

A photograph of a dirt path winding through a dense forest. The trees are lush green, with some leaves showing a yellowish tint, suggesting early autumn. The path is light brown and curves to the right. The background is filled with thick foliage and tree trunks.

Составные
условия в
разветвляющихся
алгоритмах

© М.Е.Макарова

<http://www.uchinfo.com.ua>

**Для чего
используются
разветвляющиеся
алгоритмы?**

**Для принятия
решений в
зависимости от
каких-либо
условий.**



На прошлом уроке мы узнали:

Формы записи условного оператора:

1. Полная:

IF логическое выражение

Then оператор1

Else оператор2;

2. Неполная:

IF логическое выражение **Then** оператор1;

*Ветвление на три и более
рукава можно организовывать с
помощью вложенных команд IF,
нескольких неполных команд IF,
а также, комбинируя обе
возможности.*

Сравнение строк на компьютере основано на том, что каждый символ имеет свой ASCII-код, и при сравнении строк сравниваются коды составляющих их символов.

В качестве условий ветвления действий в алгоритмах могут указываться любые равенства и неравенства над величинами и алгебраическими выражениями.

Простое условие:

- 1) состоит из двух операндов (которыми могут быть переменные, константы или выражения), соединенных операцией отношения;
- 2) может принимать одно из двух значений — *истина* или *ложь*;
- 3) используется в качестве условия в ветвлениях.

Примеры:

$$X > Y$$

$$S = R + 2$$

В операторах условного перехода можно использовать не только простые, но и так называемые составные (или сложные) условия

Логическое выражение:

- 1) может принимать одно из двух значений — **истина** или **ложь**;
- 2) состоит из одного или нескольких простых условий, соединенных логическими операциями;
- 3) используется в качестве условия в конструкции выбора.

Примеры:

НЕ ($A \geq B$)

(A ≥ B) ИЛИ (C > B)

Условия такого типа называют логическими выражениями

Самое сложное
логическое
высказывание можно
выразить с помощью
трех основных
логических операций:
НЕ, И, ИЛИ.

Унарная операция **НЕ** – логическое отрицание. Смысл этой операции можно выразить словами: "Неверно, что", то есть она меняет значение *истина* на *ложь* и наоборот.

На слайде приведена таблица истинности операции **НЕ**.

Таблица истинности операции **НЕ**

P	НЕ P
истина	ложь
ложь	истина

P — простое условие.

Пример:

НЕ ($A > 0$)

Это логическое выражение означает то же самое, что и $A \leq 0$.

Операция логического умножения **И** применяется тогда, когда переменная должна удовлетворять одновременно двум более простым условиям.

Таблица истинности операции И

P	Q	$P \text{ И } Q$
истина	истина	истина
истина	ложь	ложь
ложь	истина	ложь
ложь	ложь	ложь

P , Q – простые условия

Истина И Ложь = Ложь

"Я обут только тогда, когда у меня ботинок на левой И на правой ноге".
"На левой ноге ботинок есть".
"На правой ноге ботинка нет".
"Следовательно, я не обут".



Операция логического сложения **ИЛИ** применяется тогда, когда переменная должна удовлетворять хотя бы одному из двух и более простых условий

Таблица истинности операции ИЛИ

P	Q	P ИЛИ Q
истина	истина	истина
истина	ложь	истина
ложь	истина	истина
ложь	ложь	ложь

P и Q — простые условия.

Истина ИЛИ Ложь = Истина

"Я напьюсь, если выпью чашку чаю ИЛИ стакан минеральной воды".

"Я выпил чай".

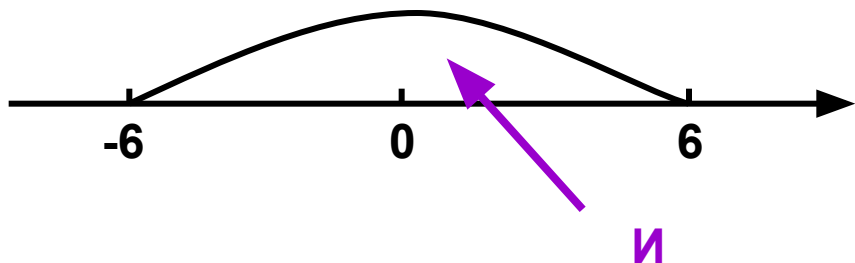
"Я не пил минеральную воду".

"Я напился".



Рассмотрим примеры:

Пример 1

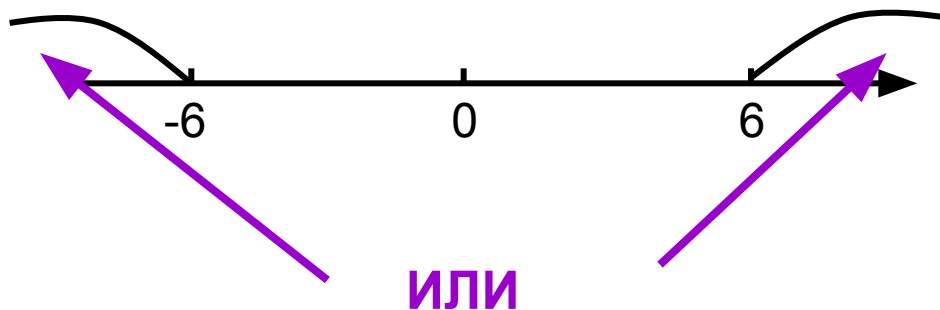


Запишем логическое выражение для случая т. X принадлежит отрезку $[-6, 6]$:
Математическая запись:
 $-6 \leq X \leq 6$.

Логическое выражение:
 $(X \geq -6) \text{ И } (x \leq 6)$

Запись выражения в Паскале:
 $(X \geq -6) \text{ AND } (x \leq 6)$

Пример 2



Запишем логическое выражение для случая т. X не принадлежит отрезку $[-6, 6]$:
Математическая запись:
 $-6 < X > 6$.

Логическое выражение:
 $(X < -6) \text{ ИЛИ } (x > 6)$

Запись выражения в Паскале:
 $(X < -6) \text{ OR } (x > 6)$

Пример 3

Фирма осуществляет набор сотрудников. Условия приема требуют не менее 5 лет рабочего стажа и возраст не более 40 лет.

Написать алгоритм, проверяющий, удовлетворяет ли кандидат этим требованиям.



Решение

Пусть переменная **a** обозначает стаж кандидата, а **b** – его возраст. Тогда должно соблюдаться условие:

(a >= 5) and (b <= 40)

Программа:

Program RP1;

Var a, b: **real**;

Begin

Writeln ('Задайте значения a, b');

Readln (a, b);

If (a >= 5) **and** (b <= 40)

Then Writeln ('Вы приняты')

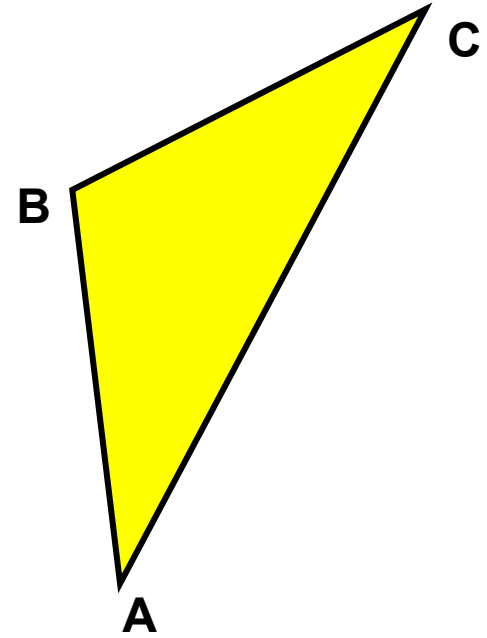
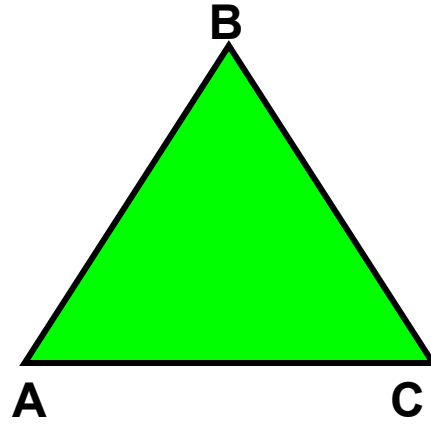
Else Writeln ('Вы не приняты');

Readln;

End.

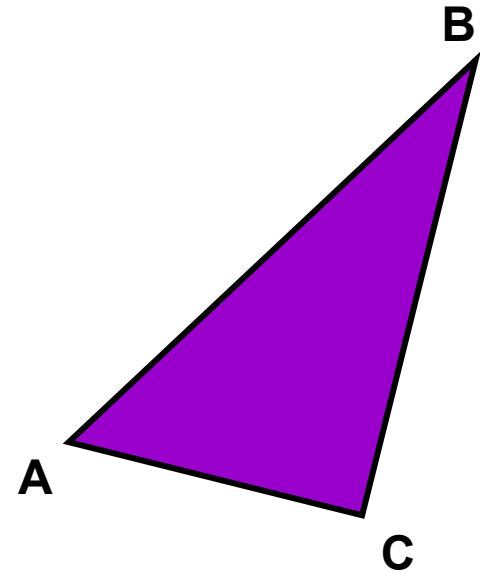
Пример 4

Написать алгоритм для определения вида треугольника (остроугольный, прямоугольный, тупоугольный), если даны величины двух его углов A и B в градусах.



Решение

```
Program Pr4;  
Var a, b, c: real;  
Begin  
  WriteLn ('Задайте величины углов a, b') ;  
  ReadLn (a, b);  
  c:=180-a-b;  
  If (a > 90) or (b > 90) or (c > 90)  
    Then WriteLn ('Треугольник тупоугольный')  
  Else  
    If (a = 90) or ( b = 90) or (c = 90)  
      Then WriteLn ('Треугольник прямоугольный')  
    Else WriteLn ('Треугольник остроугольный');  
End.
```



Условие $X < 0$ или $X > 100$ записывают:

$(X < 0)$ ИЛИ $(X > 100)$

или

НЕ $(0 \leq X$ И $X \leq 100)$

Для изменения порядка операций можно использовать круглые скобки.

В логических выражениях логические операции могут связывать несколько логических переменных, например:

$(X < 0)$ И $(Y < 0)$ И $(Z < 0)$

$(X < 0)$ ИЛИ $(Y < 0)$ И $(Z < 0)$

**Порядок выполнения действий
в логических выражениях:**

- 1. Арифметические операции.**
- 2. Операции отношения.**
- 3. Логические операции:**
 - 1) НЕ;**
 - 2) И;**
 - 3) ИЛИ.**

Логическое выражение:

1. Может принимать одно из двух значений — *истина* или *ложь*.

2. Состоит из логических переменных, соединенных логической операцией (или операции НЕ и логической переменной).

3. Очередность выполнения логических операций осуществляется согласно следующим приоритетам:

1) НЕ; 2) И; 3) ИЛИ.

4. Используется в качестве условия в операторах условного перехода.

Установить, истинны или ложны следующие условия:

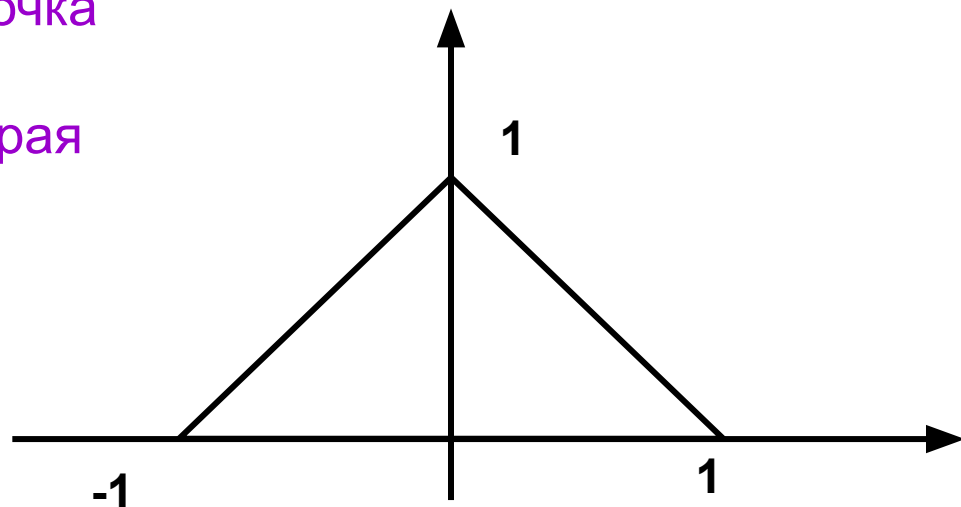
1. NOT ((6<4) AND (4>8)),

2. (A<=A+1) OR (12-3>0),

3. ((2<=2) AND (3 >=3)) OR (15>25).

2. Написать программу, которая определяет, принадлежит ли точка с координатами (X, Y) заштрихованной области, которая описывается системой неравенств:

$$\begin{cases} y \geq 0; \\ |x| + |y| \leq 1 \end{cases}$$



Входные данные: координаты точки (X, Y) **Результат:** слова «Да» или «Нет»

Program Pr7;

Uses Crt;

Var x, y: Real;

Begin

ClrScr;

Writeln('Введите координаты точки x, y');

Readln (x, y);

If (abs(x)+abs(y)<=1) and (y>=1)

Then writeln ('Да')

Else writeln ('Нет');

Readln;

End.

Примеры программ:

Ракета запускается с точки на экваторе Земли со скоростью V (км/с) в направлении движения Земли по орбите вокруг Солнца. Каким будет результат запуска ракеты в зависимости от скорости V ?

$V < 7.8$ Ракета упадет на Землю

$7.8 \leq V < 11.2$ Ракета станет спутником Земли

$11.2 \leq V < 16.4$ Ракета станет спутником Солнца

$V > 16.4$ Ракета покинет пределы Солнечной системы

Program PR5;

Var V : real;

Begin

WriteLn ('Задайте значение скорости');

ReadLn (V);

If V < 7.8 Then WriteLn('Ракета упадет на Землю');

If (V >=7.8) and (V <11.2) then writeLn ('Ракета станет спутником Земли');

If (V >=11.2) and (V <=16.4) then writeLn ('Ракета станет спутником Солнца');

If V >16.4 then writeLn ('Ракета покинет пределы Солнечной системы');

ReadLn;

end.

Практическая работа:



- Составить программу, которая определяет, являются ли разные точки с координатами (X_1, Y_1) и (X_2, Y_2) симметричными относительно осей координат или начала координат.

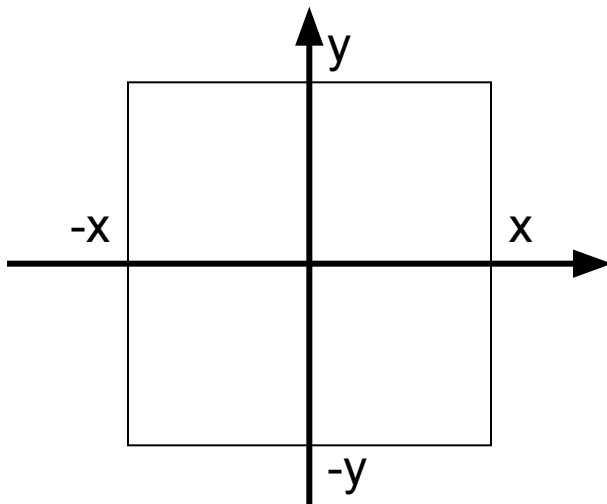
Программу составить двумя способами:

1. С использованием составных условий.
2. Без использования составных условий.

Проверить правильность программ для следующих значений параметров:

А) $X_1 = -1, Y_1 = 6, X_2 = 1, Y_2 = 6$.

Б) $X_1 = 6, Y_1 = 7, X_2 = 1, Y_2 = 6$.



Итоги урока:

На этом занятии мы познакомились с логическими операциями И, ИЛИ, НЕ.

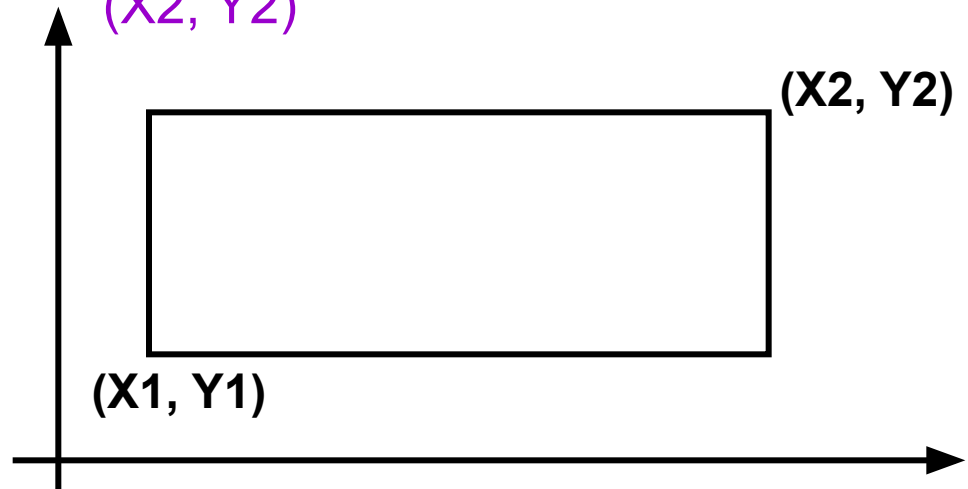
Научились составлять логические выражения и применять их в качестве сложных условий в команде сравнения IF.

Домашнее задание

1. Написать программу, которая спрашивает «Который час?» и, в зависимости от введенного времени, выводит на экран приветствие:

- «Доброе утро»,
- «Добрый день»,
- «Добрый вечер»,
- «Спокойной ночи».

2. Написать программу, которая определяет, принадлежит ли точка с координатами (X, Y) заштрихованной области, ограниченной точками (X_1, Y_1) (X_2, Y_2)



Входные данные: координаты точек (X, Y) , (X_1, Y_1) , (X_2, Y_2)

Результат: слова «Да» или «Нет».