

*Данная презентация школьнику, начинающему писать на Pascal ABC позволит узнать минимум необходимой для этого информации.*

*Автор презентации благодарит всех, из чьих электронных учебников взята подборка примеров.*


PASCAL

Алгоритмизация и программирование

# Вводная часть.

---

- Процесс решения задачи на ПК – это совместная деятельность человека и машины. Его условно можно разделить на несколько этапов. Человеку отводятся шаги, связанные с творческой деятельностью (постановкой, алгоритмизацией, программированием задачи и анализом результатов), на долю компьютера – этапы обработки информации в соответствии с разработанным алгоритмом. При этом говорят, что компьютер выполняет программу (последовательность команд), реализующую данный алгоритм.
- Команды, предназначенные для ПК, должны быть представлены в виде, понятном машине. Для этого применяются языки программирования.
- Существуют такие языки, которые отражают структуру данного класса машин, и поэтому их называют машинно-ориентированными языками или языками низкого уровня. Чтобы реализовать заданный алгоритм с помощью языка низкого уровня необходимо записать программу, представляющую собой определенную последовательность нулей и единиц. Особенностью этих языков является жесткая ориентация на определенный тип аппаратуры. В стремлении приспособить эти языки к человеку, был разработан язык символического кодирования – язык ассемблера.

- 
- Для облегчения труда программистов были созданы языки программирования, которые строились на основе определенного алфавита и строгих правил построения предложений. Отличительной особенностью этих языков является их ориентация не на систему команд той или иной ЭВМ, а на систему операторов, характерных для записи определенного класса алгоритмов. Такие языки принято называть языками программирования высокого уровня. К ним относятся: Бейсик, Фортран, Паскаль, Алгол, Си и многие другие.
  - Так как текст, записанной программы на языках высокого уровня не понятен ПК, требуется перевести его на машинный язык. Такой перевод на язык машинных кодов называется трансляцией, а выполняется он специальными программами – трансляторами.
  - Существует два основных вида трансляторов: интерпретаторы (трансляторы, производящие пооператорную обработку и выполнение программы) и компиляторы (трансляторы, преобразующие всю программу в модуль на машинном языке и только после записи в памяти ПК, выполняющие исходную программу).
- 
- 

- Язык программирования Паскаль получил свое название в честь выдающегося французского математика, физика, философа Блеза Паскаля (1623-1662), который изобрел механическую счетную машину для выполнения арифметических операций (паскалево колесо, как говорили современники). В 1965 году Международная федерация по обработке информации (IFIP) предложила нескольким специалистам в области информатики принять участие в создании нового языка программирования. Среди них был швейцарский ученый Н.Вирт. Вскоре он и его сотрудники из Федерального технического университета подготовили первую версию Паскаля, затем – первый вариант компилятора, а в 1971 году вышло описание языка. В середине 80-х гг. фирмой Borland International, Inc (США) для повышения качества и скорости разработки программ была создана система программирования Турбо Паскаль, расширившая область применения языка и развившая сам язык программирования.
- Разработка программ на Паскале включает в себя следующие действия: ввод и редактирование текста на языке программирования, трансляция и отладка программы. Для каждого этапа требуются специальные средства: редактор текстов, компилятор, включающий в себя компоновщик – для построения исполняемого компьютером программного модуля и отладчик – для отладки программы. Систему программирования Турбо Паскаль называют еще интегрированной средой программирования, так как она объединяет в себе все ранее разрозненные средства, используемые при разработке программ.

# Алфавит и словарь языка Паскаль

---

- Алфавит Паскаля составляют:
- прописные и строчные буквы латинского алфавита: A, B, C...Y, Z, a, b, c,...y, z ;
- десятичные цифры: 0, 1, 2,...9;
- специальные символы: + - \* / > < = ; # ' , . : { } [ ] ( )
- комбинации специальных символов , которые нельзя разделять пробелами, если они используются как знаки операций: «:=», «..», «<>», «<=», «>=», «{ }».
- Неделимые последовательности знаков алфавита образуют слова, отделенные друг от друга разделителями. Ими могут быть пробел, комментарий или символ конца строки. Словарь Паскаля можно разделить на три группы слов: зарезервированные слова, стандартные идентификаторы и идентификаторы пользователя.



- 
- Резервированные слова (см таб.1) имеют фиксированное написание и навсегда определенный смысл. Они не могут изменяться программистом и их нельзя использовать в качестве имен для обозначения величин.



# Зарезервированные слова

Absolute	Абсолютный	Library	Библиотека
And	Логическое И	Mod	Остаток от деления
Array	Массив	Not	Логическое НЕ
Begin	Начало блока	Or	Логическое ИЛИ
Case	Вариант	Of	Из
Const	Константа	Object	Объект
Div	Деление нацело	Procedure	Процедура
Go to	Переход на	Program	Программа
Do	Выполнять	Repeat	Повторять
Downto	Уменьшить до	String	Строка
Else	Иначе	Then	То



# Зарезервированные слова (продолжение).

---

End	Конец блока	To	Увеличивая
File	Файл	Type	Тип
For	Для	Until	До
Function	Функция	Uses	Использовать
If	Если	Var	Переменная
Interrupt	Прерывание	While	Пока
Interface	Интерфейс	With	С
Label	Метка	Xor	Исключающее ИЛИ





# Основные понятия системы программирования Паскаль

## Целочисленные типы данных

---

Тип	Диапазон	Требуемая память (байт)
<b>Byte</b>	0...255	1
Shortint	-128 ...127	1
<b>Integer</b>	-32768 ... 32767	2
Word	0 ... 65535	2
Longint	-2147483648 ...2147483647	4



# Основные понятия системы программирования Паскаль

## Вещественные типы данных.

---

Тип	Диапазон	Мантисса	Требуемая память (байт)
<b>Real</b>	$2.9 \cdot 10^{-39} \dots 1.7 \cdot 10^{38}$	11 – 12	6
Single	$1.5 \cdot 10^{-45} \dots 3.4 \cdot 10^{38}$	7 – 8	4
Double	$5.0 \cdot 10^{-324} \dots 1.7 \cdot 10^{308}$	15 – 16	8
Extended	$1.9 \cdot 10^{-4951} \dots 1.1 \cdot 10^{4932}$	19 – 20	10
Comp	$-2E+63+1 \dots 2E+63-1$	10 – 20	8



## Символьный тип данных.

---

- Литерный (символьный) тип `char` определяется множеством значений кодовой таблицы ПК. Каждому символу приписывается целое число в диапазоне от 0 до 255. Для кодировки используется код ASCII. Например код символа 'А' при русской раскладке клавиатуры будет равен 192 .
- Для размещения в памяти переменной литерного типа нужен 1 байт.



## Логический тип данных.

---

- Логический (булевский) тип `boolean` определяется двумя значениями: `true` (истина) и `false` (ложь). Он применяется в логических выражениях и выражениях отношения. Для размещения в памяти - 1 байт.



## Арифметические операции.

<b>Операция</b>	<b>Действие</b>	<b>Тип операндов</b>	<b>Тип результата</b>
+	Сложение	Целый, вещественный	Целый, вещественный
-	Вычитание	Целый, вещественный	Целый, вещественный
*	Умножение	Целый, вещественный	Целый, вещественный
/	Деление	Целый, вещественный	Вещественный
DIV	Целочисленно е деление	Целый	Целый
MOD	Остаток от деления	Целый	Целый



## Операции отношения.

---

- Операции отношения выполняют сравнение двух операндов и определяют, истинно значение или ложно. Сравнимые величины могут принадлежать к любому типу данных, и результат всегда имеет логический тип, принимая одно значение из двух: истина или ложь.

□

Операция	Название	Выражение
=	Равно	$A=B$
<>	Неравно	$A<>B$
>	Больше	$A>B$
<	Меньше	$A<B$
>=	Больше или равно	$A>=B$
<=	Меньше или равно	$A<=B$



# Стандартные математические функции.

Обращение	Тип аргумента	Тип результата	Функция
<b>Abs(x)</b>	Целый, вещественный	Целый, вещественный	Модуль аргумента
<b>Arctan(x)</b>	Целый, вещественный	Вещественный	Арктангенс
<b>Cos(x)</b>	Целый, вещественный	Вещественный	Косинус
<b>Exp(x)</b>	Целый, вещественный	Вещественный	$e^x$ - экспонента
<b>Frac(x)</b>	Целый, вещественный	Вещественный	Дробная часть x
<b>Int(x)</b>	Целый, вещественный	Вещественный	Целая часть x
<b>Ln(x)</b>	Целый, вещественный	Вещественный	Натуральный логарифм
<b>Random</b>		Вещественный	Псевдослучайное число [0,1]
<b>Random(x)</b>	Целый	Целый	Псевдослучайное число [0,x]
<b>Round(x)</b>	Вещественный	Целый	Округление до ближайшего целого
<b>Sin(x)</b>	Целый, вещественный	Вещественный	Синус
<b>Sqr(x)</b>	Целый, вещественный	Вещественный	Квадрат x
<b>Sqrt(x)</b>	Целый, вещественный	Вещественный	Корень квадратный из x
<b>Trunc(x)</b>	Вещественный	Целый	Ближайшее целое, не превышающее x по модулю



## Логические операции.

- Логические выражения в результате вычисления принимают логические значения True и False. Операндами это выражения могут быть логические константы, переменные, отношения. Идентификатор логического типа в Паскале: `boolean`.
- В Паскале имеется 4 логические операции: отрицание -NOT, логическое умножение -AND, логическое сложение - OR, исключающее «или» -XOR . Используются обозначения: T – true, F – false.

<b>A</b>	<b>B</b>	<b>Not A</b>	<b>A and B</b>	<b>A or B</b>	<b>A xor B</b>
T	T	F	T	T	F
T	F	F	F	T	T
F	F	T	F	F	F
F	T	T	F	T	T

- Приоритеты операций: `not`, `and`, `or`, `xor`. Операции отношения (`=`, `<>` ...) имеют более высокий приоритет, чем логические операции, поэтому их следует заключать в скобки при использовании по отношению к ним логических операций.





## Ввод данных с клавиатуры.

---

- Процедура чтения `Read` обеспечивает ввод данных для последующей их обработки программой. Общий вид: `Read (<список переменных>);`
- В списке перечисляются имена переменных. Значения этих переменных набираются через пробел на клавиатуре и высвечиваются на экране после запуска программы. После набора данных для одной процедуры `Read` нажимается клавиша ввода `Enter`. Значения переменных должны вводиться в строгом соответствии с синтаксисом языка Паскаль. Если соответствие нарушено, то возникают ошибки.
- Процедура чтения `Readln` аналогична процедуре `Read`, единственное отличие в том, что после считывания последнего в списке значения курсор переходит на начало новой строки.

## Пример ввода:

---

- Program primer;
- Var i, k:integer; c,d, s: real;
- begin
- readln (c,d);
- read(i,k);
- ...
- end.
- В данном случае необходимо ввести сначала два действительных числа через пробел. Переменной c присваивается значение, равное первому введенному числу, а переменной d – значение, равное второму введенному числу. После ввода этих значений курсор переходит на начало новой строки (за это отвечает In следующий за словом Read). Далее требуется ввести еще два целых числа, которые будут присвоены значениям переменных i и k соответственно.



## Вывод данных.

---

- Процедура вывода `Write` производит вывод данных.
  - Общий вид: `Write(<список вывода>);`
  - В списке вывода могут быть представлены выражения допустимых типов данных (`integer`, `real`, `char` и т.д.) и произвольный текст, заключенный в апострофы.
  - Например, `Write('Привет');``Write(34.7);``Write(45+55);` `Write(b, d);`
  - Процедура `WriteLn` аналогична процедуре `Write`. Отличие в том, что после вывода последнего в списке выражения курсор переходит на начало новой строки.
  - В процедурах вывода `Write` и `WriteLn` имеется возможность записи выражения, определяющего ширину поля вывода.
- 



## Решение задач.

---

- **Пример 1.** Составить программу, которая запрашивает возраст ребенка и затем выдает решение о приеме ребенка в школу (возраст  $\geq 7$  лет).
- Program sh;
- Var v: integer;
- Begin
- Write('Введите возраст ребенка');
- Readln(v);
- If  $v \geq 7$  then writeln('Принимаем в школу')
- else writeln ('Не принимаем в школу');
- end.



## Программа с использованием логических функций.

---

- **Пример 2.** Модифицировать данную программу, чтобы ограничить верхнюю границу приема в школу 16 годами.
- Решение:
- Program sh;
- Var v: integer;
- Begin
- Write('Введите возраст ребенка');
- Readln(v);
- If (v >= 7) and (v <= 16) then writeln('Принимаем в школу')
- else writeln ('Не принимаем в школу');
- end.



# Вычисляем периметр прямоугольника.

---

- Program perimetr;
- Var a, P:real;
- Begin
- Writeln('Введите a');
- Readln(a);
- P:=4\*a;
- writeln('Ваш периметр=', P);
- end.



# Вычисляем площадь круга.

---

- Program ploschad;
- Var r,S:real;
- Const pi=3.14;
- Begin
- Writeln('Введите r');
- Readln(r);
- S:=pi\*Sqr(r);
- writeln('Ваша площадь = ',S,' условных единиц');
- end.



# Литература

---

- ▣ *Трофимова И.А., Яровая О.В.* Информатика в схемах и таблицах. Москва: Издательство «Эксмо», 2010.
- ▣ *Копыл В.И.* Информатика. Весь школьный курс в таблицах. Минск: «Букмастер», 2011.
- ▣ *Пупышев В.В.* 128 задач по началам программирования. Москва: «Бином. Лаборатория знаний», 2009.
- ▣ Под ред. Н.В.Макаровой. Информатика и ИКТ. Учебник. 8-9, 10, 11 кл. СПб.: «Питер», 2009.
- ▣ Турбо Паскаль 7.0. – Киев: Торгово-издательское бюро ВНУ, 1996.
- ▣ *Щауцукова Л.З.* Информатика: Учеб. Пособие для 10 – 11 кл. общеобразовательных учреждений /. – М.: Просвещение, 2000.

