



ОПЕРАТОРИ ЗА ЦИКЪЛ [WHILE], [DO/WHILE]



УСЛОВЕН ОПЕРАТОР WHILE

• СЕМАНТИКА НА ОПЕРАТОРА WHILE

Чрез този оператор може да се реализира произволен цикличен процес.

```
инициализация (i)
WHILE(<УСЛОВИЕ>)
{
    ОПЕРАТОР
    корекция (i)
}
```

Пример:

```
int i = 1; - инициализация
int a = 5;
while(i <= a) - условие
{
    cout << i + a << endl; - оператор
    i++; - корекция
}
```

Докато е в сила условието, повтаряй следното...



УСЛОВЕН ОПЕРАТОР WHILE

- СЕМАНТИКА НА ОПЕРАТОРА WHILE

while (докато) – запазена дума

<условие> - булев израз

оператор – оператори за управление на изчислителния процес

Пресмята се стойността на <условие>. **Ако тя е false** изпълнението на оператора while завършва без да се изпълни тялото му. **Ако тя е true**, изчисляването на условието и изпълнението на оператора продължава, докато <условие> = **false**



УСЛОВЕН ОПЕРАТОР WHILE

• СЕМАНТИКА НА ОПЕРАТОРА WHILE

Да се състави програма, в която потребителят въвежда естествено число, а програмата извежда неговите делители.

Решение, чрез FOR цикъл

```
#include <iostream>
using namespace std;

int main()
{
    int a, n;
    cout << "Please, enter a number" << endl;
    cin >> a;

    for(int i=1; i<=100000; i++)
    {
        n=a%i;
        if(n==0)
        {
            cout << i << endl;;
        }
    }
    return 0;
}
```



УСЛОВЕН ОПЕРАТОР WHILE

- СЕМАНТИКА НА ОПЕРАТОРА WHILE

Решение, чрез WHILE цикъл

```
#include <iostream>
using namespace std;

int main()
{
    int a, n, i;
    i = 1;
    cout << "Please, enter a number" << endl;
    cin >> a;

    while(i <= a)
    {
        n = a % i;
        if(n == 0)
        {
            cout << i << endl;
        }

        i++;
    }
    return 0;
}
```



УСЛОВЕН ОПЕРАТОР DO/WHILE

- СЕМАНТИКА НА ОПЕРАТОРА DO/WHILE

Използва се за реализиране на **произволни** циклични процеси.

инициализация

DO

{

ОПЕРАТОР

корекция

}

WHILE(УСЛОВИЕ);

```
int main()
```

```
{
```

```
    int i, a;
```

```
    i = 1; - инициализация
```

```
    a = 0;
```

```
    do
```

```
    {
```

```
        оператор
```

```
        cout << i+a << endl;
```

```
        i++; - корекция
```

```
    }
```

```
    while(i<=a); - условие
```

```
}
```



УСЛОВЕН ОПЕРАТОР DO/WHILE

• СЕМАНТИКА НА ОПЕРАТОРА DO/WHILE

do (повтаряй), **while** (докато) – запазени думи
<оператор> - точно един оператор за управление на изчислителния процес
<условие> - булев израз

Изпълнява се тялото на цикъла **поне веднъж, след което се пресмята стойността на <условие>**. Ако то е **false**, изпълнението на оператора do/while **завършва**. В противен случай се повтарят действията:

- Изпълнение на тялото на цикъла
- Пресмятане на стойността на <условие>, докато <условие> = true



УСЛОВЕН ОПЕРАТОР DO/WHILE

- СЕМАНТИКА НА ОПЕРАТОРА DO/WHILE

Решение, чрез DO/WHILE цикъл

```
#include <iostream>
using namespace std;

int main()
{
    int a, n, i;
    i = 1;
    cout << "Please, enter a number" << endl;
    cin >> a;

    do
    {
        n = a % i;
        if (n == 0)
        {
            cout << i << endl;
        }

        i++;
    }
    while (i <= a);
}
```




ЗАДАЧИ

• ЗАДАЧА 24

Да се състави програма, която изчислява сумата от цифрите на всички естествени 2-цифрени числа до въведено от клавиатурата 2-цифрено число.

Пример: при въведено число 13, програмта извежда 51 ($1 + 2 + 3 + 4 + 5 + 6 + 7 + 8 + 9 + 1 + 0 + 1 + 1 + 1 + 2$).



ЗАДАЧИ

- ЗАДАЧА 25

Да се състави програма, която извежда всички естествени трицифрени числа, които нямат еднакви цифри т.е. 100, 101 и подобните на тях не се извеждат.



ЗАДАЧИ

• ЗАДАЧА 26

Да се състави програма, която по дадено естествено число от интервала $[100 - 30000]$ намира най-голямата, най-малката от цифрите му и тяхната средна стойност (на цифрите).

Пример: при въведено число 1234, най-малката цифра е 1, а най-голямата 4. Тяхната средна стойност е 2.5.



ЗАДАЧИ

• ЗАДАЧА 27

Една жаба живеела на дъното на кладенец. Гледайки непрекъснато светлото петно над себе си решила да се покачи и да го разгледа. Всеки ден тя се изкачвала по 2 метра нагоре, но през нощта се разколебавала и слизала по 1 метър надолу.

Да се състави програма, която чрез цикъл `while` описва движението на жабата. Дълбочината на кладенеца се въвежда от клавиатурата.

Пример: при 20 метра дълбочина на кладенеца жабата се изкачва на 19-тия ден.



ЗАДАЧИ

• ЗАДАЧА 28

Да се състави програма, в която компютърът да генерира случайно целочислено число в интервала $[1, 100]$, а потребителят трябва да го познае, въвеждайки число. При всяко въвеждане на число програмта трябва да изведе подходящо съобщение указващо дали въведеното число е по-малко или по-голямо от генерираното. Програмата приключва, когато потребителят познае числото и трябва да изведе на екрана броят опити необходими за пазнаване на генерираното число.



ЗАДАЧИ

- Допълнителни задачи

Едно положително цяло число е съвършено, ако е равно на сумата от своите делители (без самото число). Например 6 е съвършено, защото $6 = 1 + 2 + 3$; числото 1 не е съвършено. Да се напише процедура, която отпечатва на екрана всички съвършени числа ненадминаващи дадено положително цяло число в параметър n

Да се напише целочислена функция с параметри n и k , която намира цялото число, което се получава от положителното цяло число n , като се задраска k -тата му отдясно наляво цифра. Например, ако n е 31245 и k е 4, функцията трябва да намери числото 3245, ако k е 2, резултатът трябва да е 3125. При стойности $n = 5$, а $k=1$, функцията трябва да връща стойност 0



ЗАДАЧИ

- Допълнителни задачи

Всяка редица от равни числа в едномерен сортиран масив се нарича площадка. Да се напише програма, която намира началото и дължината на най-дългата площадка в даден сортиран във възходящ ред масив.

Пример:

1, 2, 2, 3, 3, 4, 4, 4, 4, 4, 5, 5, 5 -> най – дългата площадка е 4 с дължина 5

1, 2, 3, 4, 5 -> няма най-дълга площадка



ВЪПРОСИ