

# Лаб: Повторения с цикли – While-цикъл

Задачи за упражнение и домашно към курса ["Основи на програмирането със C++" @ СофтУни](#).

Тествайте решението си в judge системата: <https://judge.softuni.org/Contests/1175/While-Loop-Lab>

## 1. Четене на думи

Напишете програма, която чете текст от конзолата (string) и го принтира, докато не получи командата "Stop".

### Примерен вход и изход

ВХОД	ИЗХОД
Nakov	Nakov
SoftUni	SoftUni
Sofia	Sofia
Bulgaria	Bulgaria
SomeText	SomeText
Stop	
AfterStop	
Europe	
HelloWorld	

ВХОД	ИЗХОД
Sofia	Sofia
Berlin	Berlin
Moscow	Moscow
Athens	Athens
Madrid	Madrid
London	London
Paris	Paris
Stop	
AfterStop	

## 2. Парола

Напишете програма, която първоначално прочита име и парола на потребителски профил. След това чете парола за вход.

- при въвеждане на грешна парола: потребителя да се подкани да въведе нова парола.
- при въвеждане на правилна парола: отпечатваме "Welcome {username}!".

### Примерен вход и изход

ВХОД	ИЗХОД
Nakov	Welcome Nakov!
1234	
pass	
1324	
1234	

ВХОД	ИЗХОД
Gosho	Welcome Gosho!
secret	
secret	

## Насоки

- Инициализирайте две променливи **username** и **password**, които ще съдържат потребителското име и паролата:

```
string username, password;  
cin >> username >> password;
```

2. Инициализирайте променлива **input**, която ще държи въведената от потребителя парола за вход:

```
string input;  
cin >> input;
```

3. В **while** цикъл, до въвеждане на валидна парола, четете нова:

```
while (input != password) {  
    cin >> input;  
}
```

4. Когато се въведе **валидна парола** **принтирайте** съобщението за **успешен вход**:

```
string username, password;  
cin >> username >> password;  
  
string input;  
cin >> input;  
while (input != password) {  
    cin >> input;  
}  
  
cout << "Welcome " << username << "!" << endl;
```

### 3. Сума от числа

Напишете програма, която чете цяло число от конзолата и на всеки следващ ред цели числа, докато тяхната сума стане по-голяма или равна на първоначалното число. След приключване на четенето да се отпечата сумата на въведените числа.

#### Примерен вход и изход

ВХОД	ИЗХОД	ВХОД	ИЗХОД
100	100	20	21
10		1	
20		2	
30		3	
40		4	
		5	
		6	

### 4. Редица числа $2k+1$

Напишете програма, която чете число **n**, въведено от потребителя, и отпечата **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
3	1 3

вход	изход
8	1 3 7

вход	изход
17	1 3 7 15

вход	изход
31	1 3 7 15 31

## Насоки

1. Прочетете от конзолата цяло число.

```
int number;  
cin >> number;
```

2. Създайте променлива от тип **цяло число**, която ще е брояч и има **първоначална стойност 1**.

```
int k = 1;
```

3. Създайте **while** цикъл, който се повтаря докато **брояча е по-малък** от числото, което сте прочели от конзолата.

```
while (k <= number) {  
    ...  
}
```

4. При всяко повторение на цикъла **принтирайте стойността на брояча** и му **прибавяйте дадената стойност**.

```
while (k <= number) {  
    cout << k << endl;  
    k = k * 2 + 1;  
}
```

## 5. Баланс по сметка

Напишете програма, която пресмята **колко общо пари** има в сметката, след като направите **определен брой вноски**. На всеки ред ще получавате сумата, която трябва да внесете в сметката, до получаване на команда "NoMoreMoney". 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 форматирана до втория знак след десетичната запетая.

## Примерен вход и изход

вход	изход
5.51	Increase: 5.51
69.42	Increase: 69.42
100	Increase: 100.00
NoMoreMoney	Total: 174.93

вход	изход
120	Increase: 120.00
45.55	Increase: 45.55
-150	Invalid operation!
	Total: 165.55

## 6. Най-голямо число

Напишете програма, която до получаване на командата "Stop", чете **цели числа**, въведени от потребителя и намира **най-голямото** измежду тях. Въвежда се по едно число на ред.

### Примерен вход и изход

вход	изход
100	100
99	
80	
70	
Stop	

вход	изход
-10	20
20	
-30	
Stop	

вход	изход
45	99
-20	
7	
99	
Stop	

вход	изход
999	999
Stop	

вход	изход
-1	-1
-2	
Stop	

## 7. Най-малко число

Напишете програма, която до получаване на командата "Stop", чете **цели числа**, въведени от потребителя и намира **най-малкото** измежду тях. Въвежда се по едно число на ред.

### Примерен вход и изход

вход	изход
100	70
99	
80	
70	
Stop	

вход	изход
-10	-30
20	
-30	
Stop	

вход	изход
45	-20
-20	
7	
99	
Stop	

вход	изход
999	999
Stop	

вход	изход
-1	-2
-2	
Stop	

## 8. 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 ученикът бъде скъсан повече от един път, то той бива изключен и програмата приключва, като се отпечатва 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
Gosho	Gosho graduated. Average grade: 5.53
5	
5.5	
6	
5.43	
5.5	
6	
5.55	
5	
6	
6	
5.43	
5	

вход	изход
Mimi	Mimi has been excluded at 8 grade
5	
6	
5	
6	
5	
6	
6	
2	
3	