

Язык программирования Delphi

Подготовил:
Студент гр. 301
Лукьянов Илья

Алфавит языка

- 53 буквы латинского алфавита и символ подчеркивания
- Цифры от 0 до 9
- 23 спец.символа (+, -, *, /, .., :=, >, <, ', (,), {, }, [,], #, \$, ^, @, пробел)
- Комбинации спец.символов (:=, <>, ..., <=, >=, (*))

Словарь языка

- Ключевые слова
(Unit, Goto, Begin, Interface)
- Стандартные идентификаторы
(Sin, Pi, Real)
- Пользовательские идентификаторы

Структура программы

Program <Имя программы>
Uses <Список модулей>
Label <Список меток>
Const <Список констант>
Type <Описание типов>
Var <Объявление переменных>;
<Описание процедур>;
<Описание функций>;
Begin
 <Инструкции>;
End.

Подпрограмма

- Логически законченная и специальным образом оформленная часть программы, которая может вызываться для выполнения из других точек программы неограниченное число раз
- Двух видов: процедуры и функции
- Представляет собой блок и должна быть определена в разделе описания процедур и функций

Комментарии

- Пояснительный текст, который можно записывать в любом месте программы, где разрешен пробел
- Ограничен символами { и } или (* и *).
Может занимать несколько строк
- Однострочный комментарий содержит // в начале строки

Примеры комментариев

// Однострочный комментарий

{ Начало многострочного комментария

...

Окончание многострочного комментария}

Основные инструкции языка Delphi

Простые инструкции:

- Присваивания
- Перехода
- Пустая инструкция
- Вызова процедуры

Инструкция присваивания

Имя переменной := Выражение

Примеры:

- $x := x + 1;$
- $n := 17 * n - 1;$
- $s := \text{'Дата'} + \text{DateToStr (Date)};$

Инструкция перехода

Goto <Метка>;

Пример:

Label m1;

...

Goto m1;

...

m1: <Инструкция>;



Структурированные инструкции:

- Составная инструкция
- Условная инструкция
- Инструкция выбора
- Инструкция цикла
- Инструкция доступа

Составная инструкция

Begin <Оператор1>; ...; <ОператорN>; **end**;

Пример:

Begin

 Beep;

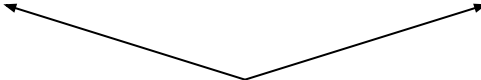
 Edit1.Text:='Ошибка';

 Exit;

End;

Условная инструкция

If <Условие> **then** <оператор1> **else** <оператор2>;



Последовательности команд,
заклученные в скобки begin..end

If <Условие> **then** <оператор>;

Составные условия объединяются логическими
операторами not, or, and

Инструкция выбора

Case <ключ выбора> **of**

<список1> : <оператор1>;

...

<списокN> : <операторN>

Else <операторы>

End;

Пример:

Case DayNumber of

1..5 : strDay := ‘Рабочий день’;

6,7 : strDay := ‘Выходной день’;

Else strDay := ‘’;

End;

Цикл с параметром

For <параметр цикла>:=<начальное значение> **to**
<конечное значение> **do** <оператор>;

- Вместо слова **to** можно использовать **downto** для уменьшения параметра цикла
- Параметр цикла – переменная порядкового типа

Оператор цикла с предусловием

While <условие> **do** <тело цикла>;

- Тело цикла может ни разу не выполниться
- В теле цикла должна быть инструкция, влияющая на значение условия

Оператор цикла с постусловием

Repeat <тело цикла> **until** <условие>;

- Тело цикла всегда выполнится хотя бы один раз
- Условие выхода должно меняться внутри операторов

Для гибкого управления циклами:

- **Break** – немедленный выход из цикла
- **Continue** – досрочное завершение очередного прохода цикла

Инструкция доступа

With <имя объекта> **do** <инструкция>;

Пример:

//Составные имена пишутся полностью

Form1.Canvas.Pen.Color:=ClRed;

Form1.Canvas.Pen.Width:=5;

Form1.Canvas.Rectangle(10,10,100,100);



//Использование инструкции доступа

With Form1.Canvas **do**

Begin

Pen.Color := ClRed;

Pen.Width := 5;

Rectangle (10,10,100,100);

End;



Спасибо за внимание!