



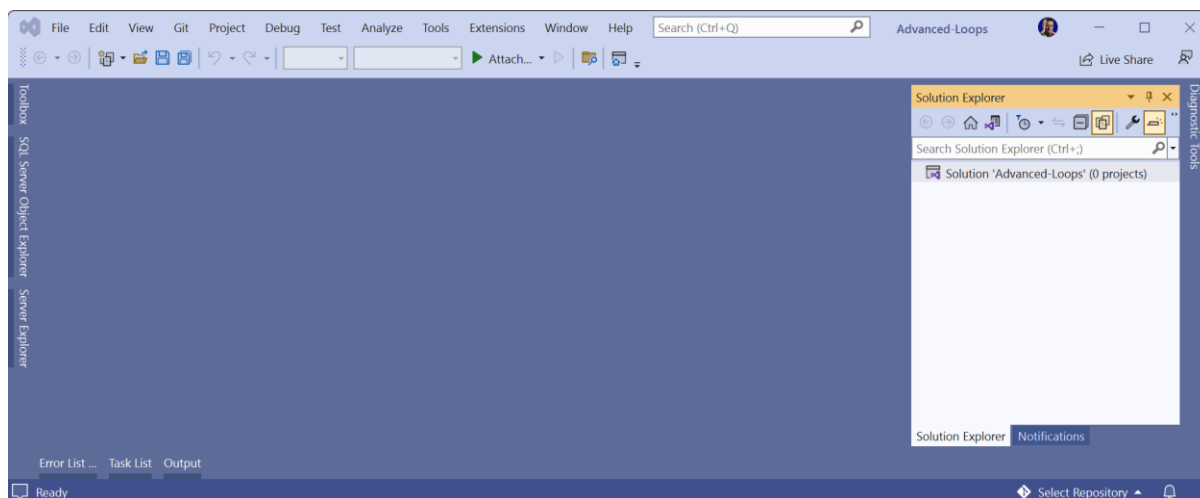
## Упражнения: Повторения с различни стъпки

### 0. Пrazно решение

Създайте празно решение, за да организирате задачите от упражненията.

1. Стартирайте Visual Studio.
2. Създайте нов проект [Create New Project].
3. Търсете и изберете: Blank Solution
4. Именувайте решението: Advanced-Loops

Сега имате създадено празно решение без проекти в него:



Добавяйте в решението по един проект за всяка задача от упражненията. Така всяка задача ще бъде в отделен проект и всички проекти ще бъдат в общото решение.

Не забравяйте да зададете да се стартира по подразбиране текущия проект, а не първият в решението, като кликнете с десен бутон на мишката върху Solution 'Advanced-Loops' → [Set StartUp Projects...] → [Current selection].



## 1. Числата от 1 до N през 3

Напишете програма, която чете число  $n$ , въведено от потребителя, и отпечатва числата от 1 до  $n$  през 3 (със стъпка 3). Примери:

ВХОД	ИЗХОД
10	1 4 7 10

ВХОД	ИЗХОД
7	1 4 7

ВХОД	ИЗХОД
15	1 4 7 10 13

### Подсказки:

1. Създайте нов проект от менюто [File] → [New] → [New Project ...]
2. Изберете шаблон на проекта [C#] → [Windows] → [Console] → [Console App].
3. Именувайте проекта: LoopByStep3.
4. Напишете кода на програмата:

```
1 namespace LoopByStep3;
2 class Program
3 {
4     static void Main(string[] args)
5     {
6         Console.Write("n = ");
7         var n = int.Parse(Console.ReadLine());
8         Console.WriteLine("The numbers are: ");
9         for (int i = 1; i < n; i+=3)
10         {
11             Console.WriteLine(i);
12         }
13     }
14 }
```

5. Стартирайте с [Ctrl+F5] и тествайте:

```
n = 10
The numbers are:
1
4
7

C:\Users\mitko\Desktop\Advanced-Loops\LoopByStep3\bin\Debug\net6.0\LoopByStep3.exe
(process 6256) exited with code 0.
To automatically close the console when debugging stops, enable Tools->Options->Deb
ugging->Automatically close the console when debugging stops.
Press any key to close this window . . .
```



## 2. Числата от N до 1 в обратен рег

Напишете програма, която чете цяло положително число  $n$ , въведено от потребителя, и печата числата от  $n$  до 1 в обратен рег (от най-голямото към най-малкото). Примери:

ВХОД	ИЗХОД
2	2 1

ВХОД	ИЗХОД
3	3 2 1

ВХОД	ИЗХОД
5	5 4 3 2 1

**Подсказка:** отпечатайте  $n$  звездички в цикъл  $n$  пъти, точно както в предната задача.

## 3. Числа – степени на 2

Напишете програма, която чете от конзолата цяло число  $n$ , въведено от потребителя, и печата числата от 1 до  $2^n$ . Примери:

ВХОД	ИЗХОД
3	1 2 4 8

ВХОД	ИЗХОД
4	1 2 4 8 16

ВХОД	ИЗХОД
5	1 2 4 8 16 32

**Подсказка:** завъртете `for`-цикъл от 0 до  $n$  и започвайки от `num = 1` на всяка стъпка умножавайте `num` по 2.

## 4. Четни степени на 2

Да се напише програма, която чете число  $n$ , въведено от потребителя, и печата четните степени на  $2 \leq 2^n$ :  $2^0, 2^2, 2^4, 2^8, \dots, 2^n$ . Примери:

ВХОД	ИЗХОД
3	1 4

ВХОД	ИЗХОД
4	1 4 16

ВХОД	ИЗХОД
5	1 4 16

ВХОД	ИЗХОД
6	1 4 16 64

ВХОД	ИЗХОД
7	1 4 16 64

**Подсказка:** започнете от 1 и в цикъл умножавайте по 4 на всяка стъпка.

## 5. Редица числа $2K+1$

Напишете програма, която чете число  $n$ , подадено от потребителя, и отпечата всички числа  $\leq n$  от редицата: 1, 3, 7, 15, 31, .... Всяко следващо число се изчислява като предишното число  $\cdot 2 + 1$ . Примери:

ВХОД	ИЗХОД
3	1 3

ВХОД	ИЗХОД
8	1 3 7

ВХОД	ИЗХОД
17	1 3 7 15

ВХОД	ИЗХОД
31	1 3 7 15 31



#### Подсказки:

- Започнете от `num = 1`.
- В цикъл докато `num` не стигне `n`, печатайте, го умножавайте по 2 и прибавяйте 1.

### 6. Число в диапазона от 1 до 100

Напишете програма, която чете цяло положително число `n` в диапазона `[1...100]`, въведено от потребителя. При въвеждане на число извън посочения диапазон, да се отпечата съобщение за грешка и потребителят да се подкани да въведе ново число. Примери:

вход / изход
Enter a number in the range [1...100]: 35 The number is: 35
Enter a number in the range [1...100]: 105 Invalid number! Enter a number in the range [1...100]: 0 Invalid number! Enter a number in the range [1...100]: -200 Invalid number! Enter a number in the range [1...100]: 77 The number is: 77

#### Подсказки:

- Въведете число.
- Повтаряйте в цикъл докато числото е невалидно: отпечатайте грешка и въведете число отново.

### 7. Най-голям общ делител (НОД)

Напишете програма, която чете две цели положителни числа `a` и `b`, въведени от потребителя, и изчислява и отпечата най-големият им общ делител (НОД). Примери:

вход	изход	вход	изход	вход	изход	вход	изход	вход	изход
24	8	67	1	15	3	100	4	10	10
16		18		9		88		10	

Подсказка: имплементирайте алгоритъма на Евклид:

<https://bg.wikipedia.org/wiki/алгоритъм-на-Евклид>.