

Основы программирования (на языке Си)

Тема 9. Циклические алгоритмы

Что такое цикл?

Цикл – это многократное выполнение одинаковых действий.

Два вида циклов:

- цикл с **известным** числом шагов (сделать 10 раз)
- цикл с **неизвестным** числом шагов (делать, пока не надоест)

Задача. Вывести на экран 10 раз слово «Привет».



Можно ли решить известными методами?

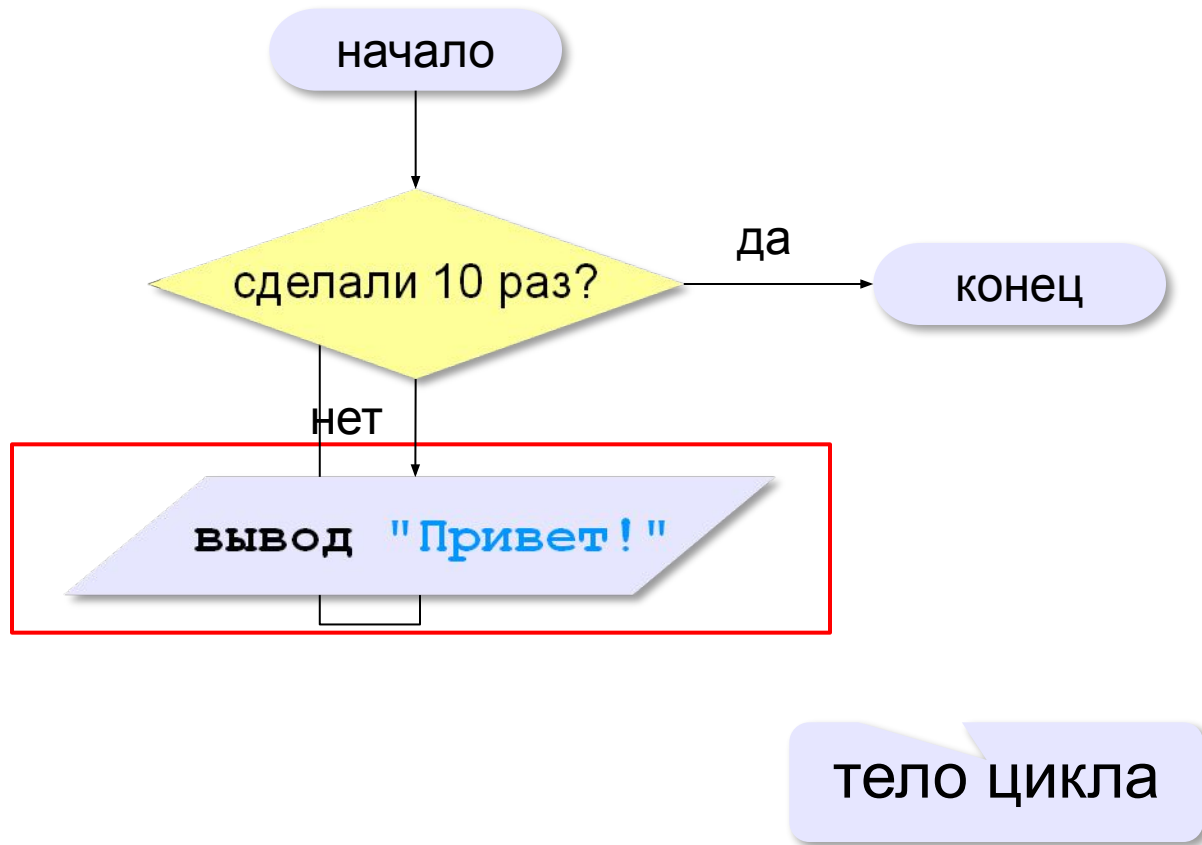
Повторения в программе

```
printf ("Привет\n" );  
printf ("Привет\n" );  
...  
printf ("Привет\n" );
```



Что плохо?

Блок-схема цикла



Как организовать цикл?

```
счётчик = 0  
пока счётчик < 10  
    printf("Привет\n");  
    увеличить счётчик на 1
```

результат операции
автоматически
сравнивается с нулём!

```
счётчик = 10  
пока счётчик > 0  
    printf("Привет\n");  
    уменьшить счётчик на 1
```



Какой способ удобнее для процессора?

Цикл с условием

Задача. Определить **количество цифр** в десятичной записи целого положительного числа, записанного в переменную n .

```
счётчик = 0
пока n > 0
    отсечь последнюю цифру n
    увеличить счётчик на 1
```

n	счётчик
1234	0

? Как отсечь последнюю цифру?

```
n = n / 10;
```

? Как увеличить счётчик на 1?

```
счётчик = счётчик + 1;
```

```
счётчик ++;
```

Цикл с условием

начальное значение
счётчика

условие
продолжения

заголовок
цикла

```
count = 0;  
while ( n > 0 )  
{  
    n = n / 10;  
    count ++;  
}
```

конец
цикла

тело цикла



Цикл с предусловием – проверка на входе в цикл!

Цикл с условием

При известном количестве шагов:

```
k = 0;
while ( k < 10 )
{
    printf ( "привет\n" );
    k ++;
}
```

Защипливание:

```
k = 0;
while ( k < 10 )
{
    printf ( "привет\n" );
}
```


Сколько раз выполняется цикл?

```
a = 4; b = 6;  
while ( a < b ) a = a + 1;
```

2 раза
a = 6

```
a = 4; b = 6;  
while ( a < b ) a = a + b;
```

1 раз
a = 10

```
a = 4; b = 6;  
while ( a > b ) a ++;
```

0 раз
a = 4

```
a = 4; b = 6;  
while ( a < b ) b = a - b;
```

1 раз
b = -2

```
a = 4; b = 6;  
while ( a < b ) a --;
```

защелкивание

Цикл с постусловием

заголовок
цикла

```
do
```

```
{
```

```
printf ("Введите n > 0: ");
```

```
scanf ( "%d", &n );
```

```
}
```

```
while ( n <= 0 );
```

тело цикла

условие
продолжения

- при входе в цикл условие **не проверяется**
- цикл всегда выполняется **хотя бы один раз**

Задачи

«А»: Напишите программу, которая получает два целых числа A и B ($0 < A < B$) и выводит квадраты всех натуральных чисел в интервале от A до B .

Пример:

Введите два целых числа :

10 12

$10 * 10 = 100$

$11 * 11 = 121$

$12 * 12 = 144$

«В»: Напишите программу, которая получает два целых числа и находит их произведение, не используя операцию умножения. Учтите, что числа могут быть отрицательными.

Пример:

Введите два числа :

10 -15

$10 * (-15) = -150$

Задачи

«С»: Ввести натуральное число N и вычислить сумму всех чисел Фибоначчи, меньших N . Предусмотрите защиту от ввода отрицательного числа N .

Пример:

Введите число N :

10000

Сумма 17709

Задачи-2

«А»: Ввести натуральное число и найти сумму его цифр.

Пример:

Введите натуральное число:

12345

Сумма цифр 15.

«В»: Ввести натуральное число и определить, верно ли, что в его записи есть две одинаковые цифры, стоящие рядом.

Пример:

Введите натуральное число:

12342

Нет.

Пример:

Введите натуральное число:

12245

Да.

Задачи-2

«С»: Ввести натуральное число и определить, верно ли, что в его записи есть две одинаковые цифры (не обязательно стоящие рядом).

Пример:

Введите натуральное число:

12342

Да .

Пример:

Введите натуральное число:

12345

Нет .

Цикл с переменной

Задача. Вывести все степени двойки от 2^1 до 2^{10} .



Можно ли сделать с циклом «пока»?

```
k = 1;
n = 2;
while ( k <= 10 )
{
    printf ("%d\n", n) ;
    n *= 2;
    k ++;
}
```

```
n = 2;
for ( k=1; k<=10; k++ )
{
    printf ("%d\n", n) ;
    n *= 2;
}
```

ЦИКЛ С
переменной

Цикл с переменной: другой шаг

```
for ( k = 10; k >= 1; k-- )  
    printf( "%d\n", k*k );
```



Что получится?

```
for ( k = 1; k <= 10; k += 2 )  
    printf( "%d\n", k*k );
```

100
81
64
49
36
25
16
9
4
1

1
9
25
49
81

Сколько раз выполняется цикл?

```
a = 1;  
for ( i = 1; i <= 3; i++ ) a = a + 1;
```

a = 4

```
a = 1;  
for ( i = 3; i <= 1; i++ ) a = a + 1;
```

a = 1

```
a = 1;  
for ( i = 1; i <= 3; i-- ) a = a + 1;
```

a = 1

```
a = 1;  
for ( i = 3; i >= 1; i-- ) a = a + 1;
```

a = 4

Задачи

«А»: Найдите все пятизначные числа, которые при делении на 133 дают в остатке 125, а при делении на 134 дают в остатке 111.

«В»: Натуральное число называется **числом Армстронга**, если сумма цифр числа, возведенных в N-ную степень (где N – количество цифр в числе) равна самому числу. Например, $153 = 1^3 + 5^3 + 3^3$.
Найдите все трёхзначные Армстронга.

Задачи

«С»: Натуральное число называется автоморфным, если оно равно последним цифрам своего квадрата. Например, $25^2 = 625$. Напишите программу, которая получает натуральное число N и выводит на экран все автоморфные числа, не превосходящие N.

Пример:

Введите N:

1000

$$1 * 1 = 1$$

$$5 * 5 = 25$$

$$6 * 6 = 36$$

$$25 * 25 = 625$$

$$76 * 76 = 5776$$

Вложенные циклы

Задача. Вывести все простые числа в диапазоне от 2 до 1000.

```
сделать для n от 2 до 1000
  если число n простое то
    вывод n
```

нет делителей [2.. n-1]:
проверка в цикле!



Что значит «простое число»?

Вложенные циклы

```
for ( n = 2; n <= 1000; n++ )  
{  
    count = 0;  
    for ( k = 2; k < n; k++ )  
        if ( n % k == 0 )  
            count++;  
    if ( count == 0 )  
        printf ( "%d\n", n );  
}
```

ВЛОЖЕННЫЙ ЦИКЛ

Вложенные циклы

```

for ( n = 1; n <= 4; n++ )
{
  for ( k = 1; k <= i; k++ )
  {
    ...
  }
}

```

n	k
1	1
2	1
2	2
3	1
3	2
3	3
4	1
4	2
4	3
4	4



Как меняются переменные?



Переменная внутреннего цикла изменяется быстрее!

Поиск простых чисел – как улучшить?

$$n = k \cdot m, \quad k \leq m \Rightarrow k^2 \leq n \Rightarrow k \leq \sqrt{n}$$

```
while ( k <= sqrt(n) )
{
  ...
}
```



Что плохо?

```
count = 0;
k = 2;
while ( k*k <= n )
{
  if ( n % k == 0 ) count++;
  k++;
}
```



Как ещё улучшить?

```
while ( k*k <= n && (count == 0) ) {
  ...
}
```

Задачи

«А»: Напишите программу, которая получает натуральные числа A и B ($A < B$) и выводит все простые числа в интервале от A до B .

Пример:

Введите границы диапазона:

10 20

11 13 17 19

«В»: В магазине продается мастика в ящиках по 15 кг, 17 кг, 21 кг. Как купить ровно 185 кг мастики, не вскрывая ящики? Сколькими способами можно это сделать?

Задачи

«С»: Ввести натуральное число N и вывести все натуральные числа, не превосходящие N и делящиеся на каждую из своих цифр.

Пример:

Введите N :

15

1 2 3 4 5 6 7 8 9 11 12 15