



**ОСНОВНЫЕ СТРУКТУРНЫЕ
ЕДИНИЦЫ ЯЗЫКА
ПРОГРАММИРОВАНИЯ PASCAL**

ЭТАПЫ ИЗГОТОВЛЕНИЯ ПРОГРАММНОГО ПРОДУКТА :

Проектирование

Кодирование

Отладка осуществляется с помощью тестов, т.е. производится

исчерпывающей проверки.



В конце 60-х – начале 70-х гг. XX в. появляется и развивается технология структурного программирования (Э. В. Дейкстр, Х. Д. Милс, Д. Е. Кнут).

В основе структурного программирования лежит теорема, которая была доказана в теории программирования: **алгоритм для решения любой логической задачи можно составить только из структур Следование, Ветвление, Цикл**, называемых базовыми алгоритмическими конструкциями.

Сложные алгоритмы состоят из соединенных между собой базовых структур.

Соединение этих структур может выполняться двумя способами: последовательным и вложенным.

□ В 1965 г. в Дартмутском университете был разработан язык Бейсик.

□ Значительным событием в истории языков программирования стало создание в 1971 г. (профессор из Швейцарии Никлаус Вирт) языка Паскаль как учебного языка структурного программирования.

Фирма Borland International, Inc (США) разработала систему программирования ТурбоПаскаль для ПК. ТурбоПаскаль – это не только язык и транслятор с него, но еще и *интегрированная среда программирования*, обеспечивающая пользователю удобство работы на языке Паскаль.

Широко известны такие версии, как TMT Pascal, GNUPascal, FreePascal.



СПОСОБЫ ТРАНСЛЯЦИИ

Реализовать тот или иной язык программирования на ЭВМ – это значит создать транслятор с этого языка для данной ЭВМ.

Существует два принципиально различных метода трансляции – **компиляция и интерпретация**.

Компиляция является аналогом полного предварительного перевода, а интерпретация – аналогом синхронного перевода.

Транслятор, работающий по принципу компиляции, называется **компилятором**, а транслятор, работающий по принципу интерпретации, – **интерпретатором**.

Структура языка программирования высокого уровня



ЛЕКСИКА ЯЗЫКА ПРОГРАММИРОВАНИЯ PASCAL

Языком называют совокупность символов, соглашений и правил, используемых для общения. Основой языка Pascal является алфавит.

Алфавит-совокупность допустимых в языке символов.

Алфавит языка Turbo Pascal (TP) можно условно разбить на следующие группы:

- символы, используемые в идентификаторах;
- разделители;
- специальные символы;
- зарезервированные слова

ОБЩИЙ ВИД ПРОГРАММЫ НА ЯЗЫКЕ ПАСКАЛЬ:

PROGRAM (имя программы);

LABEL (список меток);

CONST(список постоянных значений);

TYPE(описание сложных типов данных);

VAR(описание данных программы);

BEGIN(начало программного блока)

(алгоритм)

END. (конец программы)



Пример. Программа, вычисляющая сумму двух чисел

```
PROGRAM EXAMPLE_1;           {заголовок программы}
VAR X,Y,S:INTEGER;         {раздел описания
                             переменных}
BEGIN                       {начало раздела операторов}
    WRITELN ('ВВЕДИТЕ ДВА ЧИСЛА'); {вывод сообщения на экран}
    READLN(X,Y);           {ввод двух чисел}
    S:= X+Y;               {подсчет суммы двух чисел}
    WRITELN ('S=' ,S);    {вывод результата}
    READLN;                {задержка экрана}
END.                        {окончание раздела
                             операторов}
```



ТИПЫ ДАННЫХ

Типы данных – множество допустимых значений и операций над ними. С типом величина связаны три свойства: форма внутреннего представления, множество принимаемых значений и множество допустимых операций.



ЦЕЛЫЕ ТИПЫ

В TP имеется 5 стандартных типов целых чисел. Различаются они диапазоном, наличием или отсутствием знака, а также размером занимаемой памяти.

Тип (идентификатор)	Диапазон (множество) значений	Формат	Размер в байтах
SHORTINT	-128.. 127	Знаковый	1
INTEGER	-32768.. 32767	Знаковый	2
LONGINT	-2147483648 ..2147483647	Знаковый	4
BYTE	0.. 255	Беззнаковый	1
WORD	0.. 65535	Беззнаковый	2



Над целыми типами определены такие операции:

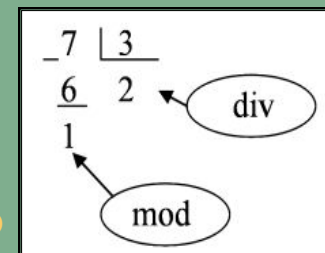
"+" - сложение;

"*" - умножение;

"-" - вычитание;

DIV- целочисленное деление;

MOD- получение остатка от целочисленного деления.



ВЕЩЕСТВЕННЫЕ ТИПЫ

В TP имеется 5 стандартных типов вещественных чисел. Их характеристики приведены в таблице:

Тип (идентификатор)	Диапазон (множество) значений	Число значащих цифр	Размер в байтах
REAL	$2.9 \times 10^{-39} .. 1.7 \times 10^{38}$	11-12	6
SINGLE	$1.5 \times 10^{-45} .. 3.4 \times 10^{38}$	7-8	4
DOUBLE	$5.0 \times 10^{-324} .. 1.7 \times 10^{308}$	15-16	8
EXTENDED	$3.4 \times 10^{-4932} .. 1.1 \times 10^{4932}$	19-20	10
COMP	$-9.2 \times 10^{18} .. 9.2 \times 10^{18}$	19-20	8



Над вещественными типами определены такие операции:

" + " - сложение;

" * " - умножение;

" - " - вычитание;

" / " - деление только для типа REAL.

Логический тип. Логический тип данных часто называют булевым по имени английского математика Д. Буля, создателя математической логики. В языке Турбо Паскаль имеются две логические константы TRUE (ПРАВДА) и FALSE (ЛОЖЬ). Логическая переменная принимает одно из этих значений и имеет тип **BOOLEAN**. Занимает в памяти 1 байт. Для сравнения данных предусмотрены следующие операции отношений: <, <=, =, <>, >, >=. А также существуют специфичные для этого типа логические операции **OR** - или; **AND** - и; **NOT** - не. При проверке некоторых условий результат операции может быть истинным или ложным.

Символьный тип. Символьный тип данных **CHAR** - это тип данных, элементами которого являются символы: буквы, цифры, знаки препинания и специальные символы. Каждому символу алфавита соответствует индивидуальный числовой код от 0 до 255. Занимает в памяти 1 байт. Символьная константа или символьная переменная - любой символ языка, заключённый в апострофы.

Строковый тип. Значением строковой величины типа **STRING** является строка переменной длины (быть может пустая). Строковая константа или строковая переменная представляет собой произвольную последовательность символов, заключённую в апострофы.



ОПИСАНИЕ КОНСТАНТ

Синтаксис описания констант:

CONST

ИмяКонстанты1 = Значение1;

ИмяКонстанты2 = Значение2;

или

ИмяКонстанты = ЗначениеВыражения;

Примеры описания констант:

CONST

Min = 0; {константа - целое число}

Max = 100; {константа - целое число}

e = 2.7; {константа - вещественное число}

SpecChar = '\'; {константа - символ}

HelpStr = 'Нажмите клавишу F1'; {константа - строка}

OK = True; {логическая константа "истина"}



ОПИСАНИЕ ПЕРЕМЕННЫХ

Синтаксис описания переменных:

VAR

ИмяПеременной1 : ИмяТипа1;
ИмяПеременной2 : ИмяТипа2;
Имя Переменной N : Конструкция Типа;

VAR

ИмяПеременной1,ИмяПеременной2,ИмяПеременной3 : ИмяТипа;

Имя типа может быть именем стандартного типа языка или введенного программистом в предшествующем блоке описания типов TYPE.
Однотипные переменные могут перечисляться через запятую перед объявлением их типа.

Примеры описания переменных:

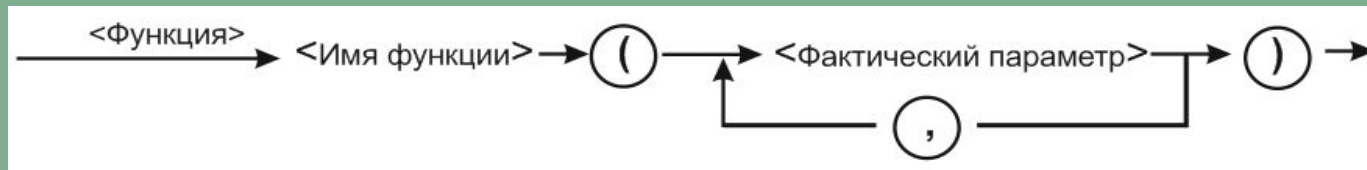
VAR

X : Real;	{вещественная переменная}
i, j, k : Integer;	{три целочисленных переменных}
T : MyType;	{переменная определенного ранее типа MyType}
D : 1..10;	{целочисленная переменная ограниченного типа}



ПРАВИЛА ЗАПИСИ СТАНДАРТНЫХ ФУНКЦИЙ И АРИФМЕТИЧЕСКИХ ВЫРАЖЕНИЙ

В языке Паскаль существует ряд заранее разработанных подпрограмм-функций, которые можно использовать как готовые объекты. Элементарные функции, такие как \sin , \cos , \ln , квадратный корень и др. называются стандартными функциями.



Правила записи стандартных функций

1. Имя функции записывается прописными буквами латинского алфавита. Имя состоит не более чем из шести букв.
2. Аргумент функции записывается в круглых скобках после имени функции.
3. Аргументом функции может быть константа, переменная или арифметическое выражение.
4. В тригонометрических функциях синуса или косинуса аргумент может быть задан только в радианной мере. Если аргумент x дан в градусах, для перевода используется формула $x\pi/180$.



Выражения формируются из операндов, знаков операций и скобок, позволяющих устанавливать требуемый порядок выполнения действий.

Операндами могут служить переменные, константы и имена функций, типы которых должны соответствовать виду операции. Большинство операций в языке Паскаль являются бинарными, т.е. в них участвуют два операнда. Остальные операции называются унарными, поскольку выполняются над единственным операндом. Бинарные операции изображаются в традиционной алгебраической записи, например $A+B$. обозначение унарной операции всегда предшествует операнду, например $-C$ (смена знака величины).

Бинарные арифметические операции стандартного языка Паскаль

Знак	Выражение	Типы операндов	Тип результата	Операция
+	A+B	R, R I, I I, R; R, I	R I R	Сложение
-	A-B	R, R I, I I, R; R, I	R I R	Вычитание
*	A*B	R, R I, I I, R; R, I	R I R	Умножение
/	A/B	R, R I, I I, R; R, I	R R R	Вещественное деление
DIV	A DIV B	I, I	I	Целое деление
MOD	A MOD B	I, I	I	Остаток от целого деления



По функциональному признаку все множество операций в Паскале, можно разделить на шесть групп:

1. арифметические операции
2. логические операции
3. операции сравнения (отношения)
4. строковая операция: «+» - объединение;
5. операции над множествами: «+», «-», «*»;
6. адресная операция @...



ПРАВИЛА ЗАПИСИ АРИФМЕТИЧЕСКИХ ВЫРАЖЕНИЙ

1. Записывать все составные части выражений в одну строку. В выражениях двухэтажные и более, верхние и нижние индексы не допускаются.
2. Использовать скобки только одного типа – круглые. В правильно записанном выражении число открывающихся скобок всегда должно равняться числу закрывающихся скобок.
3. Нельзя записывать подряд два знака арифметических операций.
4. Операции с более высоким приоритетом выполняют раньше операций с меньшим приоритетом.

Несколько записанных подряд операций с одинаковым приоритетом выполняются последовательно слева направо.

В Паскале нет операции или стандартной функции возведения числа в произвольную степень ($a^x := \text{Exp}(x * \text{Ln}(a))$)

Не следует записывать выражения, не имеющие математического смысла, например: деление на нуль, логарифм отрицательного числа и т.п.



ПРАВИЛА ЗАПИСИ ЛОГИЧЕСКИХ ВЫРАЖЕНИЙ

1. Логические выражения строятся из логических данных, логических операций и операций отношений.
2. В операциях отношения могут участвовать арифметические и логические выражения, а также символьные данные.
3. Результатом логического выражения является значение TRUE или FALSE.
4. В логическом выражении допускается использование только круглых скобок.
5. При наличие скобок сначала выполняются действия в скобках (в первую очередь в самых внутренних), а затем вне скобок.
6. В круглые скобки обязательно заключаются части выражения, стоящие слева и справа от логических операций **AND** и **OR**.

