

Вложени цикли



Определение

- Когато в тялото на един оператор за цикъл е вложен друг оператор за цикъл, говорим за **вложени цикли**. Единият се нарича **външен**, а другият – **вътрешен**.

Действие

- За всяко завъртане на външния цикъл вътрешния се извърта целия.

Пример за вложени цикли

Извеждане на таблицата за умножение до 10:

```
for(int i=1;i<=10;i++)  
    for(int j=1;j<=10;j++)  
        cout<<i<<'*<<j<<'='<<i*j<<endl;
```

Правила при вложени цикли

- ❑ Управляващите променливи на двата цикъла трябва да са различни
- ❑ Вътрешният цикъл трябва да се съдържа изцяло в тялото на външния
- ❑ От тялото на вътрешния цикъл можем да направим преход към тялото на външния, но не и обратно
- ❑ Вътрешният и външният цикъл може да са от един и същи тип, или от различни типове
- ❑ Променливи, декларирани в тялото на вътрешния цикъл, не са видими извън него

Да се въведе n и да изведе на екрана следното:

1

1 2

1 2 3

.....

.....

1 2 3 n

- `int n;` // декларираме променливите
- `cin>>n;` // въвеждаме числото
- `for(int i=1;i<=n;i++)` // цикъл за редовете
- `{` // за всеки ред ще изпълнява няколко неща
- `for(int j=1;j<=i;j++)` // цикъл за колоните
- `cout<<j;` // отпечатваме цифрата
- `cout<<endl;` // нов ред накрая на реда
- `}` // това се повтаря за всеки ред

Да се въведе n и да изведе на екрана следното:

1

2 3

3 4 5

4 5 6 7

.....

n $n+1$.. $2*n-1$

- `int n;` // декларираме променливите
- `cin>>n;` // въвеждаме числото
- `for(int i=1;i<=n;i++)` // цикъл за редовете
- `{` // за всеки ред ще изпълнява няколко неща
- `for(int j=i;j<2*i;j++)` // цикъл за колоните
- `cout<<j;` // отпечатваме цифрата
- `cout<<endl;` // нов ред накрая на реда
- `}` // това се повтаря за всеки ред

Въвеждане на числа до нула и отпечатване на броя на простите числа между тях

```
□ int chislo, br=0;           // декларираме променливите
□ bool prosto;               // флаг дали числото е просто
□ do {                       // повтаряме:
□   cin>>chislo;             // въвеждаме числото
□   prosto=true;             // предполагаме че е просто
□   for(int i=2;i<=chislo/2;i++) // проверяваме дали е така
□     if(chislo%i==0) {      // ако се дели на друго...
□       prosto=false;       // ...значи не е просто
□       break;              // прекъсваме проверката
□     }
□   if (prosto)&&(chislo!=0) br++; // броим само простите
□ } while (chislo!=0);       // докато не въведем 0
□ cout<<br<<endl;           // отпечатваме го
```

Въвеждане на k и n числа и намиране на k -цифрените между тях

```
□ int chislo, k, n, br;           // декларираме променливите
□ cin >> k >> n;                 // въвеждаме входните данни
□ for(int i=1; i<=n; i++) {      // повтаряме n пъти
□   cin >> chislo;                // въвеждаме числото
□   br=0;                        // инициализация на цикъла
□   while (chislo!=0) {           // докато не свърши числото
□       chislo/=10;              // изтриваме последната цифра
□       br++;                    // и я отброяваме
□   }
□   if (br==k)                   // ако е с k цифри...
□       cout << "това е!" << endl; // ...отпечатваме че е познато
□ }
```


Задачи

1. Да се напише програма, която позволява да се въвеждат числа до въвеждане на 0 и отпечатва числата от 1 до въведеното число.
2. Да се отпечатат с числа следните фигури (само че не до 3, а до произволно N):

```
* * *  
* * *  
* * *
```

```
1  
2 2  
3 3 3
```

```
1 1 1  
2 2 2  
3 3 3
```

```
1 2 3  
1 2 3  
1 2 3
```

```
3 3 3  
2 2 2  
1 1 1
```

```
1 2 3  
1 2  
1
```

```
1 2 3 4  
4 3 2 1  
1 2 3 4  
4 3 2 1
```

```
1 0 1 0  
0 1 0 1  
1 0 1 0  
0 1 0 1
```

Задачи

3. Да се отпечатаат всички числа в интервала $[a, b]$, по едно на ред. На същия ред да се отпечатаат и всички числа, на които даденото число се дели без остатък.
4. Да се напише програма, която при въвеждане на цели числа m и n , където $m \leq n$, извежда простите числа в интервала $[m, n]$; ако няма такива, да изведе подходящо съобщение.
3. Да се напише програма, която отпечата матрица за умножение на числа, подобна на долната, но не до 3, а до 10.
1 2 3 4

1 | 1 2 3 4
2 | 2 4 6 8
3 | 3 6 9 12
5. Да се напише програма, която извежда всички числа в интервала $(1..1000)$, сумата от цифрите на които е равна на X

Задачи

6. Напишете програма която извежда всички числа в интервала $[0..1000]$, които имат поне две равни цифри. Как може решението да бъде оптимизирано, така че да има минимален брой завъртания на циклите.
7. Да се напише програма, която позволява да се въведе брой предмети N . След това за всеки предмет позволява да се въведат оценките по него (до въвеждане на 0), след което извежда средния успех по този предмет. Да се направи програмата да извежда накрая и средния успех за срока.

Извода:



ОТ ТОЛКОВА ЦИКЛИ ВЕЧЕ ЗАЦИКЛИХМЕ