

# Лаб: Повторения с цикли – while-цикъл

Задачи за упражнение и домашно към курса "[Основи на програмирането](#)" в СофтУни.

## 1. Число в диапазона [1...100]

Напишете програма, която въвежда цяло положително **число** **n** в **диапазона [1...100]**. При въвеждане на число извън посочения диапазон, да се отпечата съобщение за грешка и **да се въведе ново число**.

### Примерен вход и изход

Вход / Изход
Enter a number in the range [1...100]: <b>35</b> The number is: 35
Enter a number in the range [1...100]: <b>105</b> Invalid number! Enter a number in the range [1...100]: <b>0</b> Invalid number! Enter a number in the range [1...100]: <b>-200</b> Invalid number! Enter a number in the range [1...100]: <b>77</b> The number is: 77

1. Създайте **нов проект** в съществуващото Visual Studio решение с име **"Number-In-Range"**.
2. **Прочетете входните данни от конзолата в искания вид** – едно цяло число **num**
3. Направете един **while** цикъл, който да се повтаря **докато** съответното число **num** е **по-малко от 1** или **по-голямо от 100**.
4. Ако програмата **влезе в тялото на цикъла**, то това ще означава, че потребителят е въвел число, което не е в диапазона [1...100]. В тялото на цикъла първо печатайте **"Invalid number!"**, след което **подканете потребителя да въведе ново число и присвоете стойността му на променливата num**.
5. **След тялото на цикъла** отпечатайте променливата **num** в желанния формат. Тя ще има стойност **между 1 и 100**, понеже програмата ще е **излязла от цикъла**.
6. Можете да си помогнете с **кода от картинката по-долу**:

```
static void Main(string[] args)
{
    Console.Write("Enter a number in the range [1...100]: ");
    int num = int.Parse(Console.ReadLine());

    while (num < 1 || num > 100)
    {
        Console.WriteLine("Invalid number!");
        Console.Write("Enter a number in the range [1...100]: ");
        num = int.Parse(Console.ReadLine());
    }
    Console.WriteLine($"The number is: {num}");
}
```

7. **Стартирайте** програмата с [Ctrl+F5] и я **тествайте** с различни входни стойности.
8. **Тествайте** решението си в **judge системата**: <https://judge.softuni.bg/Contests/Compete/Index/1014#0>

## 2. Завършване

Напишете програма, която изчислява **средната оценка** на ученик от цялото му обучение. На първия ред ще получите **името на ученика**, а на всеки следващ ред неговите годишни оценки. Ученикът преминава в следващия клас, **ако годишната му оценка е по-голяма или равна на 4.00**. Ако оценката му е под 4.00, той ще повтори класа.

При успешно завършване на **12-ти** клас да се отпечата:

```
"{име на ученика} graduated. Average grade: {средната оценка от цялото обучение}"
```

Стойността трябва да бъде форматирана до втория знак след десетичната запетая.

### Примерен вход и изход

вход	изход	вход	изход
Pesho 4 5.5 6 5.43 4.5 6 5.55 5 6 6 5.43 5	Pesho graduated. Average grade: 5.37	Ani 5 5.32 6 5.43 5 6 5.5 4.55 5 6 5.56 6	Ani graduated. Average grade: 5.45

Тествайте решението си в judge системата: <https://judge.softuni.bg/Contests/Compete/Index/1014#1>

1. Прочетете **едно име** от конзолата.
2. Създайте **две помощни променливи** – една променлива **брояч** с начална стойност **1** и една променлива с начална стойност **0**, в която ще държим **сумата от всички оценки** на дадения ученик.
3. Направете **един while** цикъл, който да се повтаря **докато брояча е по-малък или равен на 12**.
4. **В тялото на цикъла** четете **оценката на ученика за съответния клас** (реално число в интервала **[2.00...6.00]**).
5. След това направете **една проверка дали въведената оценка е по-голяма от 4.00** и ако е, я добавете **към помощната променлива за сума**, която си направихме в началото. **Увеличете стойността на брояча с 1**, понеже **щом оценката е 4 или повече, то ученикът е преминал в следващия клас**.
6. След тялото на **while** цикъла, изчислете **средната оценка от цялото обучение** като разделите **сумата от всички оценка на 12**.
7. **Принтирайте** резултата в желания формат.

Можете да си помогнете с кода от картинката, който е **нарочно замъглен**, за да помислите **как да решите задачата сами**:

```
static void Main(string[] args)
{
    string name = Console.ReadLine();
    double counter = 1;
    double sum = 0;
    while (counter <= 12)
    {
        double grade = double.Parse(Console.ReadLine());
        if (grade <= 4.00)
        {
            sum -= grade;
            counter++;
        }
        else
        {
            sum += grade;
            counter++;
        }
    }

    double average = sum / 12;
    Console.WriteLine($"{name} graduated. Average grade: {average:F2}");
}
```

### 3. Завършване - част 2

Напишете програма, която изчислява **средната оценка** на ученик от цялото му обучение. На първия ред ще получите **името на ученика**, а на всеки следващ ред неговите годишни оценки. Ученикът преминава в следващия клас, ако годишната му оценка е по-голяма или равна на **4.00**. Ако ученикът бъде скъсан повече от един път, то той бива изключен и програмата приключва, като се отпечата **името на ученика** и в **кой клас бива изключен**.

При успешно завършване на **12-ти** клас да се отпечата :

"{име на ученика} graduated. Average grade: {средната оценка от цялото обучение}"

В случай, че ученикът е изключен от училище, да се отпечата:

"{име на ученика} has been excluded at {класа, в който е бил изключен} grade"

Стойността трябва да бъде форматирана до втория знак след десетичната запетая.

### Примерен вход и изход

ВХОД	ИЗХОД	ВХОД	ИЗХОД
Gosho 5 5.5 6 5.43 5.5 6 5.55 5 6 6 5.43 5	Gosho graduated. Average grade: 5.53	Mimi 5 6 5 6 5 6 6 6 2 3	Mimi has been excluded at 8 grade

Тествайте решението си в judge системата: <https://judge.softuni.bg/Contests/Compete/Index/1014#2>

#### Насоки:

1. Използвайте решението от предната задача и добавете нужната функционалност:
2. Направете **още две помощни променливи в началото**, които да следят **дали ученика трябва да бъде изключен**. Едната променлива ще е **брояч** и трябва да е **от тип цяло число** и с **първоначална стойност нула**. С нея ще следим **колко пъти ученикът е трябвало да повтаря**. Другата променлива трябва да е **от булев тип** и да е с **началната стойност false**.
3. В тялото на **while** цикъла проверявайте **дали съответната оценка е по-ниска от 4.00** и ако е, **увеличавайте брояча с 1**.
4. Направете **още една проверка дали ученикът е бил изключван 2 пъти** (като използвате съответния брояч) и ако е, **променете стойността на булевата променлива на true** и **излезте от цикъла**.
5. След тялото на цикъла, **печатайте изхода в зависимост от това, дали ученикът е бил изключен или не**.

## 4. Редица цели числа

Напишете програма, която чете **цели числа**, **докато** не получи командата **"END"**. Принтирайте **най-голямото** и **най-малкото** число сред въведените.

#### Примерен вход и изход

ВХОД	ИЗХОД	ВХОД	ИЗХОД
10 20 304 0 50 END	Max number: 304 Min number: 0	250 5 2 0 100 1000 END	Max number: 1000 Min number: 0

Тествайте решението си в judge системата: <https://judge.softuni.bg/Contests/Compete/Index/1014#3>

## 5. Стъпки

Габи иска да започне здравословен начин на живот и си е поставила за цел да върви **10 000 стъпки всеки ден**. Някои дни обаче е много уморена от работа и ще иска да се прибере преди да постигне целта си. Напишете програма, която **чете от конзолата по колко стъпки изминава** тя всеки път като излиза през деня и **когато постигне целта си да се изписва "Goal reached! Good job!"**

Ако иска да се **прибере преди това**, тя ще въведе командата **"Going home"** и **ще въведе стъпките, които е извървяла докато се прибира**. След което, ако не е успяла да постигне целта си, на конзолата трябва да се изпише: **"{разликата между стъпките} more steps to reach goal."**

#### Примерен вход и изход

Вход	Изход	Вход	Изход
1000 1500 2000 6500	Goal reached! Good job!	1500 300 2500 3000 Going home 200	2500 more steps to reach goal.

Вход	Изход	Вход	Изход
1500 3000 250 1548 2000 Going home 2000	Goal reached! Good job!	125 250 4000 30 2678 4682	Goal reached! Good job!

Тествайте решението си в judge системата: <https://judge.softuni.bg/Contests/Compete/Index/1014#4>

## 6. Баланс по сметка

Напишете програма, която пресмята колко общо пари има в сметката, след като направите определен брой вноски. На първия ред ще получите **колко вноски трябва да се направят**. На всеки следващ ред ще получавате сумата, която трябва да внесете в сметката, **докато не се достигне броя вноски**. При всяка получена сума на конзолата трябва да се извежда **"Increase: " + сумата** и тя да се **прибавя в сметката**. Ако получите число **по-малко от 0** на конзолата трябва да се изведе **"Invalid operation!"** и **програмата да приключи**. Когато програмата приключи трябва да се принтира **"Total: " + общата сума** в сметката закръглена до втория знак след десетичната запетая.

### Примерен вход и изход

ВХОД	ИЗХОД
3	Increase: 5.51
5.51	Increase: 69.42
69.42	Increase: 100
100	Total: 174.93

ВХОД	ИЗХОД
5	Increase: 120
120	Increase: 45.55
45.55	Invalid operation!
-150	Total: 165.55

Тествайте решението си в judge системата: <https://judge.softuni.bg/Contests/Compete/Index/1014#5>

## 7. Чаша с вода

Всеки офис си има диспенсър за вода. Напишете програма, която отчита дали служител от офиса е успял да напълни чашата си успешно.

Диспенсърът има три бутона :

- Easy - лесен (50 милилитра)
- Medium - среден (100 милилитра)
- Hard - силен (200 милилитра)

Ще получите **обемът на чашата** и на всеки следващ ред **кой бутон е бил натиснат**. Ако чашата се напълни или прелее програмата **приключва**.

Ако чашата прелее отпечатайте **колко вода е била излята**. При успешно напълване отпечатайте **броя на натисканията на бутона**.

### Примерен вход и изход

ВХОД	ИЗХОД
------	-------

ВХОД	ИЗХОД
------	-------

500 Hard Medium Easy Easy Medium	The dispenser has been tapped 5 times.	600 Hard Medium Medium Easy Hard	50ml has been spilled.
---	--	---	------------------------

Тествайте решението си в judge системата: <https://judge.softuni.bg/Contests/Compete/Index/1014#6>

## 8. Редица числа $2k+1$

Напишете програма, която чете число  $n$ , въведено от потребителя, и отпечатва **всички числа  $\leq n$  от редицата:**  
1, 3, 7, 15, 31, .... Всяко следващо число се изчислява като умножим **предишното** с 2 и добавим 1.

### Примерен вход и изход

вход	изход	вход	изход	вход	изход	вход	изход
3	1 3	8	1 3 7	17	1 3 7 15	31	1 3 7 15 31

Тествайте решението си в judge системата: <https://judge.softuni.bg/Contests/Compete/Index/1014#7>

Подсказки:

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

## Примерна изпитна задача

### 9. \*Преместване

Тествайте решението си [тук](#).

На осемнадесетия си рожден ден на Хосе взел решение, че ще се изнесе да живее на квартира. Опаковал багажа си в **кашони** и намерил подходяща обява за апартамент под наем. Той започва да пренася своя багаж **на части**, защото не може да пренесе целия наведнъж. Има ограничено **свободно пространство** в новото си жилище, където може да разположи вещите, така че мястото да бъде подходящо за живеене.

Напишете **програма, която изчислява свободния обем от жилището на Хосе, който остава след като пренесе багажа си.**

Бележка: Един кашон е с точни размери: 1m. x 1m. x 1m.

### Вход

Потребителят въвежда следните данни на отделни редове:

- Широчина на свободното пространство - цяло число в интервала [1...1000]
- Дължина на свободното пространство - цяло число в интервала [1...1000]
- Височина на свободното пространство - цяло число в интервала [1...1000]
- На следващите редове (до получаване на команда "Done") - брой кашони, които се пренасят в квартирата - цели числа в интервала [1...10000];

Програмата трябва да приключи прочитането на данни при команда "Done" или ако свободното място свърши.

## Изход

Да се отпечата на конзолата **един** от следните редове:

- Ако стигнете до командата "Done" и има още свободно място:  
"{брой свободни куб. метри} Cubic meters left."
- Ако свободното място свърши преди да е дошла команда "Done":  
"No more free space! You need {брой недостигащи куб. метри} Cubic meters more."

## Примерен вход и изход

Вход	Изход	Обяснение
10 10 2 20 20 20 20 122	No more free space! You need 2 Cubic meters more.	$10 * 10 * 2 = 200$ кубични метра. $20 + 20 + 20 + 20 + 122 = 202$ кубични метра. $200 - 202 = 2$ недостигащи кубични метра
10 1 2 4 6 Done	10 Cubic meters left.	$10 * 1 * 2 = 20$ кубични метра. $4 + 6 = 10$ кубични метра. $20 - 10 = 10$ кубични метра.