

מבוא לתכנות

הרצאה 3 – מנגנון Switch ולולאות

While

סמטר 1

Switch Case

- Switch הינו מנגנון לתת לתוכנית לבחור באפשרות אחת מני רבות כמו שראינו בדוגמת הelse-if

```
switch (variable)
{
    case constant1:
        code;
        break;
    case constant2:
        code not allowed here!!!
    case constant3:
        code;
        break;
    default:
        code;
        break;
}
```

המשך

- המשתנה יכול להיות מכל סוג: בוליאני, תו, מחרוזת, מספר שלם
- רק מקרה אחד ייבחר
- אם אף מקרה לא תקף אז תיבחר ברירת המחדל (default)
- ברירת המחדל אינה הכרחית
- לאחר כל מקרה אנו נכתוב: *break;*
- אפשרי לצרף מספר מקרים כל עוד לא כבתבנו אף שורת קוד ביניהם!
- אנו נשתמש ב-*else-if* במקרה והתנאים יותר מורכבים.

Switch Case דוגמה

```
string input = Console.ReadLine();
switch (input)
{
    case "step1":
        Console.WriteLine("running step ONE!");
        break;
    case "step2":
        Console.WriteLine("running step TWO!");
        break;
    case "step3":
    case "step4":
        Console.WriteLine("running step THREE OR FOUR!");
        break;
    default:
        Console.WriteLine("in the default case...");
        break;
}
```

- לחלק מהפקודות יש דרך מקוצרת לכתוב אותן

```
num += 5; //num = num + 5;
num -= 5; //num = num - 5;
num *= 3; //num = num * 3;
num /= 3; //num = num / 3;
num++;    //num = num + 1; done at the end
num--;    //num = num - 1; done at the end
++num;    //num = num + 1; done at the beginning
--num;    //num = num - 1; done at the beginning

num = 1;
Console.WriteLine(num++); //print 1
Console.WriteLine(num);   //print 2
Console.WriteLine(++num); //print 3
Console.WriteLine(num);   //print 3
```

While

- לרוב, עלינו לחזור על פעולה מספר פעמים.
דוגמה פשוטה:

```
string name = "Avi";  
Console.WriteLine("Hello " + name);  
Console.WriteLine("Hello " + name);  
Console.WriteLine("Hello " + name);  
Console.WriteLine("Hello " + name);  
Console.WriteLine("Hello " + name);
```

while(תנאי)

{

בלוק הוראות במידה ותנאי
הלולאה הוא אמת

}

- פתרון אחר:

```
int i = 1;  
while (i <= 5)  
{  
    Console.WriteLine("Hello " + name);  
    i++;  
}
```

הפתרון הנ"ל הינו אלגנטי, קריא, ניתן להרחבה ובטוח יותר.

- ניתן להשתמש במשתנה השליטה:

```
i = 3;
while (i <= 7)
{
    Console.WriteLine(i);
    i++;
}
```

- Output:

```
3
4
5
6
7
```

- מה קורה בקוד הבא?

```
i = 7;
while (i >= 3)
{
    Console.WriteLine(i);
    i--;
}
```

- Output:

```
7
6
5
4
3
```


לולאה אינסופית

```
while (i > 0)
{
    Console.WriteLine(i);
    i++;
}
```

- מה הבעיה עם הקוד הנ"ל?
- אם האיטרציה אמורה להפסיק בהתאם לתנאי כלשהו, עלינו לדאוג שהתנאי יהפוך ל"שקר" בשלב זה או אחר במהלך הלולאה.
- אחרת, הלולאה לא תפסיק לעולם!
- זו נקראת "לולאה אינסופית".
- יצירת לולאות אינסופיות בשוגג זו טעות קידוד נפוצה.

הכרזת break

- אנו יכולים להפסיק את לולאת הwhile באופן מידתי באמצעות הכרזת :break

```
i = 3;
while (i <= 7)
{
    Console.WriteLine(i);
    i++;
    if (i==5)
        break;
}
```

```
i = 3;
while (i <= 7 && i != 5)
{
    Console.WriteLine(i);
    i++;
}
```

- Break גורם לתוכנית להפסיק מידית את הרצת הלולאה ולהמשיך לשורות הקוד הבאות אחריה
- למרות ששימוש בBreak הוא לגיטימי, הוא נחשב ל"מוצא אחרון". אנו נעדיף בדרך כלל לעצור את פעולת הלולאה באמצעות כתיבת קוד אשר הופך את תנאי הלולאה ל"שקר".

הכרזת continue

- אנו נוכל להפסיק רק את האיטרציה (הסיבוב) הנוכחית של הלולאה באופן מידי באמצעות שימוש ב:continue

```
i = 3;
while (i <= 7)
{
    if (i==6)
    {
        i++;
        continue;
    }
    Console.WriteLine(i);
    i++;
}
```

פלט:

3
4
5
7

- continue גורם לתוכנית להפסיק את האיטרציה הנוכחית ולעבור לאיטרציה הבאה.

תוכנית המסורין

```
int sum = 0;
int i = 0;
int n = int.Parse(Console.ReadLine());

while (i < n)
{
    sum += i+1;
    i++;
}
Console.WriteLine(sum);
```

- מה התוכנית הזו עושה?
- מהו n?
- מהו sum?
- נוכל לכתוב אותה בדרכים אחרות ולקבל את אותה התוצאה?

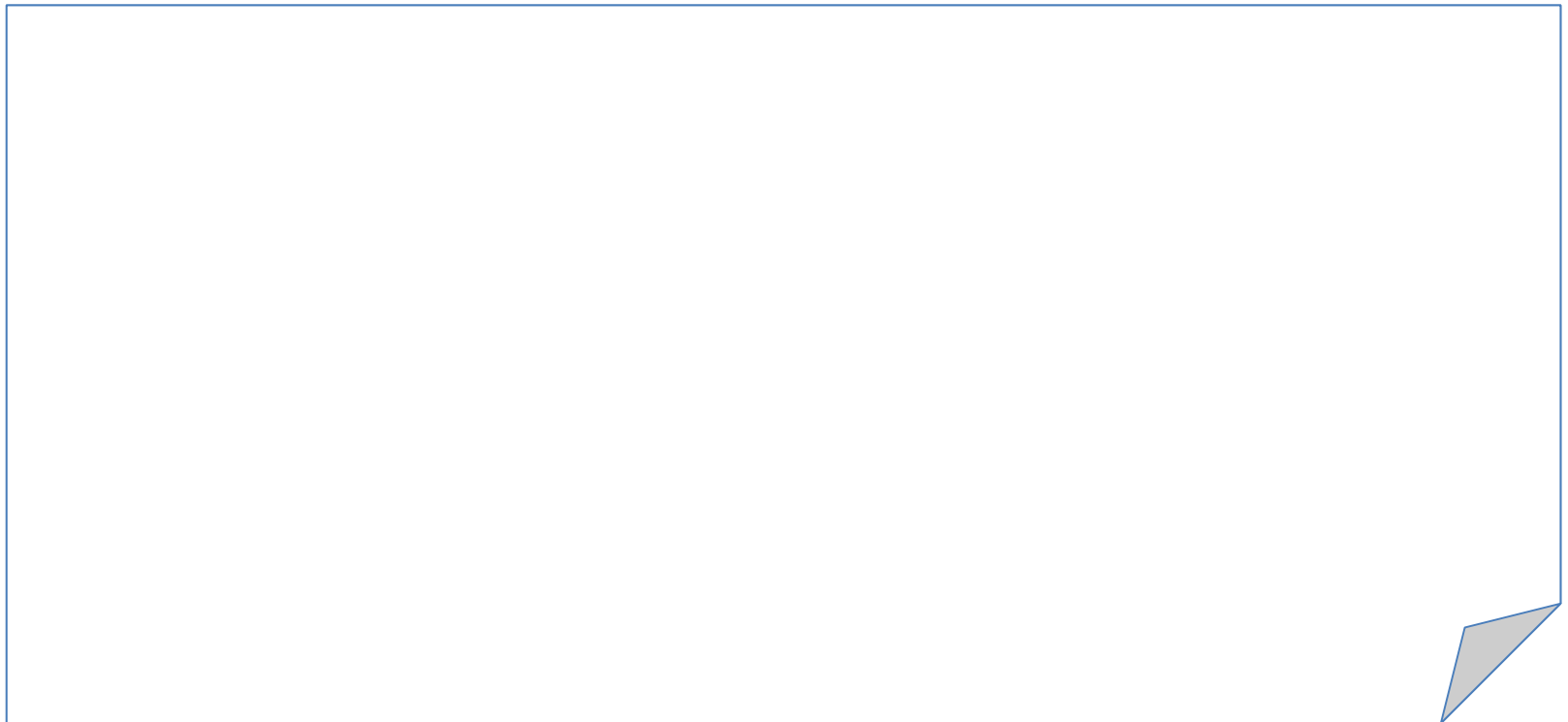
תרגיל 1

- כתבו את התוכנית `factorial_n`
- עצרת - $\text{Factorial}_n = n!$



תרגיל 2

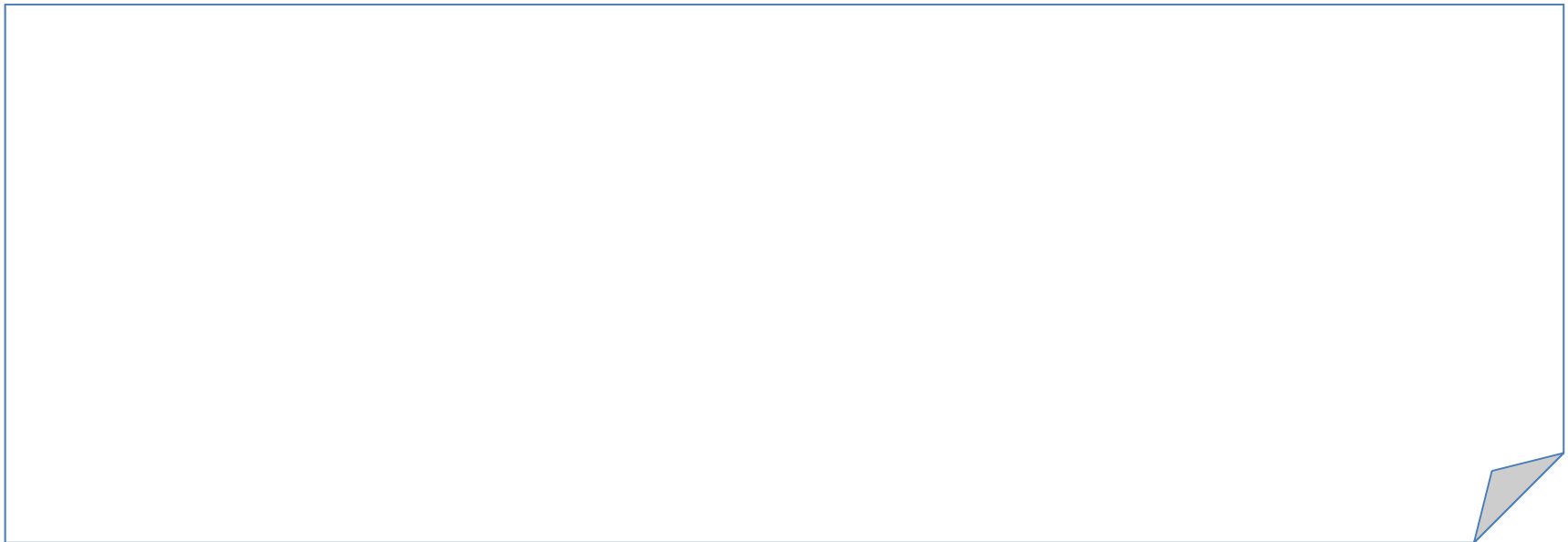
- כתבו את התוכנית 7-בום.
- קראו מספר מקסימלי מהמשתמש והדפיסו את כל המספרים מ1 ועד המקסימום, אך כאשר המספר להדפסה מתחלק ב7, הדפיסו "Boom" במקום.



תרגיל 3

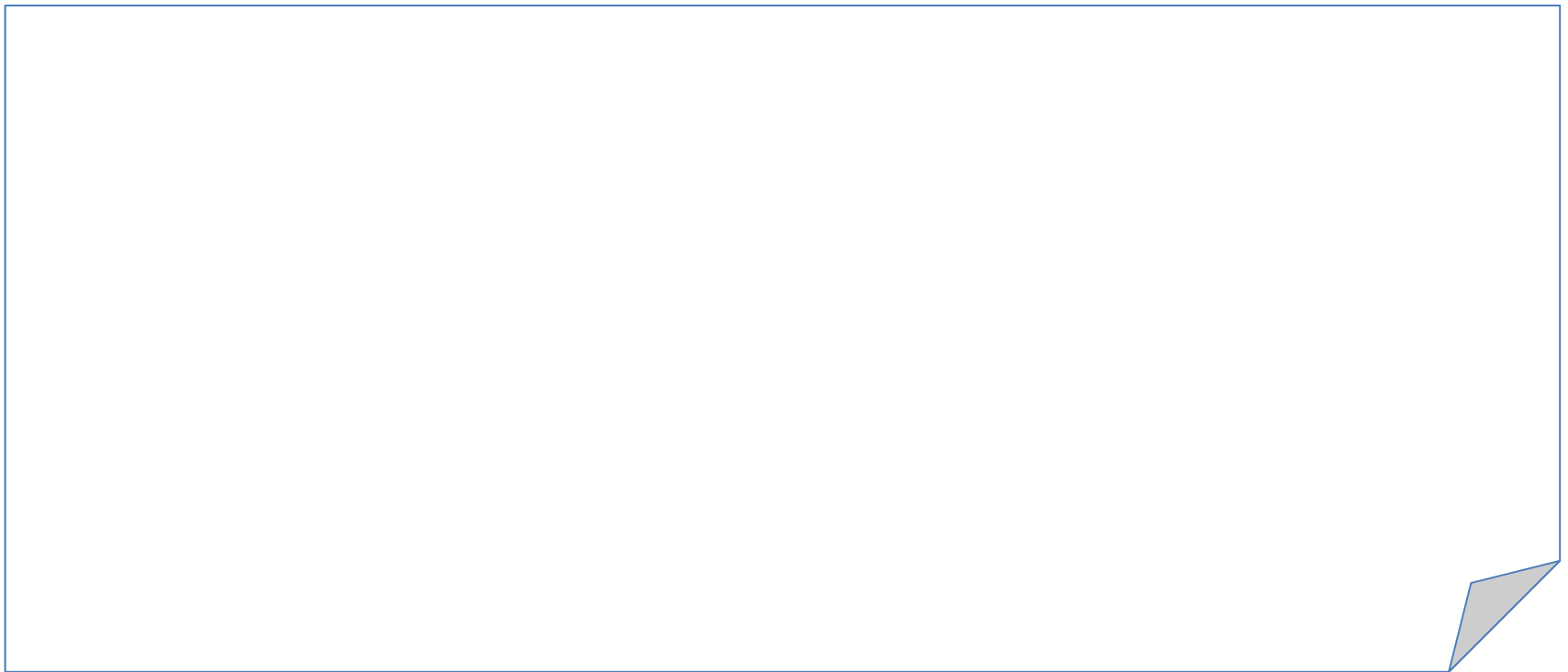
- בהינתן מספר n , הדפיסו את כל המספרים מ-2 ועד 10 אשר מתחלקים ב- n .

– דוגמה: אם $n = 28$, התוצאה תהיה 2, 4, 7



תרגיל 4

- כתבו תוכנית שקוראת מספרים מהמשתמש, עד שהוא רושם את המילה "done".
- ואז, התוכנית מדפיסה את הממוצע של כל המספרים שהוכנסו.



תרגיל 5

- כתבו תוכנית שקוראת מספר מהמשתמש ומחליטה אם הוא ראשוני או לא.

Do while

- משתמשים במקרים שבהם לפחות איטרציה אחת צריכה לצאת לפועל לפני בדיקת התנאי

```
string input;  
do  
{  
    Console.WriteLine("not DONE yet?");  
    input = Console.ReadLine();  
} while (input != "done");
```

```
do  
{  
    כלל הפקודות שיש  
    לבצע אם התנאי נכון  
} while(תנאי);
```