

Лекция 8. Операторы условных и безусловных переходов языка Си

8.1. Оператор *if*

Для организации условных и безусловных переходов в программе на языке Си используются операторы: *if - else*, *switch* и *goto*.

Оператор *if-else* записывается следующим образом:

```
if (проверка_условия)
{
    оператор_1;
}
else
{
    оператор_2;
}
```

Если условие в скобках принимает истинное значение, выполняется *оператор_1*, если ложное - *оператор_2*. Если необходимо выполнить только один оператор, то их можно не заключать в фигурные скобки. В операторе *if* слово *else* может отсутствовать.

В операторе *if - else* непосредственно после ключевых слов *if* и *else* должны следовать другие операторы. Если хотя бы один из них является оператором *if*, его называют вложенным. Согласно принятому в языке Си соглашению слово *else* всегда относится к ближайшему предшествующему ему *if*.

Пример 1. Пример программы сравнения двух чисел, введенных с клавиатуры, с выводом результата на экран.

```
#include <stdio.h>                                //подключение заголовочного файла
#include <stdlib.h>                                //для перехода на русский язык

int a, b;                                         //объявление переменных типа integer

void main()                                     //основной цикл
{
    system("chcp 1251");                         //начало основного цикла
    system("cls");                               //переходим в консоли на русский язык
    printf("Введите число 1: ");                //очищаем окно консоли
    scanf_s("%d", &a);                           //выводим в консоль фразу
    printf("Введите число 2: ");                //считывание числа в переменную a
    scanf_s("%d", &b);                           //выводим в консоль фразу
                                                //считывание числа в переменную b

    if (a > b)
    {
        printf("Первое число больше\n");
    }
    else
    {
        if (a == b)
```

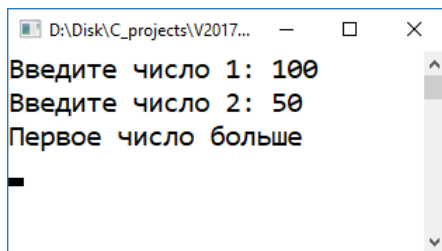
```

    {
        printf("Числа равны\n");
    }
    else
    {
        printf("Второе число больше\n");
    }
}

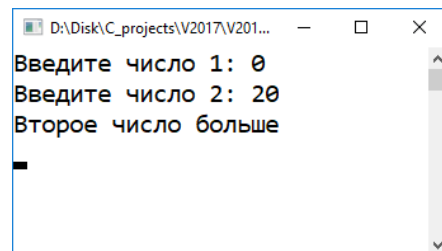
getchar();getchar();getchar();
} //конец основного цикла

```

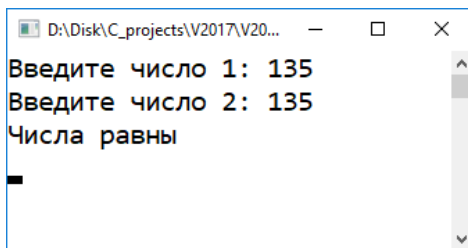
Ниже представлены результаты выполнения программы для разных вариантов введенных чисел (рис. 1).



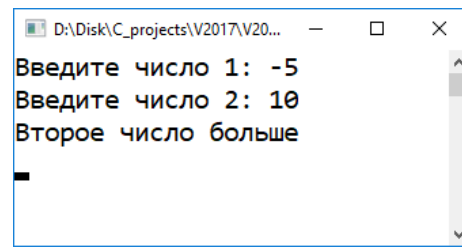
а



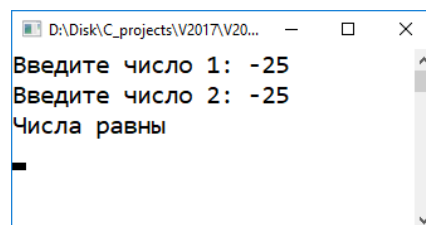
б



в



г



д

Рис. 1. Программа сравнения двух чисел, введенных с клавиатуры, с выводом результата на экран: а) первое число – 100, второе число – 50; б) первое число – 0, второе число – 20; в) первое число – 135, второе число – 135; г) первое число – -5, второе число – 10; д) первое число – -25, второе число – -25

8.2. Оператор switch

Оператор **switch** позволяет выбрать одну из нескольких альтернатив. Он записывается в следующем формальном виде:

```

switch (выражение)
{

```

```

        case константа_1: операторы_1;
        break;
        case константа_2: операторы_2;
        break;
        default: операторы_default;
    }

```

Здесь вычисляется значение целого выражения в скобках (его иногда называют селектором) и оно сравнивается со всеми константами (константными выражениями). Все константы должны быть различными. При совпадении выполнится соответствующий вариант операторов (один или несколько операторов). Вариант с ключевым словом **default** реализуется, если ни один другой не подошел (слово **default** может и отсутствовать). Если **default** отсутствует, а все результаты сравнения отрицательны, то ни один вариант не выполняется.

Для прекращения последующих проверок после успешного выбора некоторого варианта используется оператор **break**, обеспечивающий немедленный выход из переключателя **switch**.

Пример 2. Пример программы определения равенства двух чисел, введенных с клавиатуры, с выводом результата на экран.

```

#include <stdio.h>                                //подключение заголовочного файла
#include <stdlib.h>                                //для перехода на русский язык

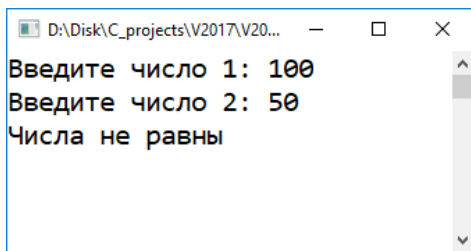
int a, b, c;                                     //объявление переменных типа integer

void main()                                     //основной цикл
{
    system("chcp 1251");                          //начало основного цикла
    system("cls");                                //переходим в консоли на русский язык
    printf("Введите число 1: ");                  //очищаем окно консоли
    scanf_s("%d", &a);                            //выводим в консоль фразу
    printf("Введите число 2: ");                  //считывание числа в переменную a
    scanf_s("%d", &b);                            //выводим в консоль фразу
    c = a - b;                                    //считывание числа в переменную b
    switch (c)                                    //вычисляем разницу
    {
        case 0: printf("Числа равны\n");
        break;
        default: printf("Числа не равны\n");
    }

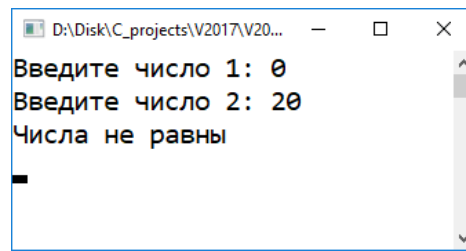
    getchar(); getchar();
}                                                  //конец основного цикла

```

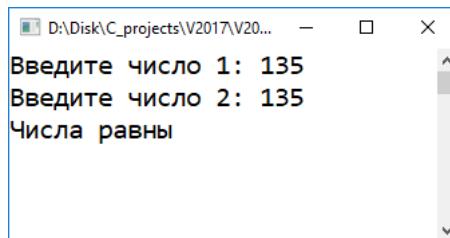
На рис. 2 представлены результаты выполнения программы для разных вариантов введенных чисел.



a



б



в

Рис. 2. Программа сравнения двух чисел, введенных с клавиатуры, с выводом результата на экран: а) первое число – 100, второе число – 50; б) первое число – 0, второе число – 20; в) первое число – 135, второе число – 135

8.3. Оператор *goto*

Рассмотрим правила выполнения безусловного перехода, который можно представить в следующей форме:

***goto* метка;**

Метка - это любой идентификатор, после которого поставлено двоеточие. Оператор ***goto*** указывает на то, что выполнение программы необходимо продолжить, начиная с оператора, перед которым записана метка. Метку можно поставить перед любым оператором в той функции, где находится соответствующий ей оператор ***goto***. Метку не надо объявлять.

Пример 3. Пример программы расчета значения $s = \sum_{n=1}^x \frac{n}{n+1}$. Значение x

вводится с клавиатуры.

```
#include <stdio.h>
#include <stdlib.h>
```

//подключение заголовочного файла
//для перехода на русский язык

```
float n,x;
float s;
```

//объявление переменных типа float
//объявление переменных типа float

```
void main()
{
    system("chcp 1251");
    system("cls");
    printf("Введите число x: ");
    scanf_s("%f", &x);
```

//основной цикл
//начало основного цикла
//переходим в консоли на русский язык
//очищаем окно консоли
//выводим в консоль фразу
//считывание числа в переменную x

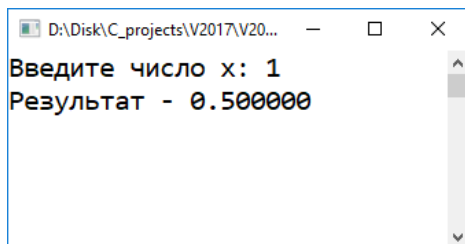
```

m1: ++n;
    if (n <= x)
    {
        s += n/(n+1);
        goto m1;
    }

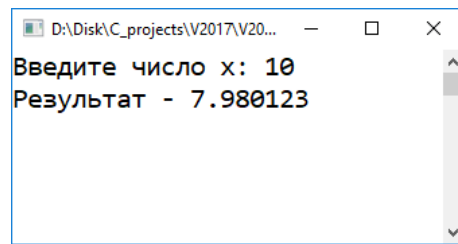
printf("Результат - %f\n",s);
getchar(); getchar();
}                                     //конец основного цикла

```

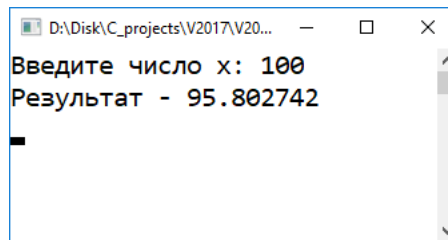
На рис. 3 представлены результаты выполнения данной программы для различных значений переменной x (1, 10 и 100).



a



б



в

Рис. 3. Программа расчета значения $s = \sum_{n=1}^x \frac{n}{n+1} \therefore$ а) $x=1$; б) $x=10$; в) $x=100$