , 3 , 2 , 2 , 1 , 1  
ד. 10 , 20 , 30 , 40 , 4 , 3 , 2 , 1  
ה. א , ב , ג , ד , ג , ב , א  
ו. 1 , 2 , 3 , 4 , 5 , 6 , 7 , 8 , 9 , 8 , 6 , 4 , 2



8. נתונות 2 פעולות והזימון שלהן בתכנית הראשית:

<pre>public static void <b>seriesA</b> ( int n ) {     if ( n &lt;= 1 )         System.out.print ( n + " " );     else     {         System.out.print ( n + " " );         <b>seriesA</b> ( n - 1 );     } }</pre>	<pre>public static void <b>seriesB</b> ( int n ) {     if ( n &gt;= 4 )         System.out.print ( n + " " );     else     {         <b>seriesB</b> ( n + 1 );         System.out.print ( n + " " );     } }</pre>
הזימון <b>seriesA</b> ( 4 );	הזימון <b>seriesB</b> ( 1 );

א. הסברי/י ב- 2-3 שורות מה מבצעת הפעולה **seriesA**:

---



---



---

ב. האם הפעולות הנ"ל יציגו פלט זהה ?

ג. הסברי/י את ההבדל בין הפעולות הנ"ל:

---



---



---

\* 9. חזרי/י לתרגיל 7 סעיפים א' ו- ב' והוסף/י את הפתרון השני לסדרה.

10. פעולה רקורסיבית שמזמנת פעולה רקורסיבית:

א. כתבי/י פעולה רקורסיבית בשם **printStars** המקבלת מספר שלם N ומדפיסה N כוכביות על המסך (כתבי/י פעולה במחלקה).

ב. כתבי/י פעולה רקורסיבית בשם **printRowStars** המקבלת מספר שלם M ומדפיסה N שורות של כוכביות על המסך, כאשר מספר הכוכביות בכל שורה גדל ב- 1.

שימו/י לב ! יש להשתמש בפעולה שכתבת בסעיף א'.

ג. כתבי/י תכנית ראשית הקולטת מספר שלם N ומזמנת את הפעולה **printRowStars**.

דוגמה: עבור הקלט N = 6, ייתקבל הפלט:


```
*
**
***
****
*****
*****
```




פתרון לתרגיל 6 :

<pre>public static void <b>printStars</b> ( int n ) {     if ( n &lt; 1 )         System.out.println ( ) ;     else     {         System.out.print ( ' * ' ) ;         <b>printStars</b> ( n - 1 ) ;     } }</pre>	<pre>public static void <b>printRowStars</b> ( int m ) {     if ( m &lt; 1 )         System.out.print ( m + " " ) ;     else     {         <b>printRowStars</b> ( m - 1 ) ;         <b>printStars</b> ( m ) ;     } }</pre>
--	---

**11.** כתבי/ פעולה רקורסיבית בשם **multiplicationTable** המדפיסה את לוח הכפל.

רמז : רצוי לחלק את המשימה לתת-משימות. 

**12.** נתונות 2 פעולות (אחת רקורסיבית) והזימון של הפעולה **printA** בתכנית הראשית :

<pre>public static void <b>print</b> ( int[ ] A , int i ) {     if ( i &gt;= A.length )         System.out.println ( " &lt;-- המערך " ) ;     else     {         System.out.print ( A[ i ] + " " ) ;         <b>print</b> ( A , i + 1 ) ;     } }</pre>	<p> <b>שים לב !</b>  <u>אסור</u> לזמן את הפעולה  הרקורסיבית עם ++ i  כי זו רקורסיה <u>אינסופית</u>.</p>
<pre>public static void <b>printA</b> ( int[ ] A ) {     <b>print</b> ( A , 0 ) ; }</pre>	
<pre>int [ ] a = { 1 , 2 , 3 , 4 , 5 , 6 } ; <b>printA</b> ( a ) ;</pre>	<p>הזימון בתכנית הראשית :</p>

א. מה יהיה הפלט של הזימון ; **print** ( A , 0 ) ?

---



---

ב. הסברי/י ב- 2-3 שורות מה מבצעת הפעולה **print** :

---



---



---

**13.** כתבי/ פתרון רקורסיבי נוסף לפעולה **print** (מתרגיל 12).



14. נתונות 2 פעולות והזימון שלהן בתכנית הראשית:

```
public static void printString1 ( String S , int L , int H )
{
    if ( L > H )
        System.out.print ( " " );
    else
        if ( L == H )
            System.out.print ( S.charAt( L ) + " " );
        else
        {
            System.out.print ( S.charAt( L ) + " " + S.charAt( H ) + " " );
            printString1 ( S , L + 1 , H - 1 );
        }
}
```

👉 שימי לב !

אסור לזמן את השיטה הרקורסיבית עם ++ L או H - ,  
כי זו רקורסיה אינסופית.

String s = "123456" ;

הזימון בתכנית הראשית:

**printString1** ( s , 0 , s.length() - 1 ) ;

```
public static void printString2 ( String S )
{
    if ( S.length() == 0 )
        System.out.print ( " " );
    else
        if ( S.length() == 1 )
            System.out.print ( S.charAt( 0 ) + " " );
        else
        {
            System.out.print ( S.charAt( 0 ) + " " + S.charAt( S.length() - 1 ) + " " );
            printString2 ( S.substring( 1 , S.length() - 1 ) );
        }
}
```

String s = "123456" ;

הזימון בתכנית הראשית:

**printString2** ( s ) ;

א. מה יהיה הפלט של הזימון ? **printString1**( s , 0 , s.length() - 1 ) ;

\_\_\_\_\_

ב. הסברי/י ב- 2-3 שורות מה מבצעת כל אחת מהפעולות הנ"ל:

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_

ג. בחרי את הדרך המועדפת עליך, נמקי/י תשובתך

\_\_\_\_\_

\_\_\_\_\_

\_\_\_\_\_





15. לפניך פעולה בשפת JAVA :

```
public static void printM ( int [ ] [ ] mat , int i , int j )
{
    if ( i >= mat.length )
        System.out.println ( ) ;
    else
        if ( j >= mat[ i ].length )
        {
            System.out.println ( ) ;
            printM ( mat , i + 1 , 0 ) ;
        }
        else
        {
            System.out.print ( mat[ i ][ j ] + " " ) ;
            printM ( mat , i , j + 1 ) ;
        }
}
```

10	20	30
40	50	60
70	80	90
100	110	120

mat

א. מה יהיה הפלט של הזימון ? `printM ( mat , 0 , 0 )` ;

---

---

---

---

---

ב. הסבר/י ב- 2-3 שורות מה מבצעת הפעולה `printM` :

---

---

---



עבודה נעימה !

[ JWork\_Recursion1 ]