

Лаб: Повторения с цикли – for-цикъл

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

Тествайте решенията си тук: <https://judge.softuni.bg/Contests/Compete/Index/1173#0>

1. Числа от 1 до 100

Напишете програма, която отпечатва числата от 1 до 100, всяко на нов ред.

Примерен вход и изход

вход	изход
(няма)	1
	2
	3
	...
	98
	99
	100

Насоки

1. Създайте **нов проект с име "Numbers1To100"**.
2. Отидете в тялото на метода **main()** и напишете решението на задачата. Можете да си помогнете с кода от картинката по-долу:

```
int main() {  
  
    for(int i = 1; i<=100; i++){  
        cout << i << endl;  
    }  
  
    return 0;  
}
```

2. Числата от N до 1 в обратен ред

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

Примерен вход и изход

вход	изход	вход	изход	вход	изход
2	2 1	3	3 2 1	5	5 4 3 2 1

Насоки

1. Прочетете **едно цяло число** от конзолата – **n**:

```
int n;  
cin >> n;
```

2. Направете **един for цикъл** от **n** до **0**, но **вместо да увеличавате променливата i с 1**, я **намалявайте с 1** при всяка итерация на цикъла:

```
for(int i = n; i>0; i--){  
  
}
```

3. В тялото на цикъла **принтирайте променливата i**:

```
int main(){  
    int n;  
    cin >> n;  
  
    for(int i = n; i>0; i--){  
        cout << i;  
    }  
}
```

3. 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).

Примерен вход и изход

вход	изход
10	1 4 7 10

вход	изход
7	1 4 7

вход	изход
15	1 4 7 10 13

Насоки

1. Прочетете входните данни от конзолата – **едно цяло число n**

```
int n;  
cin >> n;
```

2. Направете **for цикъл** от **1 до n (включително)** и задайте **стъпка 3**. Това означава, че при всяка итерация на цикъла, променливата **i** **ще увеличава стойността си с 3**, вместо с 1:

```
for(int i = 1; i<=n; i+=3){  
  
}
```

3. В тялото на цикъла принтирайте променливата `i`:

```
for(int i = 1; i<=n; i+=3) {  
    cout << i << endl;  
}
```

4. Четни степени на 2

Да се напише програма, която чете число `n`, въведено от потребителя, и печата четните степени на 2: 2^0 , 2^2 , 2^4 , 2^6 , ..., 2^n .

Примерен вход и изход

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

5. Поток от символи

Напишете програма, която чете текст(стринг), въведен от потребителя и печата всеки символ от текста на отделен ред.

Примерен вход и изход

вход	изход	вход	изход
softuni	s o f t u n i	ice cream	i c e c r e a m

Насоки

- Прочетете входният текст:
- Направете **for** цикъл с начална стойност на контролната променлива от 0 до `input.length()` (дължината на текста). На всяка итерация взимайте буквата на позиция във въведената дума равна на стойността на контролната променлива `i`, чрез метода `charAt()`
- На всяка итерация принтирайте стойността на променливата `letter`:

6. Сумиране на гласните букви

Да се напише програма, която чете **текст** (стринг), въведен от потребителя, и изчислява и отпечатва **сумата от стойностите на гласните букви** според таблицата по-долу:

буква	a	e	i	o	u
стойност	1	2	3	4	5

Примерен вход и изход

вход	изход	коментар
hello	6	e + o = 2 + 4 = 6
hi	3	i = 3
bamboo	9	a + o + o = 1 + 4 + 4 = 9
beer	4	e + e = 2 + 2 = 4

Насоки

1. Прочетете входните данни и ги запазете в променлива **word**. Създайте променлива за сумата на буквите.

```
string word;
cin >> word;

int sum = 0;
```

2. Направете **for** цикъл от 0 до **word.length()** (дължината на текста).

```
string word;
cin >> word;

int sum = 0;

for(int i = 0; i < word.length(); i++){
}
```

3. Проверете всяка буква **word[i]** дали е гласна и съответно добавете към сумата стойността ѝ:

```
for(int i = 0; i < word.length(); i++){
    char letter = word[i];

    if(word[i] == 'a'){
        sum++;
    }else if(word[i] == 'e'){
        sum += 2;
    }else if(word[i] == 'i'){
        sum += 3;
    }else if(word[i] == 'o'){
        sum += 4;
    }else if(word[i] == 'u'){
        sum += 5;
    }
}
```

7. Сумиране на числа

Да се напише програма, която **чете n**-на брой **цели числа**, въведени от потребителя и **ги сумира**.

- От първия ред на входа се въвежда броят числа **n**.
- От следващите **n** реда се въвежда по едно цяло число.

Програмата трябва да прочете числата, да ги сумира и да отпечата сумата им.

Примерен вход и изход

вход	изход
2	30
10	
20	

вход	изход
3	-60
-10	
-20	
-30	

вход	изход
4	43
45	
-20	
7	
11	

вход	изход
1	999
999	

вход	изход
0	0

Насоки

1. Първо въведете едно число **n** (броят числа, които предстои да бъдат въведени).

```
int n;  
cin >> n;
```

2. Създайте променлива **sum = 0**, която ще държи сумата на числата. (в началото няма още прочетени числа, и съответно сумата е равна на 0).

```
int sum = 0;
```

3. В цикъл **n** пъти прочетете по едно цяло число **num** и го прибавете към сумата (**sum = sum + num**).

```
int main(){  
  
    int n;  
    cin >> n;  
  
    int sum = 0;  
  
    for(int i = 0; i < n; i++){  
        cin >> num;  
        sum += num;  
    }  
    return 0;  
}
```

4. Накрая отпечатайте получената сума.

```
cout << sum;
```

8. Редица цели числа

Напишете програма, която **чете n** на брой **цели числа**. Принтирайте **най-голямото** и **най-малкото** число сред въведените.



Примерен вход и изход

вход	изход	вход	изход
5	Max number: 304	6	Max number: 1000
10	Min number: 0	250	Min number: 0
20		5	
304		2	
0		0	
50		100	
		1000	

Насоки

1. Създайте две променливи от тип цяло число, като променливата, в която ще пазите най-голямото число започва от възможно най-малкото число за типа, а променливата, в която ще пазите най-малкото число започва от възможно най-голямото число за типа. Константите `INT_MAX` и `INT_MIN` се намират в библиотеката `<climits>`

```
int maxNumber = INT_MAX;  
int minNumber = INT_MIN;
```

2. Създайте променлива `n`, която да чете броя числа, които ще бъдат прочетени.

```
int n;  
cin >> n;
```

3. В цикъл `n` пъти прочетете по едно цяло число.

```
for(int i = 0; i < n; i++){  
    int num;  
    cin >> num;  
}
```

4. Проверете дали стойността е по-голяма и по-малка от променливите, които създадохте и ако е, ги презаписвайте. След края на цикъла отпечатайте най-голямото и най-малкото число.

```

for(int i = 0; i < n; i++){
    int num;
    cin >> num;

    if(num > maxNumber){
        maxNumber = num;
    }

    if(num < minNumber){
        minNumber = num;
    }

    cout << "Max number: " << maxNumber;
    cout << "Min number: " << minNumber;
}

```

9. Лява и дясна сума

Да се напише програма, която чете **2*n-на брой** цели числа, подадени от потребителя, и проверява дали **сумата на първите n числа** (лява сума) е равна на **сумата на вторите n числа** (дясна сума). При равенство печата "Yes, sum = " + **сумата**; иначе печата "No, diff = " + **разликата**. Разликата се изчислява като положително число (по абсолютна стойност).

Примерен вход и изход

вход	изход	коментар
2 10 90 60 40	Yes, sum = 100	10+90 = 60+40 = 100

вход	изход	коментар
2 90 9 50 50	No, diff = 1	90+9 ≠ 50+50 Difference = 99-100 = 1

Насоки

1. Въведете n.

```

int n;
cin >> n;

```

2. Създайте променливи за сумата на лявата и дясната половина.

```

int leftSum = 0;
int rightSum = 0;

```

3. Направете **цикъл за лявата половина** като прочитате число **num** на всеки ред и го сумирайте към **leftSum**.

```
for(int i = 0; i<n; i++){
    int num;
    cin >> num;

    leftSum += num;
}
```

4. Аналогично направете същото и за дясната половина.

```
for(int i = 0; i<n; i++){
    int num;
    cin >> num;

    rightSum += num;
}
```

5. Изчислете **разликата** между сумите по абсолютна стойност: **abs(leftSum - rightSum)** и проверете ако разликата е 0, отпечатайте "Yes" + **сумата**; иначе отпечатайте "No" + **разликата**.

```
int dif = abs(leftSum - rightSum);
if(dif == 0){
    cout << "Yes, sum = " << leftSum;
}else{
    cout << "No, diff = " << rightSum;
}
```

10. Четна / нечетна сума

Да се напише програма, която чете **n-на брой** цели числа, подадени от потребителя, и проверява дали **сумата от числата на четни позиции** е равна на **сумата на числата на нечетни позиции**. При равенство да се отпечата два реда: "Yes" и на нов ред "Sum = " + **сумата**; иначе да се отпечата "No" и на нов ред "Diff = " + **разликата**. Разликата се изчислява по абсолютна стойност.

Примерен вход и изход

вход	изход	коментар
4 10 50 60 20	Yes Sum = 70	10+60 = 50+20 = 70

вход	изход	коментар
4 3 5 1 -2	No Diff = 1	3+1 ≠ 5-2 Diff = 4-3 = 1

вход	изход	коментар
3 5 8 1	No Diff = 2	5+1 ≠ 8 Diff = 6-8 = 2

Насоки

1. Въведете **n** и направете **цикъл от 1 до n включително** като на всеки ред четете число **num**.


```
int main() {

    int n;
    cin >> n;

    for(int i = 1; i<=n; i++){
        int num;
        cin >> num;
    }

    return 0;
}
```

2. Създайте две променливи, които ще съдържат **сумата на четните и нечетните позиции**.

```
int main() {

    int n;
    cin >> n;
    int odd = 0;
    int even = 0;

    for(int i = 1; i<=n; i++){
        int num;
        cin >> num;
    }

    return 0;
}
```

3. Проверете числото дали е на **четна** или **нечетна** позиция и увеличете съответната сума, като имате в предвид, че **променливата i** отговаря на **позицията**.

```
for(int i = 1; i<=n; i++){
    int num;
    cin >> num;

    if(i%2 == 0){
        even += num;
    }else{
        odd += num;
    }
}
```

4. Изчислете **разликата** между сумите по абсолютна стойност: **abs(odd - even)** и проверете ако разликата е **0**, отпечатайте **"Yes"** и на нов ред **"Sum = " + сумата**; иначе отпечатайте **"No"** и на нов ред **"Diff = " + разликата**.

```

int diff = abs(odd - even);
if(diff == 0){
    cout << "Yes" << endl;
    cout << "Sum = " << odd;
}else{
    cout << "No" << endl;
    cout << "Diff = " << diff;
}

```

Примерна изпитна задача

11. Умната Лили

Лили вече е на **N** години. За всеки свой **рожден ден** тя получава подарък. За **нечетните** рождени дни (1, 3, 5...n) получава **играчки**, а за всеки **четен** (2, 4, 6...n) получава **пари**. За **втория** рожден ден получава **10.00** лв, като **сумата се увеличава с 10.00** лв., за всеки **следващ четен** рожден ден (2 -> 10, 4 -> 20, 6 -> 30...и т.н.). През годините Лили тайно е спестявала парите. **Братът** на Лили, в **годините**, които тя **получава пари**, **взима по 1.00** лев от тях. Лили **продала** играчките получени през годините, **всяка за P лева** и **добавила** сумата **към спестените пари**. С парите искала да си **купи пералня за X лева**. Напишете програма, която да **пресмята, колко пари е събрала** и дали **й стигат да си купи пералня**.

Вход

Програмата прочита **3 числа**, въведени от потребителя, на отделни редове:

- **Възрастта** на Лили - **цяло число** в интервала [1...77]
- **Цената на пералнята** - **число** в интервала [1.00...10 000.00]
- **Единична цена на играчка** - **цяло число** в интервала [0...40]

Изход

Да се отпечата на конзолата един ред:

- Ако парите на Лили са достатъчни:
 - "Yes! {N}" - където **N** е остатъка пари след покупката
- Ако парите не са достатъчни:
 - "No! {M}" - където **M** е сумата, която не достига
- Числата **N** и **M** трябва да са **форматирани до вторият знак след десетичната запетая**.

Примерен вход и изход

вход	изход	Коментари
10 170.00 6	Yes! 5.00	<p>Първи рожден ден получава играчка; 2ри -> 10лв; 3ти -> играчка; 4ти -> 10 + 10 = 20лв; 5ти -> играчка; 6ти -> 20 + 10 = 30лв; 7ми -> играчка; 8ми -> 30 + 10 = 40лв; 9ти -> играчка; 10ти -> 40 + 10 = 50лв.</p> <p>Спестила е -> 10 + 20 + 30 + 40 + 50 = 150лв. Продала е 5 играчки по 6 лв. = 30лв. Брат ѝ взел 5 пъти по 1 лев = 5лв. Остават -> 150 + 30 - 5 = 175лв. 175 >= 170 (цената на пералнята) успяла е да я купи и са и останали 175-170 = 5 лв.</p>

21 1570.98 3	No! 997.98	Спестила е 550лв . Продала е 11 играчки по 3 лв. = 33лв . Брат ѝ взимал 10 години по 1 лев = 10лв. Останали 550 + 33 – 10 = 573лв 573 < 1570.98 – не е успяла да купи пералня. Не ѝ достигат 1570.98–573 = 997.98лв
--------------------	------------	---

Насоки

1. Прочетете **входните данни** - **възрастта** на Лили, **цената на пералнята** и **цената на една играчка**:

```
int age;
cin >> age;

double washingMachinePrice;
cin >> washingMachinePrice;

int toyPrice;
cin >> toyPrice;
```

2. Създайте променливи, които ще пазят **броя години**, в които Лили получава **пари** и **играчки**.

```
int moneyYears = 0;
int toyYears = 0;
```

3. Във **for** цикъл според от **1** до **годините на Лили**, добавяйте към броя на годините в, който получава пари или към годините на получените играчки.

```
for(int i = 1; i <= age; i++){
    if(i%2 == 0){
        moneyYears++;
    }else{
        toyYears++;
    }
}
```

4. Създайте променлива **money**, която ще пази парите на Лили. Във **for** цикъл от **1** до броя на годините, в които Лили получава пари, увеличавайте **money** с **10 * за всяка година**.

```
for(int i = 1; i <= moneyYears; i++){
    money += 10 * i;
}
```

5. Към **money**, добавете произведението на броя на годините играчки * цената на играчките, като извадите броя на годините получени пари.

```
money += (toyYears * toyPrice) - moneyYears;
```

6. Създайте променлива, която ще има за **стойност**, абсолютната стойност на **разликата от цената на пералнята и парите**, с които разполага Лили.

```
double dif = abs(washingMachinePrice - money);
```

7. Проверете дали **парите**, с които разполага Лили са достатъчни, да купи **пералнята машина** и съответно принтирайте **"Yes!" + dif**. Ако парите не са достатъчни, принтирайте **"No!" + dif**.

```
cout.setf(ios::fixed);  
cout.precision(2);  
  
if(money >= washingMachinePrice){  
    cout << "Yes!" << dif;  
}else{  
    cout << "No! " << dif;  
}
```