Тема урока: «Программирование разветвляющихся алгоритмов. Условный оператор».

Сегодня мы переходим к изучению синтаксиса условного оператора в языке Pascal.

Теперь повторим материал из раздела «Алгоритмизация» и предыдущих занятий:

• какую алгоритмическую конструкцию называют ветвлением;

• какие формы ветвления вы знаете;

• как записываются логические функции И, ИЛИ, НЕ в языке Pascal.

(Отвечаем на вопросы письменно. Запишите, что написано красным цветом)

Ветвление – алгоритмическая конструкция, в которой в зависимости от результата проверки условия («да» или «нет») предусмотрен выбор одной из двух последовательностей действий (ветвей).

Существуют полная и сокращенная формы ветвления.

AND, OR, NOT соответственно.

А теперь рассмотрим, какой синтаксис имеют операторы ветвления в языке Pascal. Для начала я предлагаю вам заготовить таблицу из двух колонок. Колонки таблицы назовите «Полная форма условного оператора» и «Сокращенная форма условного оператора». Ниже запишем общий вид полной и сокращенной форм условного оператора.

Полная форма имеет следующий синтаксис:

if <условие>

then <оператор 1>

else <оператор 2>;

Сначала записывается служебное слово if, затем – условие, от истинности которого зависит выполнение той или иной ветки условного оператора. После условия записывается служебное слово then, после которого записывается оператор, который выполнится в случае истинности условия. Затем записывается служебное слово else, после которого записывается оператор, который выполнится в том случае, если условие окажется ложным.

Отметьте в тетрадях, что перед служебным словом else знак «;» не ставится.

Краткая форма условного оператора имеет следующий синтаксис:

if <условие>

then <оператор 1>;

Сначала записывается служебное слово if, затем – условие, от истинности которого зависит выполнение той или иной ветки условного оператора. После условия записывается служебное слово then, после которого записывается оператор, который выполнится в случае истинности условия.

Обратите внимание, что если в результате проверки условие окажется ложным, то условный оператор в краткой форме просто будет пропущен, и программа перейдет к выполнению следующих за ним операторов. Для полной формы условного оператора такое утверждение не верно – он выполнится всегда (будет выполнен оператор одной из веток условного оператора).

Также отметьте, что условие в условном операторе – это простое (записанное с помощью операций отношения) или сложное (записанное с помощью логических операций) логическое выражение.

Простое условие a>b

Сложное условие (a>100) and (a mod 2 =0)

Рассмотрим с вами следующий фрагмент программы:

a:=11;

b:=a div 3;

a:=a-b;

if a\*b mod 2 = 0

then writeln(a\*b)

else writeln(b-a);

Какими будут значения переменных a и b после выполнения этой последовательности операторов? Что будет выведено на экран? (запишите в тетрадь)

Рассмотрим с вами следующие задачи (запишите её условие в своих тетрадях): Задача пользователь вводит три различных числа, среди которых необходимо найти минимальное и вывести его на экран.

Записываем решение в своих тетрадях.

program minimum;

var

a, b, c, min: real;

begin

read(a, b, c);

min:=a;

if b < min then min:=b;

if c < min then min:=c;

writeln('Min -> ', min);

end.

Задача 2. Пишем условие и решение

 Написать программу, определяющую, принадлежит ли точка A(x, y) закрашенной области, в том числе и границе области:



program point;

var

x, y: real;

begin

readln(x, y);

if (x >= -1) and (x <= 3) and (y >= -2) and (y <= 4) then

writeln('Точка принадлежит области')

else

writeln('Точка не принадлежит области');

end.

Обратите внимание на сложное условие.

Выполните следующие задания (на оценку)

1. Составьте верную последовательность структуры программы и запишите ее в тетрадь
* Название программы
* Конец программы
* Начало программы
* Тело программы
* Раздел описания переменных
1. Определите тип данных

7,5 мама -1567 2,03 Катерина @

1. Определите значение переменной b после выполнения фрагмента программы.

 a := -5

b := 5 + 7 \* a

b := b / 2 \* a

1. *Рассмотрите внимательно программу*

Program *p\_1;*

Var *x: integer; y:real;*

 Begin

*write(‘Введите значение переменной х’);*

*read(x);*

*y:=x/(x-5);*

*write(‘x=’,x)*

end.

Выполните программу для значения переменной х равной 10,0,5.

Запишите результаты выхода (что получится при каждом х).

Обратите внимание, что при х=5 будет ошибка!!! (на 0 делить нельзя!)

 Измените программу так, чтобы она работала правильно (Если делитель ноль, то выводила сообщение на 0 делить нельзя!)

Запишите измененную программу.

Вышлете мне записи за урок и выполненные задания.

1. Составьте верную последовательность структуры программы

• Название программы

• Раздел описания переменных

• Начало программы

• Тело программы

• Конец программы

2. Определите тип данных

3. Определите значение переменной в после выполнения фрагмента программы.

 a := -5

b := 5 + 7 \* a

b := b / 2 \* a

У. Молодцы! Переходим к проверки домашнего задания. Займите свои места за

компьютером. Запустите программу Паскаль. Прочитайте условие задачи.

(Задача 1. Вычислить значения функции заданной следующим образом: y=х/(x-5))

Program p\_1;

Var x: integer; y:real;

 Begin

write(‘Введите значение переменной х’);

read(x);

1. Составьте верную последовательность структуры программы

• Название программы

• Раздел описания переменных

• Начало программы

• Тело программы

• Конец программы

2. Определите тип данных

3. Определите значение переменной в после выполнения фрагмента программы.

 a := -5

b := 5 + 7 \* a

b := b / 2 \* a

У. Молодцы! Переходим к проверки домашнего задания. Займите свои места за

компьютером. Запустите программу Паскаль. Прочитайте условие задачи.

(Задача 1. Вычислить значения функции заданной следующим образом: y=х/(x-5))

Program p\_1;

Var x: integer; y:real;

 Begin

write(‘Введите значение переменной х’);

read(x);

y:=x/(x-5);

write(‘x=’,x)

end.

У. Какая кнопка предназначена для запуска программа? Запустите программу.

Выполните программу для значения переменной х равной 10,0,5.

Назовите результаты выхода.

2, 0, ошибка вещественного деления на 0.

Почему возникла ошибка? (на 0 делить нельзя!)

 Можем ли мы изменить программу, что бы она была верной для любых чисел.

Почему?