

מאגר מעבדות לשפות התכנות החדשות

ניתן להשתמש בחומרים לצורך הוראה בלבד.

לא ניתן לפרסם את החומרים או לעשות בהם כל שימוש מסחרי

ללא קבלת אישור מראש מצוות הפיתוח בראשות ד"ר תמר פז.

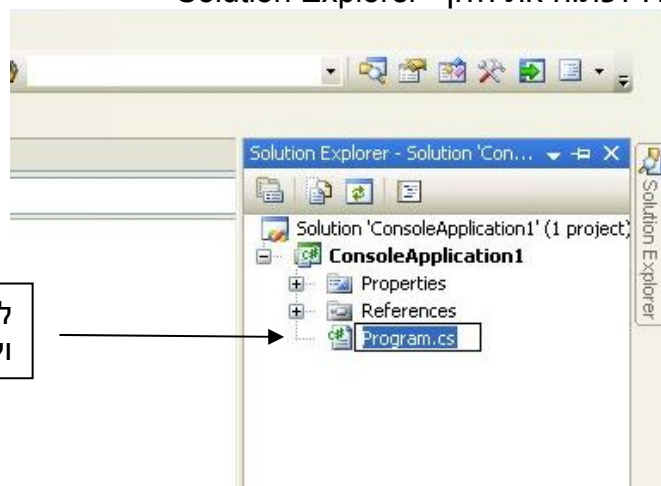
המעבדה בקובץ זה מיועדת לתלמידים הלומדים מדעי המחשב בשפת התכנות C# והיא מותאמת לסביבת Visual C# 2005 Express . המעבדה מיועדת לשיעורי המעבדה והיא מבוססת על שיטת ההוראה לפיה הלימוד של כל נושא חדש יפתח בהתנסות אישית במעבדה. לאחריה, יבוא דיון כיתתי, שבעקבותיו ייפתרו משימות שונות.

המעבדה מתרכזת בלימוד נושא מרכזי אחד: **פעולות (שיטות).**
כדאי לבקש מהתלמידים להוסיף טענות כניסה ויציאה לכל פעולה.

בשפת C# מיוצרת באופן אוטומטי המחלקה class Program .
אם רוצים לשנות את שם המחלקה לשם שונה - יש לבצע זאת כך:

1. לפתוח את חלון Solution Explorer

לסמן את הקובץ program.cs
ולבצע rename לשם המבוקש



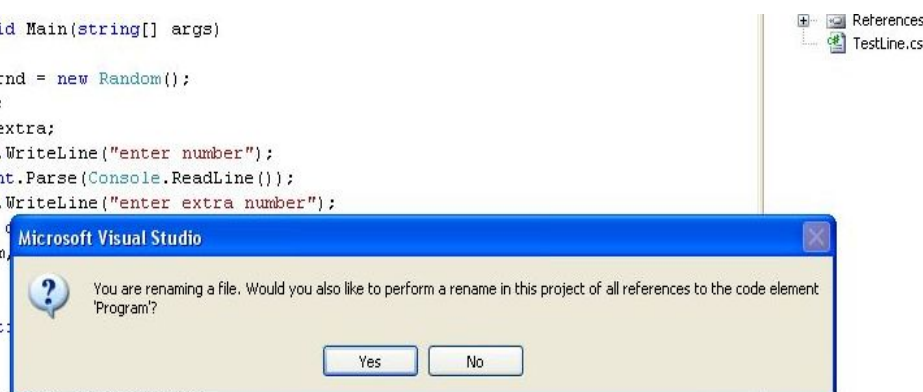
2. לאשר את השינוי

```
void Main(string[] args)
```

```
    rnd = new Random();  
    n;  
    extra;  
    .WriteLine("enter number");  
    int.Parse(Console.ReadLine());  
    .WriteLine("enter extra number");  
    =  
    m;
```

```
at:
```

```
=
```



פעולות (שיטות)

הכרנו בעבר פעולות שמוגדרות בשפה. כמו למשל, `abs`, `sqrt`.
במעבדה זו נלמד להעשיר את השפה ולהוסיף לה פעולות כרצוננו.

```
class Program
{
    private static double Max(double a, double b)
    {
        double maximum;
        if (a > b)
            maximum = a;
        else
            maximum = b;
        return maximum;
    }

    static void Main(string[] args)
    {
        double n1, n2, result;
        Console.WriteLine("enter two numbers ");
        n1 = double.Parse(Console.ReadLine());
        n2 = double.Parse(Console.ReadLine());
        result = Max(n1, n2);
        Console.WriteLine("maximum is " + result);
    }
}
```

משימה 1 – חלק א'

הקלידו והריצו את המחלקה הבאה.
כאשר תתבקשו, הקלידו שני מספרים כרצונכם.
מומלץ ללחוץ על מקש Enter אחרי כל מספר.

השלימו:

- הפעולה הראשית (Main) קולטת מהמשתמש

_____ מספרים ומציגה את _____

- במחלקה מוגדרת פעולה נוספת בשם `Max`.

הפעולה `Max` מקבלת שני נתונים מטיפוס

מספר ממשי ומחזירה את _____

- `private static double Max (double a , double b)` היא החתימה של הפעולה `Max`.

- החלק `(double a , double b)` מסמן כי בעת ההפעלה של הפעולה יש לתת לה _____

ערכים מטיפוס `double`

- המילה `double` לפני שם הפעולה `Max` מסמנת כי הפעולה תחזיר ערך מטיפוס _____

- המילה `private` מציינת כי הפעולה `Max` היא פעולה פרטית של המחלקה `Program` ואי אפשר

לזמן אותה ממחלקות אחרות. (אם לא מציינים זאת - הפעולה תחשב `public`)

- במשמעות המילה `static` נדון מאוחר יותר.

- ההוראה `return maximum` היא הוראת החזרה. תפקידה לדאוג שהפעולה תחזיר ערך.

- בכל פעולה מומלץ לתת הוראת החזרה:

return המונח על-ידי הפעולה ;

משימה 1 – חלק ב'

```
class Program
{
    private static double Max(double a, double b)
    {
        double maximum;
        if (a > b)
            maximum = a;
        else
            maximum = b;
        return maximum;
    }

    static void Main(string[] args)
    {
        double n1 , n2 , result;
        Console.WriteLine("enter two numbers ");
        n1 = double.Parse(Console.ReadLine());
        n2 = double.Parse(Console.ReadLine());
        result = Max (n1,n2);
        Console.WriteLine("the maximum is " + result);
    }
}
```

a ו-b הם הנתונים עליהם פועלת הפעולה. נתונים כאלה נקראים **פרמטרים פורמליים** משתמשים בהם רק בהגדרת הפעולה והפעולה הראשית Main לא מכירה אותם!!

כאשר קוראים לפעולה (מזמנים את הפעולה) משתמשים בנתונים שהפעולה הקוראת (הפעולה המזמנת) מכירה. הנתונים הללו נקראים **פרמטרים אקטואליים** (כי הם הערכים האקטואליים כרגע!). במקרה שלנו הערך האקטואלי של הפרמטר a הוא הערך של המשתנה _____, והערך האקטואלי של הפרמטר _____ הוא הערך של המשתנה n2

הקריאה (הזימון) לפעולה היא כמו הזימון לפעולות הקיימות בשפה: **(נתונים) ש הפעולה** ובמקרה שלנו, הזימון לפעולה הוא באמצעות ההוראה: _____ ניתן לקרוא לפעולה גם במקרים אחרים. כמו, למשל, בהוראת תנאי. נראה דוגמאות כאלה בהמשך.

נבדוק כעת אילו ערכים יכולים לשמש כפרמטרים אקטואליים. בעמודה הימנית של הטבלה הבאה מופיעות 5 הוראות. הקלידו **בנפרד** כל אחת מהן במקום השורה המודגשת בפעולה הראשית, הריצו את המחלקה ומלאו את הטבלה. שימו לב: **חלק מההוראות הן שגויות!**

הסבר שלכם	פלט של הפעולה	ערכים אקטואליים		הוראה לביצוע	
		a	b		
				result = Max (n1 , 100);	1
				result = Max (n2 , n1);	2
				result = Max (75 , 100);	3
				result = Max (maximum, n2);	4
				result = Max (a , b);	5

מסקנה: (מחקו את המיותר)

פרמטר אקטואלי יכול להיות:

משתנה של הפעולה הקוראת (הפעולה המזמנת).

משתנה שהוגדר בתוך הפעולה המזומנת (כמו maximum).

ערך שאינו משתנה (למשל 999).

משימה 2 – חלק א'

מכיוון שפעולה מחזירה ערך, כאשר מזמנים פעולה (קוראים לפעולה) צריך להגיד מה לעשות עם הערך המוחזר. **שנו את הפעולה הראשית בלבד** כך שתתקבל הפעולה הבאה.

- שימרו, הריצו והשלימו:

```
static void Main(string[] args)
{
    double n1 , n2 , result;
    Console.WriteLine("enter two numbers ");
    n1 = double.Parse(Console.ReadLine());
    n2 = double.Parse(Console.ReadLine());
    while (Max (n1,n2)> 0)
    {
        Console.WriteLine("enter two numbers ");
        n1 = double.Parse(Console.ReadLine());
        n2 = double.Parse(Console.ReadLine());
    }
    Console.WriteLine("max is " + Max(n1, n2));
    Console.ReadLine();
}
```

ניתן לבצע עם הערך המוחזר על-ידי פעולה, כל מה שניתן לבצע עם משתנה מטיפוס המוחזר על-ידי הפעולה. למשל, אפשר להציב אותו במשתנה

אחר, אפשר _____, ואפשר לבדוק את ערכו _____

משימה 2 – חלק ב'

הקלידו בנפרד כל אחת משתי ההוראות הבאות במקום הקטע המודגש.

שימרו, הריצו ומלאו את הטבלה. שימו לב: **שתי ההוראות שגויות!**

הוראה לביצוע	הוראה שגויה – הסבר שלכם
return Max (n1 , n2);	
Max (n1,n2) = result;	

משימה 3

```

class Program
{
    static bool Between (double a , double b , double c)
    {
        return (a < b _____ b < c);
    }

    static void Main(string[] args)
    {
        double n1 , n2 , n3;
        int count = 0;
        for (int i=1; _____; _____)
        {
            Console.WriteLine("enter three numbers");
            n1 = _____;
            n2 = double.Parse(Console.ReadLine());
            n3 = double.Parse(Console.ReadLine());
            if (Between (n1 , n2 , n3)) count++;
        }
        Console.WriteLine("there was " + count + " order three");
        Console.ReadLine();
    }
}

```

לפניכם שלד של מחלקה שהפעולה הראשית בה קולטת 5 שלשות של מספרים ומונה את מספר השלשות "המסודרות". שלשה מסודרת היא שלשה שבה המספר הראשון הוא הקטן ביותר והמספר האחרון הוא הגדול ביותר. למשל, השלשה 3 7 9 היא שלשה מסודרת, והשלשה 8 10 9 איננה שלשה מסודרת.

- השלימו את המחלקה.
- הריצו ובידקו שהתקבל הפלט הרצוי והשלימו:

הפעולה **Between מקבלת**

שלושה ערכים מטיפוס

הפעולה **מחזירה** ערך

מטיפוס

הערך שפעולה מחזירה יכול להיות מכל אחד מהטיפוסים שהכרנו בשפה: `int` , `double` , `char` , `boolean`

במחלקה האחרונה, הפעולה `Between` מזמנת (נקראת) _____ פעמים.

בכל המקרים, הזימון לפעולה `Between` נעשה באמצעות ההוראה _____.

השם של הפרמטר הפורמלי הראשון הוא `a` ונכנס אליו הערך של המשתנה _____.

השם של הפרמטר הפורמלי השני הוא _____ ונכנס אליו הערך של המשתנה _____.

השם של הפרמטר הפורמלי השלישי הוא _____ ונכנס אליו הערך של המשתנה _____.

• הסבירו מדוע ההוראה `if (Between (n1 , n2 , n3)) count++;` היא הוראה תקינה _____

משימה 4

א. הפעולה Multi אמורה לקבל שני מספרים חיוביים ושלמים ולהחזיר את תוצאת הכפל שלהם בלי להשתמש בפעולת הכפל (*). בפעולה נפלה שגיאה אחת. תקנו אותה.

```
private static int Multi (int a , int b)
{
    int i , result;
    result = 0;
    for (i= 1; i<=a; i++)
        result = result + a;
    return result;
}
```

i - result הם משתנים
מקומיים רק של הפעולה Multi

אפשר להצהיר בפעולה על משתנים שיהיו רק שלה.

משתנים אלו אינם מוכרים על-ידי הפעולה הראשית והם נקראים משתנים מקומיים של הפעולה!

ב. כתבו פעולה ראשית שקולטת מספרים חיוביים ושלמים (לפחות מספר אחד), ופולטת את מכפלתם. קליטת הנתונים תיפסק עם קליטת המספר 0 (מספר זה לא יוכפל במספרים האחרים). למשל אם ייקלטו המספרים (משמאל לימין): 0 1 6 2 3 , הפעולה תפלוט 36.

השתמשו בפעולה Multi

מומלץ ללחוץ על מקש Enter אחרי כל מספר.

משימה 5

א. הפעולה Code מקבלת שני מספרים שלמים ומחזירה תו. אם המספר הראשון שהתקבל גדול מהשני, הפעולה תחזיר את התו 'ק', אם הראשון קטן מהשני, הפעולה תחזיר 'ח', ואם הם שווים, יוחזר התו 'z'.

ב. כתבו פעולה ראשית שמגרילה 8 זוגות של מספרים בתחום שבין -100 לבין 100. הפעולה תפלוט שורה של תווים בהתאם למספרים שהוגרלו.

למשל אם המספרים שהוגרלו הם (משמאל

לימין): 12 12 -99 9 8 -9 0 90 100 -8 7 -56 56 0 12 אז הפעולה תפלוט:
pznnpnpz

זיכרו: יש חשיבות לסדר הופעת הפרמטרים האקטואליים.

משימה 6

כתבו פעולה ראשית שמציגה עבור כל אחד מהמספרים 1-5 את העצרת שלו.
הפלט יראה כך:

```
1 factorial is 1
2 factorial is 2
3 factorial is 6
4 factorial is 24
5 factorial is 120
```

כתבו תחילה פעולה fact שמקבלת מספר ומחזירה את העצרת שלו.

משימה 7

כתבו פעולה ראשית שקולטת מספרים חיוביים שלמים ומציגה עבור כל מספר את סכום ספרותיו ואת מספר ספרותיו. קליטת הנתונים תפסק כאשר ייקלט מספר שלילי.
כתבו תחילה שתי פעולות:
sumDigit שמקבלת מספר ומחזירה את סכום ספרותיו.
numDigit שמקבלת מספר ומחזירה את מספר ספרותיו.

משימה 8

בכל המחלקות שכתבנו עד כה, היתה רק הוראת זימון אחת לפעולה (שבחלק מהמקרים, התבצעה מספר פעמים) והפעולה שימשה לחלוקת לתת משימות. בבעיה הבאה נגדיר פעולה ונזמן אותה ממקומות שונים בפעולה הראשית.

תלמידי שכבת י' יצאו לטיול שנתי. נכתוב פעולה שתקלוט את מספר התלמידים בשכבה ותציג את מספר האוטובוסים שיש להזמין (בכל אוטובוס ניתן להסיע לכל היותר 50 תלמידים), ואת מספר החדרים שיש

```
private static int NumBus (int num)
{
    int order ;
    if (num % 50 == 0 )
        order = _____
    else
        order = (num / 50) + 1;
    return order;
}
```

להזמין באכסניה (בכל חדר יכולים ללון 6 תלמידים לכל היותר).
א. הפעולה NumBus מקבלת את מספר התלמידים ומחזירה את מספר האוטובוסים הדרוש.
• השלימו את הפעולה.

```
private static int NumRooms (int num)
{
}
}
```

ב. הפעולה NumRooms תקבל את מספר התלמידים ותחזיר את מספר החדרים הדרוש. השלימו את הפעולה.

ג. הפעולות NumBus ו- NumRooms שונות זו מזו בגורם אחד (מלבד שמן כמובן). מהו? _____

```
private static int NumItem (int num , int capacity)
{
  int order ;
  if (num % capacity == _____ )
    order = (num / capacity);
  else
    order = _____ ;
  return order;
}
```

ד. הפעולה NumItem היא פעולה כללית שמקבלת מספר התלמידים וקיבולת (50 במקרה של אוטובוסים, ו-6 במקרה של חדרים). הפעולה מחזירה את מספר הפריטים הדרוש. השלימו את הפעולה.

ה. כתבו כעת את המחלקה המלאה (השתמשו בפעולה NumItem). שימרו, הריצו ובדקו כי התקבלו התוצאות הרצויות.

ו. שנו את המחלקה כך שבנוסף למספר התלמידים בשכבה, המחלקה גם תקלוט מהמשתמש את מספר המקומות באוטובוס ואת מספר המיטות בחדר.

הפעם הפעולה הראשית צריכה לבצע את המשימות הבאות (את האלגוריתם הבא):

1. לקלוט את מספר התלמידים.
2. לקלוט את מספר המקומות באוטובוס.
3. לקלוט _____
4. לחשב את מספר האוטובוסים הדרוש.
5. לחשב את מספר החדרים הדרוש.
6. להציג כפלט את _____ ואת מספר החדרים.

שימרו, הריצו ובדקו כי התקבלו התוצאות הרצויות.

משימה 9

כתבו מחלקה שקולטת רביעיות של מספרים ומבצעת את הפעולות הבאות:
 אם המספר הראשון ברביעייה הוא 1 אז המחלקה תציג את הממוצע של המספר השני והמספר השלישי.
 אם המספר הראשון ברביעייה הוא 2 אז המחלקה תציג את הממוצע של המספר השני והמספר הרביעי.
 אם המספר הראשון ברביעייה הוא 3 אז המחלקה תציג את הממוצע של המספר השלישי והמספר הרביעי.
 קליטת הנתונים תיפסק כאשר המספר הראשון ברביעייה לא יהיה 1 או 2 או 3

יש להשתמש בפעולה אחת לפחות בנוסף לפעולה הראשית.

משימה 10 – חלק א'

הקלידו, הריצו את המחלקה ורשמו מה היא מבצעת:

```
class Program
{
    private static double Func1 (double a)
    {
        return (a+1);
    }

    private static double Func2 ()
    {
        Random rnd = new Random();
        double x;
        x = rnd.NextDouble();
        return (x + Func1 (x));
    }

    public static void Main(string[] args)
    {
        Console.WriteLine(Func2 ( ));
    }
}
```

שימו לב לזימון המיוחד לפעולה **Func2**

מסקנות (מחקו את המיותר):

1. פעולה *יכולה* / *לא יכולה* לזמן פעולה אחרת שהוגדרה במחלקה.
2. פעולה *חייבת* / *לא חייבת* לקבל פרמטרים.
3. (השלימו): כאשר פעולה לא מקבלת פרמטרים:

- א. בחתימה של הפעולה, במקום רשימת הפרמטרים, רושמים _____
- ב. בעת הזימון לפעולה, צריך להוסיף _____ לאחר שם הפעולה.

השלימו: משמעות ההוראה `rnd.NextDouble()`; היא זימון של הפעולה _____
 פעולה זו *כן* / *לא* (מחקו את המיותר) מקבלת ערכים.
 באופן דומה, משמעות ההוראה `int.Parse(Console.ReadLine());` היא זימון של הפעולה _____
 גם פעולה זו לא מקבלת אף ערך. הפעולה מחזירה מספר שלם שהשתמש הקליד.

משימה 10 – חלק ב'

```

class Program
{
    private static double Func2 ()
    {
        Random rnd = new Random();
        double x;
        x = rnd.NextDouble();
        return (x + Func1 (x));
    }

    private static double Func1 (double a)
    {
        return (a+1);
    }

    public static void Main(string[] args)
    {
        Console.WriteLine(Func2 ( ));
    }
}

```

- שנו את המחלקה כך שהפעולה Func2 תופיע לפני הפעולה Func1.
- הריצו את המחלקה.

מסקנה (השלימו):

חשיבות לסדר הגדרת הפעולות.

כלומר, פעולה יכולה לזמן גם פעולות

שהוגדרו

משימה 11

```

class OneLine
{
    private static void Line (int count, double n)
    {
        int i ;
        for (i=1; i<=count; i++)
            Console.WriteLine(n+ " ");

        Console.WriteLine("");
    }

    public static void main(String[] args)
    {
        int num;
        double extra;

        Console.WriteLine("enter number");
        num = int.Parse(Console.ReadLine());
        Console.WriteLine("enter extra number");
        extra = double.Parse(Console.ReadLine());
        Line (num, extra);
    }
}

```

- המחלקה הבאה קולטת מספר טבעי (num) ומספר ממשי נוסף (extra), ופולטת בשורה אחת num פעמים את המספר הנוסף extra.
- הקלידו והריצו את המחלקה.
- כאשר תתבקשו, הקלידו מספר טבעי ומספר נוסף כרצונכם.

במחלקה מוגדרת פעולה בשם **Line**.

השלימו:

- לפעולה Line יש שני פרמטרים:

count מטיפוס _____

_____ מטיפוס _____

- לפעולה Line יש משתנה מקומי אחד:

_____ מטיפוס _____

שימרו את המחלקה. נזדקק לה במשימה 13.

במה שונה הפעולה Line מהפעולות שהכרנו עד כה (שימו לב לחתימה של הפעולה ולזימון של הפעולה)

מסקנות

1. (מחקו את המיותר): פעולה **חייבת** / **א** **חייבת** להחזיר ערך.

2. (השלימו): כאשר פעולה לא מחזירה ערך:

א. בחתימה של הפעולה, במקום הערך המוחזר, רושמים: _____

ב. בגוף הפעולה אין הוראת _____ ערך.

ג. בעת הזימון לפעולה, לא צריך לעשות כלום עם הערך המוחזר.

פעולה כזו שלא מחזירה ערך נקראת **פרוצדורה**

3. **גם Main היא פעולה!!!**

Main היא פעולה ראשית – הפעולה שמופעלת "באופן אוטומטי" עם הפעלת המחלקה.

החתימה `public static void Main(String[] args)` מציינת כי:

Main היא פעולה ש **א** / **א** **א** מחזירה ערך. (מחקו את המיותר)

הערך שהפעולה main מקבלת הוא `args` מטיפוס `String[]` ועל כך נרחיב בהמשך.

משימה 12

• כתבו מחלקה שקולטת מהמשתמש 3 מספרים כלשהם ומציגה את כל הסכומים האפשריים כולל

הביטויים לחישוב הסכומים. למשל, אם נקלטו המספרים 1 2 3 המחלקה תציג:

$$1 + 2 = 3$$

$$1 + 3 = 4$$

$$2 + 3 = 5$$

יש לשלב במחלקה את הפעולה הבאה ולזמן אותה מספר פעמים:

```
private static void SumTwo (int a , int b)
{
    Console.WriteLine(a + "+" + b + "=" + (a+b));
}
```

• הקלידו, הריצו ובדקו שהתקבל הפלט המבוקש.

משימה 13

```

class OneLine
{
    static void Line (int count, double num)
    {
        int i ;
        for (i=1; i<=count; i++)
            Console.WriteLine(num+ " ");

        Console.WriteLine("");
    }

    public static void Main(string[] args)
    {
        int num;
        double extra;

        Console.WriteLine("enter number");
        num = int.Parse(Console.ReadLine());
        Console.WriteLine("enter extra number");
        extra = double.Parse(Console.ReadLine());
        Line (num, extra);
    }
}

```

נשנה כעת את הפעולה Line ממשימה 11 כך שהשם של הפרמטר יהיה num ולא n שימו לב, אין שום שינוי אחר במחלקה! ובפרט גם במחלקה Main יש משתנה בשם num. מה תהיה תגובת המחשב כאשר נריץ עכשיו את המחלקה? _____

• הריצו את המחלקה

• בדקו מהו הפלט

• הסבירו מדוע התקבל פלט זה? _____

משימה 14

```

class Sod
{
    public static void Main(string[] args)
    {
        a(35);
    }
    private static void a (int number)
    {
        Console.WriteLine("a starts with "+ number);
        b (1000);
        b (number);
        Console.WriteLine("a ends with "+ number);
    }
    private static void b (int number)
    {
        Console.WriteLine("in b number = "+ number);
    }
}

```

• מה יהיה הפלט של המחלקה הבאה? _____

• הקלידו, הריצו ובדקו מהו הפלט _____

• הסבירו מדוע התקבל פלט זה? _____

משימה 15

```

class Sod
{
    public static void Main(string[] args)
    {
        a(35);
    }
    private static void a (int number)
    {
        Console.WriteLine("a starts with "+ number);
        b (number);
        b (mispar);
        Console.WriteLine("a ends with "+ number);
    }
    private static void b (int mispar)
    {
        Console.WriteLine("in b mispar = "+ mispar);
    }
}

```

- מה יהיה הפלט של המחלקה הבאה?

- הקלידו ושימרו.

מופיעה הודעת שגיאה!

הסבירו מדוע התקבלה ההודעה.

משימה 16

```

class avarege
{
    public static void Main(string[] args)
    {
        double a , b, m;
        Console.WriteLine("enter two numbers");
        a = double.Parse(Console.ReadLine());
        b = double.Parse(Console.ReadLine());
        average (a, b , m);
        Console.WriteLine ("the average of "+a+" and "+b+" is "+m);
    }
    private static void average (double a, double b, double m)
    {
        m = (a+b)/ 2;
    }
}

```

- המחלקה הבאה אמורה לקלוט

שני מספרים, לחשב את

הממוצע ולהציג הודעה

מתאימה. המחלקה לא מבצעת

את הנדרש. הסבירו מה

הבעיה.

- תקנו את המחלקה

- שימרו, הריצו ובידקו כי התקבל הפלט המבוקש.

משימה 17 – חלק א'

- כתבו פעולה ראשית שקולטת שני מספרים שלמים בתחום 1-9. הפעולה תציג את ההדפסה הבאה:

88888	עבור הקלט 8 , 5 יודפס:	999	עבור הקלט 9 , 3 יודפס:
8888		99	
888		9	
88			
8			

הערות:

- כתבו מסננת קלט לבדיקה כי המספרים הנקלטים הם בתחום הרצוי.
- כתבו תחילה פעולה שדואגת להדפסת שורה אחת (העזרו במשימה 11).
- **רמז:** הביטוי לשינוי מונה הלולאה צריך להיות --

משימה 17 – חלק ב'

- הוסיפו לפעולה את ההוראות הדרושות כך שיוצגו הציורים הבאים:

88888	עבור הקלט 8 , 5 יודפס:	999	עבור הקלט 9 , 3 יודפס:
8888		99	
888		9	
88		9	
8		99	
8		999	
88			
888			
8888			
88888			

משימה 18

- לקראת פתיחת שנת הלימודים נערך סקר לגבי מחיר "סל ממוצע לתלמיד". הסקר נערך בשלוש רשתות שיווק. נכתוב מחלקה שתקלוט את מחיר הסל בכל אחת מהרשתות ותציג את המחיר הממוצע, ואת ההפרש בין כל אחד מהמחירים לבין המחיר הממוצע.
- ראשית, נחלק לתת משימות (אלגוריתם למחלקה הראשית):
 1. קליטת שלושה מחירים (בפעולה הראשית)
 2. חישוב מחיר ממוצע (פעולה - average)
 3. חישוב והדפסת ההפרש בין כל אחד מהמחירים לבין המחיר הממוצע (פעולה - difference)

- כתבו את המחלקה השלמה.

- הקלידו, שימרו, הריצו ובדקו כי התקבל הפלט הרצוי.

מתי נחליט על כתיבת פעולה שמחזירה ערך ומתי נחליט כי מתאים יותר לכתוב פעולה שלא

מחזירה ערך? (מדוע difference צריכה להיות פעולה שלא מחזירה ערך)? _____

משימה 19

```

1 class Grades
2 {
3     static int highNum , highGrade;
4     public static void Main(string[] args)
5     {
6         int num , grade ;
7         Console.WriteLine("enter num");
8         highNum = int.Parse(Console.ReadLine());
9         Console.WriteLine("enter grade");
10        highGrade = int.Parse(Console.ReadLine());
11        for (int i = 2; i<11; i++)
12        {
13            Console.WriteLine("enter num");
14            num = int.Parse(Console.ReadLine());
15            Console.WriteLine("enter grade");
16            grade= int.Parse(Console.ReadLine());
17            High (num , grade);
18        }
19        Console.WriteLine(highNum + " got the highest grade");
20    }
21
22    private static void High (int num , int grade)
23    {
24        if (grade > highGrade)
25        {
26            highGrade = grade;
27            highNum = num;
28        }
29    }
30 }

```

המחלקה הבאה קולטת 10 מספרי תלמיד וציונים. המחלקה מציגה את המספר של התלמיד בעל הציון הגבוה ביותר (מספור השורות הוא רק לצורך הדיון ואינו חלק מהמחלקה).

המשתנים highNum ו- highGrade מוגדרים בתחילת המחלקה. משתנים כאלה לא שייכים לפעולה מסוימת, אלא שייכים למחלקה כולה!

המילה **static** בשורה 3 מסמנת כי המשתנים המוגדרים בשורה זו שייכים למחלקה כולה.

השלימו:

▪ בשורות 7-10 קולטים את הנתונים של התלמיד הראשון.

את מספר התלמיד מציבים ב-

ואת הציון _____

מציבים ב- _____

▪ בלולאה שמתחילה בשורה 11 ,

קולטים את הנתונים של _____

ומזמנים את הפעולה High

שתפקידה _____

• מדוע highGrade ו- highNum לא יכולים להיות משתנים של הפעולה Main וגם לא יכולים להיות משתנים של הפעולה High ?

• הקלידו, שימרו והריצו. כאשר תתבקשו, הקלידו מספרי תלמיד וציונים כרצונכם.

משתנה גלובלי הוא משתנה שמוגדר בתחילת המחלקה. כל הפעולות מכירות אותו, יכולות

להשתמש בו ויכולות גם לשנות אותו.

הגדרה של משתנה גלובלי: **static משתנה טיפוס המשתנה**

משימה 20

המחלקה שבעמוד הבא מציגה למשתמש 10 תרגילי חשבון. לכל תרגיל כזה, המחלקה מציגה את התרגיל, קולטת תשובה מהמשתמש, ומגיבה בהתאם לתשובה. לאחר הצגת 10 תרגילים, המחלקה שואלת את המשתמש האם ברצונו להמשיך ולתרגל. במידה ותשובת המשתמש חיובית, מוצגים בפניו 10 תרגילים נוספים.

א. פיתחו את המחלקה MathEx, הריצו ובדקו שהיא אכן מבצעת את הנדרש.

ב. השלימו:

• תפקיד הפעולה ten_exe הוא: _____

• הפעולה choose מחזירה את _____.

• הפעולה calculate מגרילה שני מספרים בתחום שבין _____ לבין _____, ומכניסה אותם למשתנים n_1 ו- n_2 . הפעולה מחזירה את תוצאת הפעולה operation עם שני המספרים שהוגרלו.

• הפעולה one_exe מקבלת שני פרמטרים: התוצאה הרצויה וסוג הפעולה המבוקשת. הפעולה מבצעת:

א. מציבה במשתנה _____ +, -, או * בהתאם לסוג הפעולה המבוקש.

ב. מדפיסה את התרגיל.

ג. _____.

ד. בודקת האם היא נכונה.

ה. מגיבה בהתאם.

ג. מדוע המשתנים n_1 ו- n_2 מוכרחים להיות משתנים גלובליים? _____

ד. גם לפעולה הראשית (Main) וגם לפעולה one_exe יש משתנה בשם answer. המשתנים הם מטיפוסים שונים ובכל זאת המחלקה תקינה. מדוע? _____

ה. אילו שינויים יש לערוך במחלקה אם נרצה להפוך את choose לפעולה שלא מחזירה ערך? _____

ו. בצעו במחלקה את השינויים הבאים.

לאחר כל שינוי, הריצו ובדקו את המחלקה.

- כדי למנוע עבודה עם מספרים שליליים, המחלקה תדאג שתמיד n_1 יהיה גדול מ- n_2 .
 - במידה והמשתמש נתן תשובה שגויה לתרגיל שהוצג בפניו, המחלקה תאפשר לו לתת תשובה נוספת. אם שגה גם בפעם השניה, המחלקה תציג הודעה מתאימה ותמשיך הלאה.
 - לפני כל סדרה של עשרה תרגילים, המחלקה תשאל את המשתמש באיזה תחום הוא מעוניין שיהיו המספרים המשתתפים בתרגילים, ותפעל בהתאם לבקשת המשתמש.
- לשם ביצוע שינוי זה, יש להוסיף עוד פעולה!


```

class MathEx
{
static int n1 , n2;
public static void Main(string[] args)
{
    bool answer = true;
    while (answer )
    {
        ten_exe( );
        Console.WriteLine("do you want more
            exercises? Type true or false");
        answer = bool.Parse(Console.ReadLine());
    }
    Console.WriteLine("good bye");
}
private static void ten_exe ()
{
    int result , i, operation;
    for (i=1; i<=10; i++)
    {
        operation = choose();
        result = calculate (operation);
        one_exe (result , operation);
    }
}
private static int choose ()
{
    int operation;
    do {
        Console.WriteLine("choose operation: 1 for '+' 2 for '-' 3 for '*' ");
        operation =int.Parse(Console.ReadLine());
    } while (operation != 1 && operation != 2 && operation != 3);
    return operation;
}
private static int calculate (int operation)
{
    int result;
    Random rnd=new Random();
    n1 = rnd.Next(10); n2 = rnd.Next(10);
    switch (operation)
    { case 1 : result = n1 + n2; break;
      case 2 : result = n1 - n2; break;
      default : result = n1 * n2; break;
    }
    return result;
}
private static void one_exe (int result, int operation)
{
    int answer;
    char op;
    switch (operation)
    { case 1 : op = '+'; break;
      case 2 : op = '-'; break;
      default : op = '*'; break;
    }
    Console.WriteLine("{0}{1}{2}= ? ",n1,op,n2);
    answer= int.Parse(Console.ReadLine());
    if (answer == result)
        Console.WriteLine("excellent");
    else
        Console.WriteLine("wrong");
}
}

```

בזכות החלוקה לתת-פעולות, המחלקה קריאה ומובנת!



לולאת do...while היא "מלכודת קלט"