

## Операторы ветвления

Операторы ветвления позволяют изменить порядок выполнения операторов в программе. К операторам ветвления относятся условный оператор if и оператор выбора switch.

### Условный оператор if

Условный оператор *if* используется для разветвления процесса обработки данных на два направления. Он может иметь одну из форм: *сокращенную* или *полную*.

Форма *сокращенного оператора if*:

if (B) S;

где B - логическое или арифметическое выражение, истинность которого проверяется; S - оператор: простой или составной.

При выполнении сокращенной формы оператора if сначала вычисляется выражение B, затем проводится анализ его результата: если В истинно, то выполняется оператор S ; если B ложно, то оператор S пропускается. Таким образом, с помощью сокращенной формы оператора if можно либо выполнить оператор S, либо пропустить его.

Форма *полного оператора if*:

if (B) S1; else S2;

где B - логическое или арифметическое выражение, истинность которого проверяется; S1, S2 - оператор: простой или составной.

При выполнении полной формы оператора if сначала вычисляется выражение B, затем анализируется его результат: если B истинно, то выполняется оператор S1, а оператор S2 пропускается;

если В ложно, то выполняется оператор S2, а S1 - пропускается. Таким образом, с помощью полной формы оператора if можно выбрать одно из двух альтернативных действий процесса обработки данных.

### **Пример 1.**

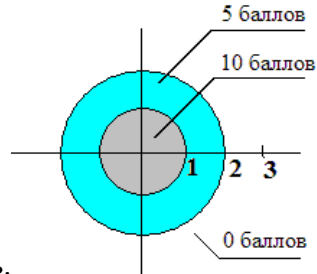
```
if (a > 0) x=y;           // Сокращенная форма с простым оператором
if (++i) {x=y; y=2*z;}    // Сокращенная форма с составным оператором
if (a > 0 || b<0) x=y; else x=z; // Полная форма с простым оператором
if (i+j-1) { x= 0; y= 1;} else {x=1; y:=0;} // Полная форма с составными операторами
```

### **Пример 2.**

```
static void Main()
{
    Console.Write("x= ");
    float x = float.Parse(Console.ReadLine());
    Console.Write("y=");
    float y = float.Parse(Console.ReadLine());
    if (x < y ) Console.WriteLine("min= "+x);
    else Console.WriteLine("min= "+y);
}
```

*Результат работы программы:*

x	y	min
0	0	0
1	-1	-1
-2	2	-2



**Пример 3.** Дана мишень.  
по данной мишени.

Подсчитать количество очков после выстрела

```
static void Main()
{
    int Ball=0;
    Console.Write("x= ");
    float x = float.Parse(Console.ReadLine());
    Console.Write("y= ");
    float y = float.Parse(Console.ReadLine());
    if (x * x + y * y <= 1) Ball = 10; //окружность с радиусом 1
    else if (x * x + y * y <= 4) Ball = 5; //окружность с радиусом 2
    Console.WriteLine("Ball= " + Ball);
}
```

*Результат работы программы:*

x	y	Ball
0	0	10
1	-1	5
-2	2	0