Программирование на языке C#

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».

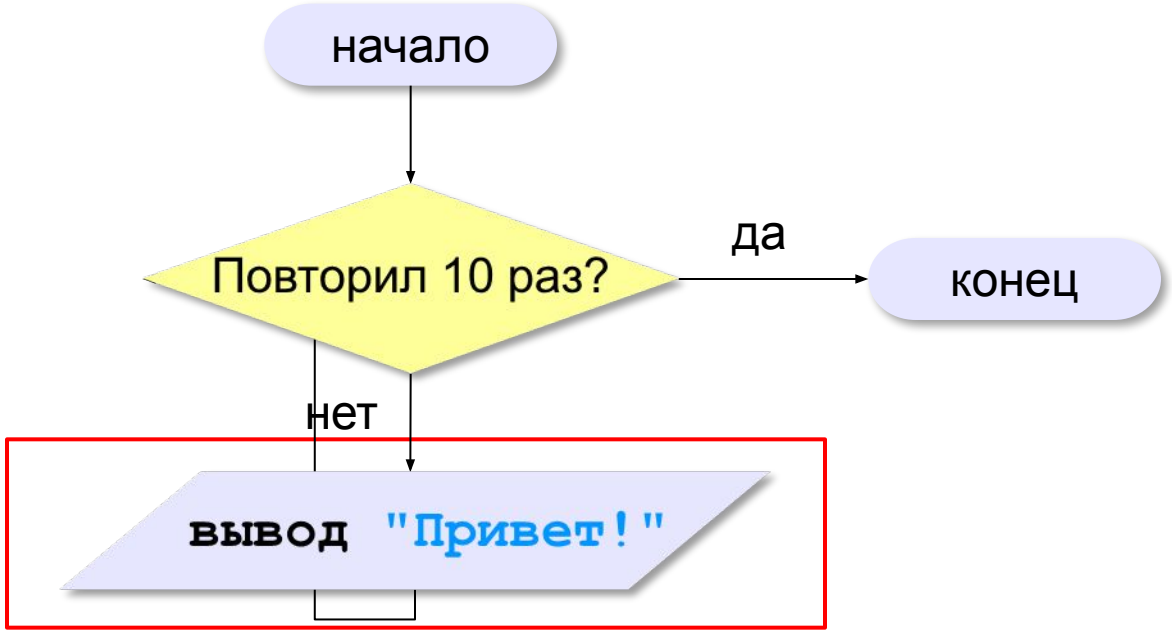
Можно ли решить известными методами?

Повторения в программе

```
Console.WriteLine("Привет");
Console.WriteLine("Привет");
...
Console.WriteLine("Привет");
```

Что плохо?

Блок-схема цикла



тело цикла

Как организовать цикл?

```
счётчик = 0
пока счётчик < 10
   вывести "Привет\n";
   увеличить счётчик на 1
```

```
счётчик = 10
пока счётчик > 0
   вывести "Привет\n";
   уменьшить счётчик на 1
```

Оператор цикла while

```
while (условие) оператор;
```

условие: любое выражение, которое возвращает тип данных **bool**

оператор: одна или блок инструкций внутри фигурных скобок {}

Цикл с условием

При известном количестве шагов:

```
k = 0;
while (k < 10)
{
  Console.WriteLine("привет");
  k++;
}
```

Зацикливание:

```
k = 10;
while (k > 0)
{
    Console.WriteLine("привет");
}
```

Цикл с условием

Задача. Найти сумму целых чисел от 1 до **n**.

```
sum = 0
счётчик = 1
пока счётчик <= n
   увеличить sum на счётчик
   увеличить счётчик на 1
```

sum	счётчик
0	1

? Как увеличивать сумму?

```
sum = sum + счётик;
```

? Как увеличить счётчик на 1?

```
счётчик = счётчик + 1;
```

```
счётчик ++;
```

Цикл с условием

начальное значение

условие продолжения

заголовок цикла

```
count = 1;
while ( count <= n
{
   sum += count;
   count++;
}
```

конец цикла

тело цикла

Цикл с предусловием – проверка на входе в цикл!

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 --;
```

зацикливание

Упражнение 1:

Напишите программу, которая получает два целых положительных числа и находит их произведение, **не используя** операцию умножения.

Пример:

```
Введите два числа:
10 15
10 * 15 = 150
```

Цикл с постусловием

заголовок цикла

тело цикла

```
do
{
    Console.Write("Введите n > 0: ");
    n = Convert.ToInt32(Console.ReadLine());
}
while ( n <= 0 );
```

условие продолжения

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

Оператор цикла for

```
for (инициализация; условие; итерация)
    оператор;
```

```
инициализация;
while (условие)
{
    оператор;
    итерация;
}
```

Оператор цикла for

Задача. Найти сумму целых чисел от 1 до **n**.

```
int sum = 0;
for (int i=1; i<=n; i++)
{
  sum += i;
}
```

Цикл с переменной: другой шаг

```
for ( k = 10; k >= 1; k-- )
  Console.WriteLine(k*k);
```

Что получится?

```
100
81
64
49
36
25
16
9
4
1
```

```
for ( k = 1; k <= 10; k += 2 )
  Console.WriteLine(k*k);
```

```
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 = 5; i >= 0; i-=3 ) a = a + 1;
```

a = 3

Упражнение 2:

Ввести целое положительное число *n* и определить является ли число простым (не имеют делителей, кроме 1 и *n)*. Использовать оператор **for**

Пример:

```
Введите целое число:
10
Составное
```

Пример:

```
Введите целое число:
7
Простое
```

Цикл с переменной: другой шаг

```
for ( k = 10; k >= 1; k-- )
  Console.WriteLine(k*k);
```

Что получится?

```
100
81
64
49
36
25
16
9
4
1
```

```
for ( k = 1; k <= 10; k += 2 )
  Console.WriteLine(k*k);
```

```
1
9
25
49
81
```

Цикл с переменной: другой шаг

```
for ( k = 10; k >= 1; k-- )
  Console.WriteLine(k*k);
```

Что получится?

```
100
81
64
49
36
25
16
9
4
1
```

```
for ( k = 1; k <= 10; k += 2 )
  Console.WriteLine(k*k);
```

```
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

Вложенные циклы

Задача. Вывести в консоль таблицу умножения

```
for (int i = 1; i < 10; i++ )
{
  Console.WriteLine($"Для {i}");

  for (int j = 1; j < 10; j++ )
  {
    Console.WriteLine($"{i}*{j} = {i*j}");
  }
}
```

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 )
    Console.WriteLine(n);
}
```

вложенный цикл

Вложенные циклы

```
for ( i = 1; i <= 4; i++ )
{
   for ( k = 1; k <= i; 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?

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

```
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 - 1

«A»: 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
```

«B»: 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 - 2

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

Пример:

```
Введите натуральное число:
12345
Сумма цифр 15.
```

«D»: 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 - 3

«E»: 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.
```