

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://softuni.bg/trainings/3273/programming-basics-with-cpp-january-2021>.

1. Четене на думи

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

Примерен вход и изход

ВХОД	ИЗХОД	ВХОД	ИЗХОД
Nakov SoftUni Sofia Bulgaria SomeText Stop	(няма изход)	Sofia Berlin Moscow Athens Madrid London Paris Stop	(няма изход)

Насоки

1. Инициализирайте променлива **input**, която ще държи въведената от потребителя дума:

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

2. В **while** цикъл, до въвеждане на командата "Stop", четете нова дума:

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

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.

Примерен вход и изход

ВХОД	ИЗХОД	ВХОД	ИЗХОД
------	-------	------	-------

Nakov 1234 pass 1324 1234	Welcome Nakov!		Gosho secret secret	Welcome Gosho!
---------------------------------------	----------------	--	---------------------------	----------------

Насоки

1. 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 цели числа, докато не получи командата **"Stop"**. След приключване на четенето да се отпечата **сумата на всички въведени числа**.

Примерен вход и изход

ВХОД	ИЗХОД	ВХОД	ИЗХОД
------	-------	------	-------

10	105	1	21
20		2	
30		3	
45		4	
Stop		5	
		6	
		Stop	

Насоки

1. Инициализирайте променлива **sum**, в която ще добавяме прочетените числа:

```
int sum = 0;
```

2. Инициализирайте променлива **input**, която ще държи въведеното от потребителя число, или командата "Stop":

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

3. В **while** цикъл, до въвеждане на командата "Stop", четете ново число и го прибавяйте към сумата:

```
while (input != "Stop")
{
    int currentNum = stoi(input);
    sum += currentNum;
    cin >> input;
}
```

4. Когато се въведе командата "Stop", принтирайте сумата на числата:

```
int sum = 0;
string input;
cin >> input;

while (input != "Stop")
{
    int currentNum = stoi(input);
    sum += currentNum;
    cin >> input;
}
cout << sum << endl;
```

4. 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

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 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
------	-------

вход	изход
------	-------

3	Increase: 5.51	5	Increase: 120
5.51	Increase: 69.42	120	Increase: 45.55
69.42	Increase: 100	45.55	Invalid operation!
100	Total: 174.93	-150	Total: 165.55

6. Най-голямо число

Напишете програма, която чете **n-на брой цели числа** ($n > 0$), въведени от потребителя, и намира **най-голямото** измежду тях. Първо се въвежда броят числа **n**, а след това самите **n** числа, по едно на ред.

Примерен вход и изход

вход	изход	вход	изход	вход	изход	вход	изход	вход	изход
2	100	3	20	4	99	1	999	2	-1
100		-10		45		999		-1	
99		20		-20				-2	
		-30		7					
				99					

7. Най-малко число

Напишете програма, която чете **n-на брой цели числа** ($n > 0$), въведени от потребителя и намира **най-малкото** измежду тях. Първо се въвежда броят числа **n**, а след това самите **n** числа, по едно на ред.

Примерен вход и изход

вход	изход	вход	изход	вход	изход	вход	изход	вход	изход
2	99	3	-30	4	-20	1	999	2	-2
100		-10		45		999		-1	
99		20		-20				-2	
		-30		7					
				99					

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 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	Pesho graduated. Average grade: 5.37	Ani	Ani graduated. Average grade: 5.45
4		5	
5.5		5.32	
6		6	
5.43		5.43	

4.5		5	
6		6	
5.55		5.5	
5		4.55	
6		5	
6		6	
5.43		5.56	
5		6	

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

Примерна изпитна задача

10. *Преместване

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:

1. Широчина на свободното пространство - цяло число в интервала [1...1000]
2. Дължина на свободното пространство - цяло число в интервала [1...1000]
3. Височина на свободното пространство - цяло число в интервала [1...1000]
4. 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.