

Теория для Занятия №14-15.

Составной оператор. Условный оператор.

Составной оператор – это последовательность операторов, перед которой стоит слово **BEGIN**, а после – **END**. Слова **BEGIN** и **END** часто именуя операторными скобками.

Пример составного оператора:

BEGIN

a := 10;

b := a*5;

Write(a, b)

END;

Составной оператор может содержать любое количество простых операторов. Он допускает вложенность, т.е. может содержать внутри себя другие составные операторы.

ВАЖНО: Составной оператор применяется в тех случаях, когда синтаксис языка Паскаль допускает использование только одного оператора, в то время как алгоритм требует задания некоторой последовательности действий. В Паскале все управляющие структуры (операторы) не различают простой и составной оператор: там где стоит простой оператор, можно поставить и составной.

Условный оператор IF

Условный оператор **IF** служит для организации процесса вычислений (изменения последовательности выполнения операторов) в зависимости от какого-либо логического условия.

Синтаксис оператора IF:

IF Условие **THEN** Оператор1 **ELSE** Оператор2;

или

IF Условие **THEN** Оператор;

Условие может быть представлено логической константой (**FALSE** – ложь, **TRUE** – истина), переменной логического типа или логическим выражением.

Если условие истинно, то выполняется оператор (простой или составной), следующий за словом **THEN**, в противном случае, когда условие ложно, будет

выполняться оператор, следующий за словом ELSE. Альтернативная ветвь ELSE может отсутствовать, если в ней нет необходимости. В таком “усеченном” варианте, в случае, если условие оказывается ложным, ничего не происходит и выполняется оператор следующий за условным оператором IF.

Пример 1.

Пусть необходимо составить программу, вычисляющую значение функции:

$$y(x) = \begin{cases} x, & \text{при } x > 0 \\ 0, & \text{для всех остальных } x \end{cases}$$

VAR

x, y : real;

BEGIN

Write(‘Введите значение аргумента: ’); ReadLn(x);

IF x>0 **THEN** y := x

ELSE y := 0;

Write(‘Значение функции y = ’, y)

END.

Условные операторы могут быть вложенными друг в друга:

IF Условие **THEN if** ПодУсловие **then ... else ...**

ELSE ...;

Обратите внимание на отсутствие точки с запятой между ключевыми словами.

При вложениях условных операторов самое главное – не запутаться в вариантах сочетаний условий (отчасти этому может помочь ступенчатая форма записи операторов). Всегда действует правило: альтернатива ELSE считается принадлежащей ближайшему условному оператору IF, не имеющему ветви ELSE.

Например:

IF Условие1 **THEN if** Условие2 **then** Оператор_А

ELSE Оператор_Б;

По записи может сложиться впечатление, что Оператор_Б будет выполняться только при “невыполнении” Условия1. Но на самом деле он будет отнесен к Условию2 и выполнится только при выполнении Условия1 и невыполнении Условия2! Если необходимо, чтобы Оператор_Б выполнялся именно при

невыполнении Условия1, то единственный выход таков: заключить вложенный (усеченный) оператор IF в операторные скобки begin...end:

IF Условие1 **THEN begin**

if Условие2 **then** Оператор_А

end

ELSE Оператор_Б;

Практическая часть

1. Составить программу нахождения корней КВУР (квадратного уравнения). Коэффициенты **a**, **b**, **c** вводятся с клавиатуры. **(программу прислать по почте!)**
2. Составить программу, определяющую по введенной дате и месяцу рождения знак зодиака. **(программу прислать по почте!)**
3. Клетки шахматной доски пронумерованы от 1 до 64 по строкам слева направо и снизу вверх. Написать программу, которая по заданному номеру **N** клетки, выводит номера всех клеток, имеющих с ней общую сторону. **(программу прислать по почте!)**
4. Напишите программу, которая определяет вид треугольника (равнобедренный, равносторонний, прямоугольный и т.д) по длинам сторон **a**, **b**, **c**. (Эту задачу называют тестом Г.Майерса на профпригодность. Если вы в своей программе сумеете реализовать порядка 10 различных ситуаций, то вам следует выбрать программирование своей специальностью 😊)