

Программирование
Программирование и
алгоритмизация
Основы программирования
Введение в специальность

Лекция 3.

Си – простые типы данных,
развилки, цикл DO WHILE, БЛОК
СХЕМЫ!

Что будем изучать на курсе?

- 1) Основы основ – Си, VS, if, do while
- 2) Простейшая (статическая) графика в WinAPI. (Функции, рекурсия)
- 3) Динамическая графика (игры) – (циклы, switch, случайные числа, таймер)
- 4) Массивы – одномерные и двумерные. В играх – естественно.
- 5) Структуры, файлы, строки (в том числе и в играх)
- 6) * Память, указатели, динамическая память.

Основные типы данных

Тип	Длина байт	Диапазон значений	Минимальное значение	Максимальное значение
char	1	$2^8 = 256$	-128	+127
short	2	$2^{16} = 65\,536$	-32 768	32767
long	4	$2^{32} = 4,294,967,296$	-2,147,483,648	+2,147,483,647
float	4	IEEE 754 standard	$1.175494351 \times 10^{-38}$	3.4028234×10^{38}
double	8	IEEE 754 standard	2.22507e-308	1.79769e+308

int = short/long

<https://ru.wikipedia.org/wiki/%D0%A1%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0%D1%82%D0%B8%D0%BF%D0%BE%D0%B2%D0%A1%D0%B8>

Строка форматирования

Тип	scanf/printf
char	%c
short	%hi
int	%d или %i
long	%li
float	%f
double	%lf
long double	%Lf

http://youngcoder.ru/lessons/3/formatnyi_vyvod_printf.php

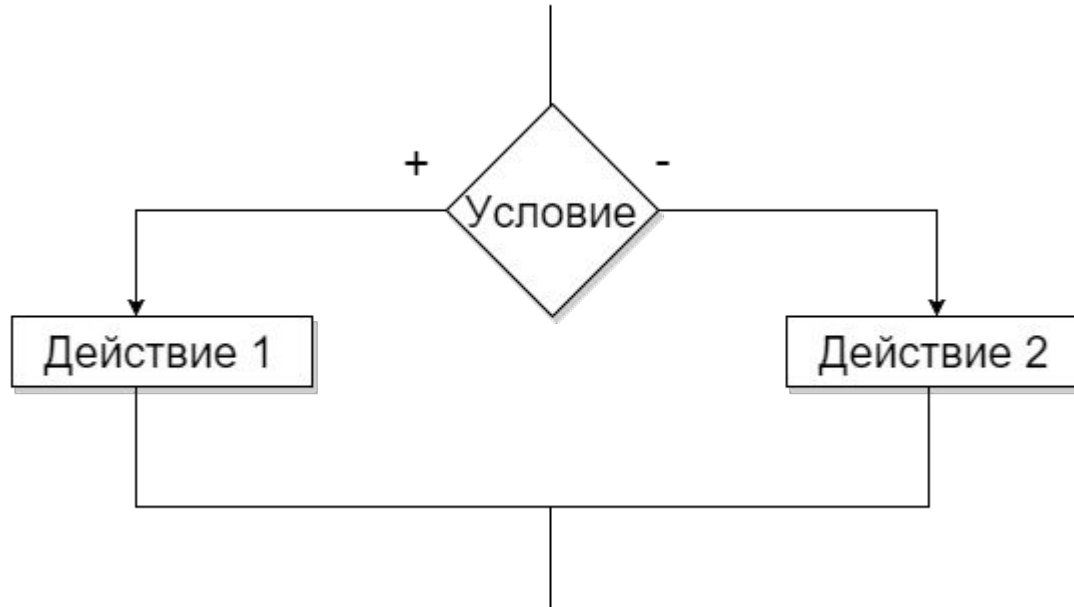
<http://www.c-cpp.ru/content/printf>

<https://ru.wikipedia.org/wiki/Printf>

Консоль – что из себя представляет.

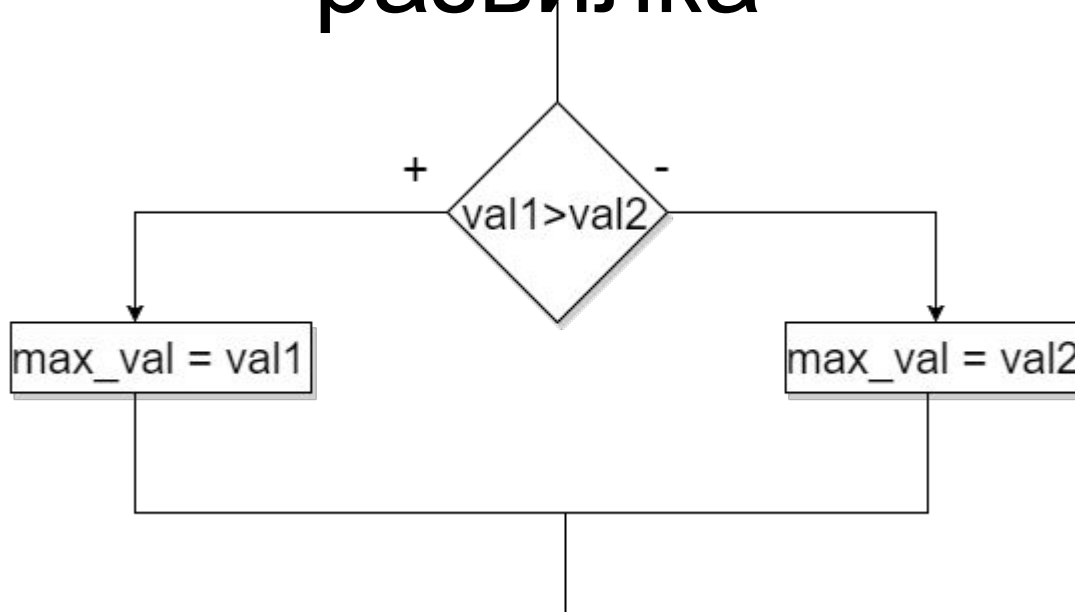
Знакомство – что это такое.

Развилка (if)



```
if (Условие)  
    Действие1;  
else  
    Действие2;
```

Найти максимум - полная развилка

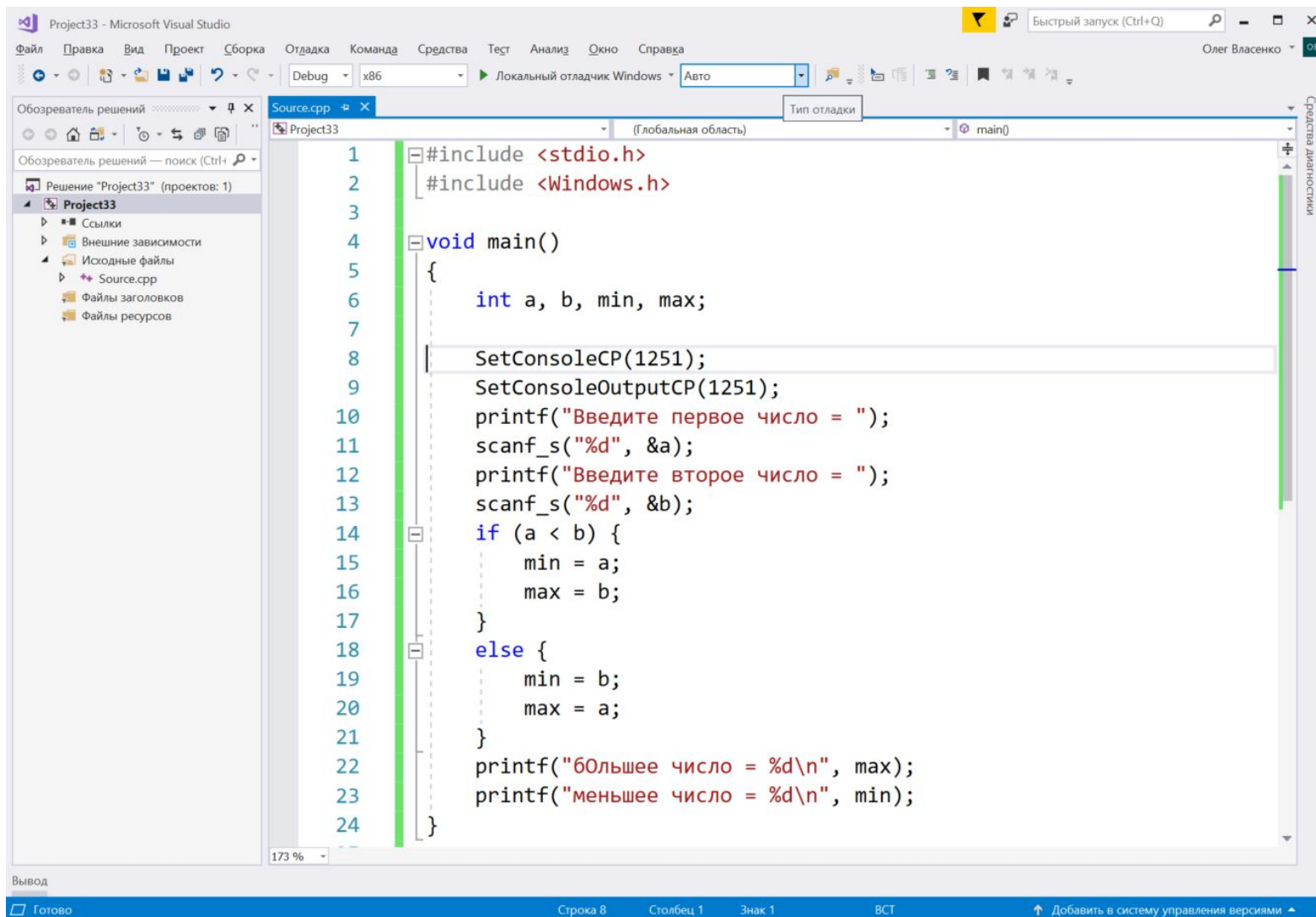


Фрагмент кода:

```
if (val1 > val2) {  
    max_val = val1;  
} else {  
    max_val = val2;  
}
```

Задача. Какое число больше? (1)

Ввести два числа. Вывести БОльшее число. И вывести меньшее число.

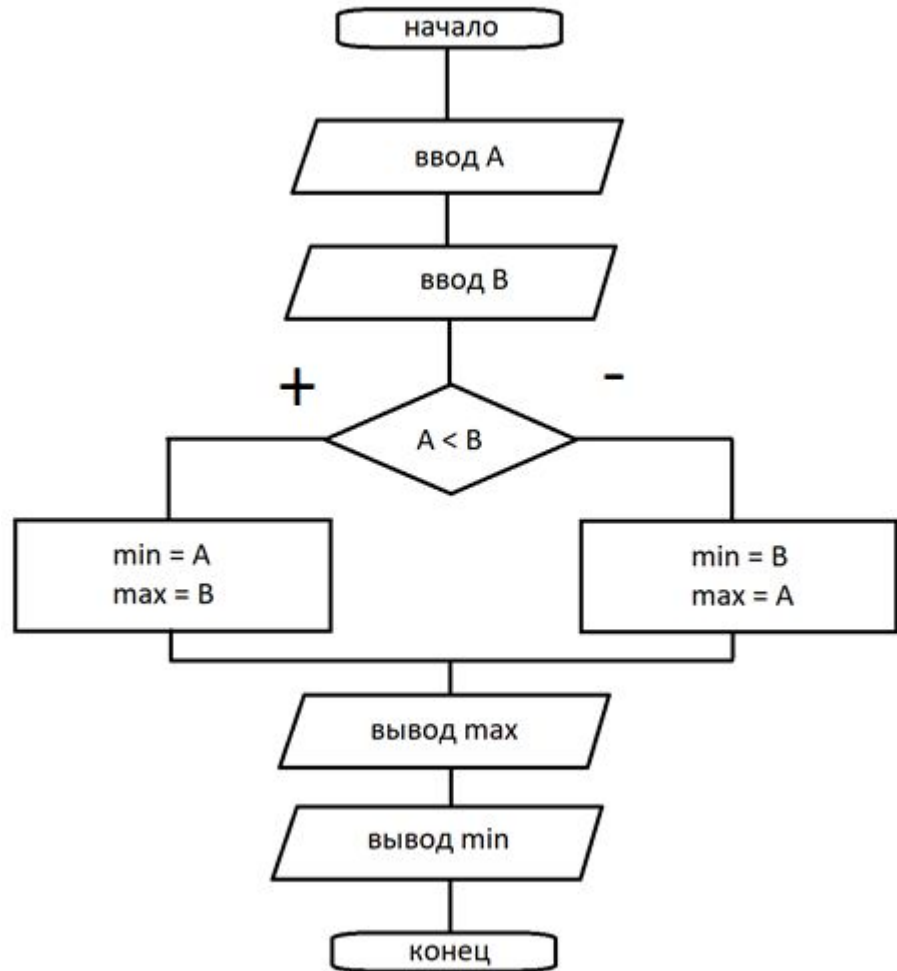


The screenshot shows the Microsoft Visual Studio IDE with a C++ source file named 'Source.cpp'. The code implements a program to find the maximum and minimum of two integers. The program includes `<stdio.h>` and `<windows.h>`. It defines a `main` function that declares variables `a`, `b`, `min`, and `max`. It sets the console title to '1251' and outputs the prompt 'Введите первое число = '. It then reads the first number into `a` using `scanf_s`. Similarly, it prompts for and reads the second number into `b`. An `if` statement checks if `a < b`. If true, `min` is set to `a` and `max` to `b`. Otherwise, `min` is set to `b` and `max` to `a`. Finally, it prints the maximum value with the prompt 'БОльшее число = ' and the minimum value with the prompt 'меньшее число = '.

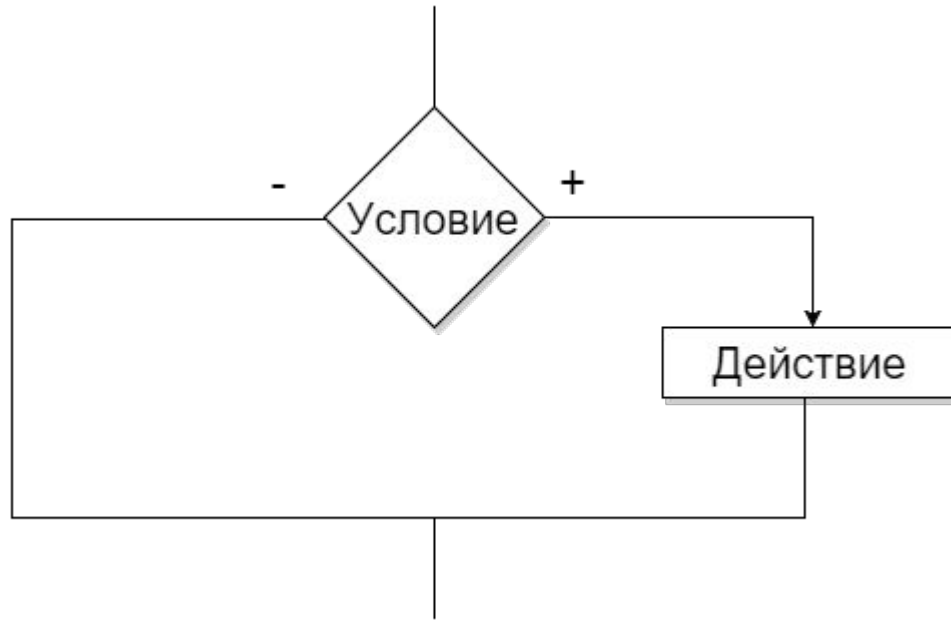
```
1 #include <stdio.h>
2 #include <windows.h>
3
4 void main()
5 {
6     int a, b, min, max;
7
8     SetConsoleCP(1251);
9     SetConsoleOutputCP(1251);
10    printf("Введите первое число = ");
11    scanf_s("%d", &a);
12    printf("Введите второе число = ");
13    scanf_s("%d", &b);
14    if (a < b) {
15        min = a;
16        max = b;
17    }
18    else {
19        min = b;
20        max = a;
21    }
22    printf("БОльшее число = %d\n", max);
23    printf("меньшее число = %d\n", min);
24 }
```

Задача. Какое число больше? (2)

Блок схема для программы

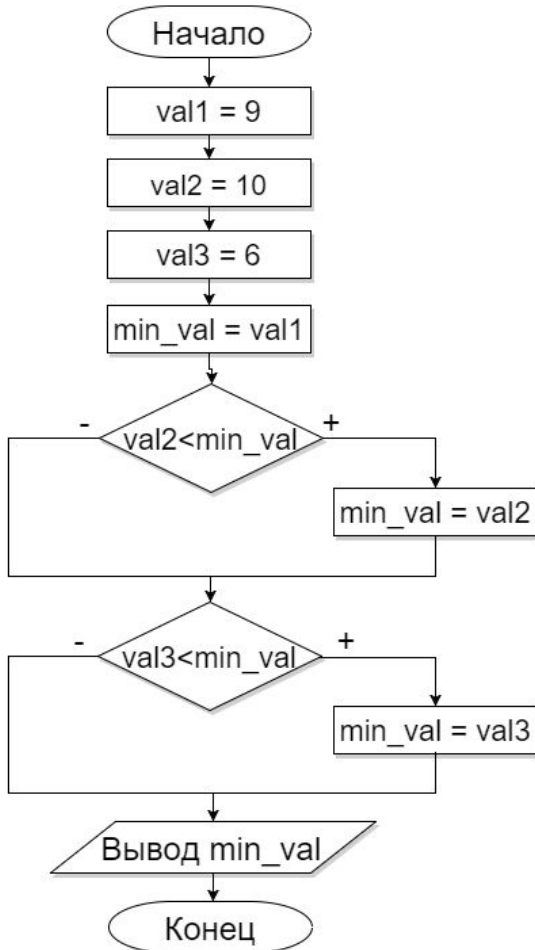


Усеченная развилка



```
if (Условие) {  
    Действие;  
}
```

Минимум из 3 чисел



```
void main() {
```

```
int val1 = 9;
```

```
int val2 = 10;
```

```
int val3 = 6;
```

```
int min_val = val1; // берем за минимальный val1
```

```
if (val2 < min_val) { // если второе меньше  
    min_val = val2; // то теперь минимальное val2  
}
```

```
if (val3 < min_val) { // если третье меньше  
    min_val = val3; // то теперь минимальное val3  
}
```

```
printf("min_val = %i", min_val);
```

```
}
```

Логические операции

Оператор	Описание
&&	Логическое И (AND)
	Логическое ИЛИ (OR)
!	Логическое унарное НЕ (NOT)

A	!A
0	1
1	0

A	B	A && B	A B
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	1

```
if (time < 7.00 || day >= 6) rest();
```

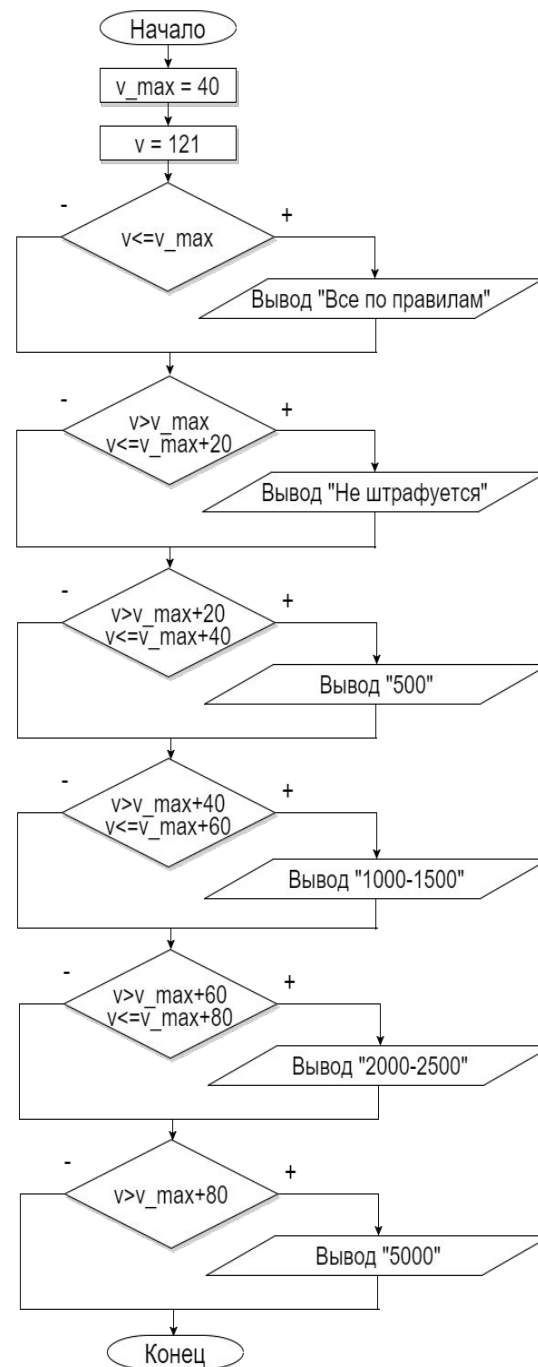
```
if (!closed && money > 1000) eat();
```

Штраф за превышение скорости

```
void main() {
    int v_max = 40; // Максимально разрешенная скорость
    int v = 70; // Реальная скорость!

    if (v <= v_max) {
        printf("Все по правилам!");
    }
    if ((v > v_max) && (v <= v_max + 20)) {
        printf("не штрафуются");
    }
    if ((v > v_max + 20) && (v <= v_max + 40)) {
        printf("500");
    }
    if ((v > v_max + 40) && (v <= v_max + 60)) {
        printf("1000-1500");
    }
    if ((v > v_max + 60) && (v <= v_max + 80)) {
        printf("2000-2500");
    }
    if (v > v_max + 80) {
        printf("5000");
    }
}
```

Блок схема



Штраф за превышение скорости - тесты

тесты, чтобы проверить все ситуации (при разрешенной скорости 40 км/час):

№	Что выводим	Вход
1	Все по правилам	38
2	Не штрафуются	59
3		
4		
5		
6		

Зачем нужны блок схемы?

- 1) Зачем учиться рисовать блок схемы, если их нельзя скомпилировать и выполнить? В них нельзя автоматически найти ошибки?
- 2) Зачем учиться рисовать блок схемы, если программа длиннее 100 строк ужасно выглядит в виде блок схемы и абсолютно нечитаема?
- 3) И зачем тратить наше бесценное время?