

Богатов Р.Н.

# Программирование на языке высокого уровня

---

C++ ► Лекция 3.5 ► Примеры задач на циклы  
с заранее неизвестным числом повторений

Кафедра АСОИУ ОмГТУ, 2016

# Разбиение числа на цифры

```
// Задача: разбить десятичную запись числа на цифры (порядок не
// Задача: вычислить сумму цифр десятичной записи числа.
// Задача: определить максимальную цифру в десятичной записи числа.

#include <stdio.h>
#include <conio.h>
#include <locale.h>

void main()
{
    setlocale(0, ".1251");
    printf("Введите натуральное число N:");
    int n;
    scanf("%d", &n);

    int max = -1;
    do
    {
        int x = n % 10;
        printf("%d ", x);
        if (x > max) max = x;
        n /= 10;
    } while (n!=0);

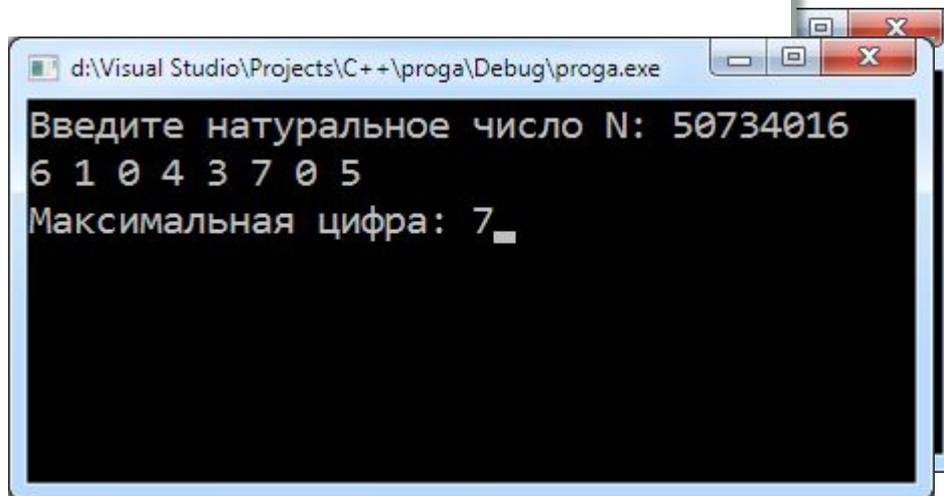
    printf("\nМаксимальная цифра: %d", max);

    _getch();
}
```

Добавь  
до цикла

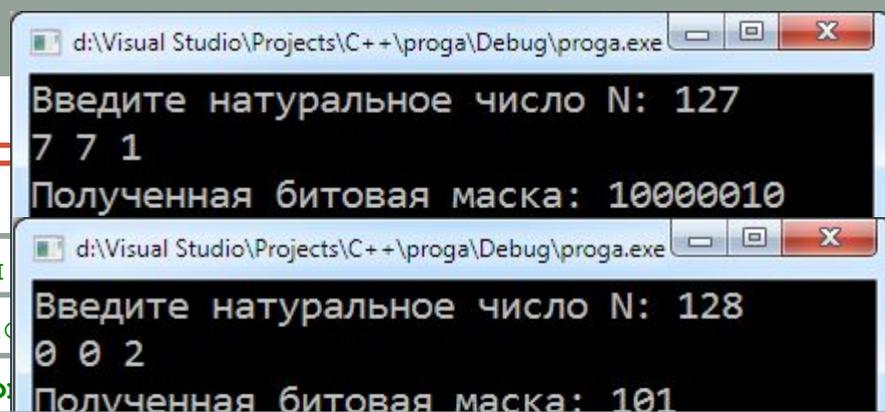
Добавь  
в цикле

Добавь  
после  
цикла



```
d:\Visual Studio\Projects\C++\proga\Debug\proga.exe
Введите натуральное число N: 50734016
6 1 0 4 3 7 0 5
Максимальная цифра: 7_
```

# Разбиение числа на цифры



```
d:\Visual Studio\Projects\C++\proga\Debug\proga.exe
Введите натуральное число N: 127
7 7 1
Полученная битовая маска: 10000010

d:\Visual Studio\Projects\C++\proga\Debug\proga.exe
Введите натуральное число N: 128
0 0 2
Полученная битовая маска: 101
```

```
// Задача: определить, содержит ли троичная запись числа нуль
// Задача: определить, все ли цифры троичной записи числа различны
// Задача: определить, содержит ли восьмеричная запись числа нуль
// Задача: определить, все ли цифры восьмеричной записи числа различны.

...
int m = 0; // битовая маска
do
{
    int x = n % 8;
    printf("%d ", x);
    if ((m & (1<<x)) > 0) // если есть бит в разряде x (цифра x уже встречалась)
    {
        m = -1; // сигнальное значение
        break;
    }
    m = m | (1<<x);
    n /= 8;
} while (n!=0);

if (m < 0)
    printf("\nЦифры в восьмеричной записи числа повторяются!");
else
    printf("\nВсе цифры восьмеричной записи числа различны!");
...

```

# Перевод числа в другую систему счисления

```
#include <stdio.h>
#include <conio.h>
#include <locale.h>

void main()
{
    setlocale(0, ".1251");
    printf("Десятичное число: ");
    int n; scanf("%d", &n);
    printf("Основание системы счисления: ");
    int r; scanf("%d", &r);

    int z = 0, d = 1;
    do
    {
        int x = n % r;
        z += x*d;
        d *= 10;
        n /= r;
    } while (n!=0);

    printf("%d-ичная запись числа: %d", r, z);

    _getch();
}
```

Так как основной алгоритм в обратном порядке, то будем копировать и умножать старшие разряды с помощью цикла. Мы последовательно принимаем

Пример: если  $z = 345$ ,  $d = 100$ , то  $z += x*d$  сделает  $z$  равным 2345. Мы умножим  $z$  на 10, чтобы следующую цифру до

```
d:\Visual Studio\Projects\C++\proga\Debug\...
Десятичное число: 15
Основание системы счисления: 16
16-ичная запись числа: 15_?
```

```
d:\Visual Studio\Projects\C++\proga\Debug\...
Десятичное число: 1024
Основание системы счисления: 2
2-ичная запись числа: 1410065408_?
```

Так как результат копится в переменной типа `int`, то это накладывает ограничения: основание системы счисления не может быть больше 10, а результат не может содержать более 8-9 цифр.

(Чтобы устранить эти недостатки, результат нужно копировать не в переменную, а в массиве или строке, что мы освоим, но позже.)

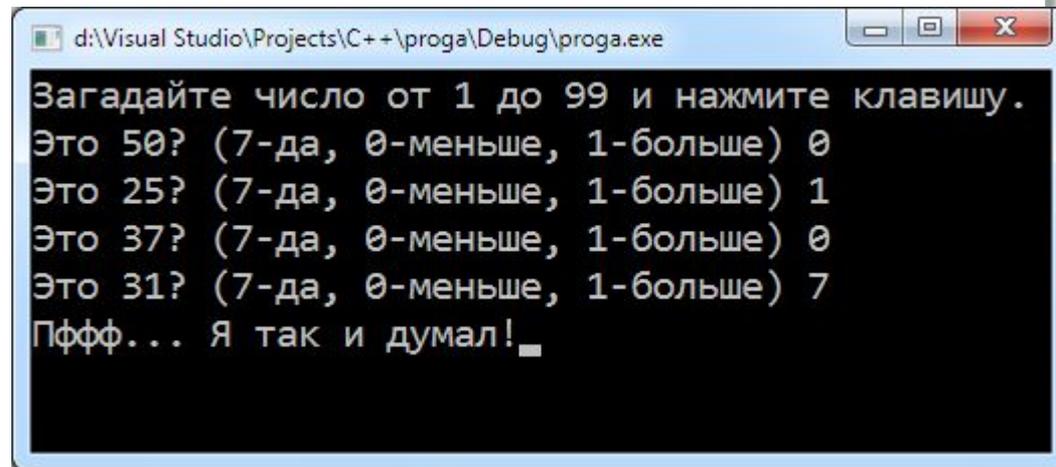
```
10-ичная запись числа: 2071934528_
```

# Игра «Загадай число!»

```
...
printf("Загадайте число от 1 до 99 и нажмите клавишу.\n" );
_getch();

int a = 1, b = 100; // интервал догадки
for(;;)
{
    int c = a + (b - a) / 2; // вычисление догадки
    printf("Это %d? (7-да, 0-меньше, 1-больше) " , c );
    int n;
    scanf("%d", &n);

    if (n == 7)
        break;
    else
        if (n == 0)
            b = c;
        else
            a = c;
}
printf("Пффф... Я так и думал!");
_getch();
...
```



```
d:\Visual Studio\Projects\C++\proga\Debug\proga.exe
Загадайте число от 1 до 99 и нажмите клавишу.
Это 50? (7-да, 0-меньше, 1-больше) 0
Это 25? (7-да, 0-меньше, 1-больше) 1
Это 37? (7-да, 0-меньше, 1-больше) 0
Это 31? (7-да, 0-меньше, 1-больше) 7
Пффф... Я так и думал!
```

## Упражнения для самостоятельной работы

- Проверьте, является ли заданное натуральное число палиндромом (остаётся таким же после записи в обратном порядке).
- Дано натуральное число  $N$ . Определите количество значимых цифр в пятеричной записи числа  $N$ .
- Дано натуральное число. Замените в нем все "единицы" на "пятёрки".
- Выведите все натуральные делители заданного числа.
- Определите, является ли заданное натуральное число «совершенным», т.е. равным сумме всех своих натуральных делителей, кроме самого этого числа (например, совершенное число  $6 = 1 + 2 + 3$ ).
- Проверьте, является ли заданное натуральное число простым (простые числа делятся без остатка только на себя и на единицу).
- От квадрата с единичной стороной отрезают квадраты со сторонами  $1/2$ ,  $1/3$ ,  $1/4$  и так далее, повторяя процесс до бесконечности. Требуется найти площадь оставшейся фигуры. Решите задачу численно с требуемой точностью.
- Вычислите сумму цифр данного *действительного* числа.
- Найдите наибольший общий делитель двух натуральных чисел по алгоритму Евклида без использования рекурсии.
- Дано натуральное число  $N$ . Определите минимальное основание системы счисления, в представлении которой все цифры этого числа будут различными.