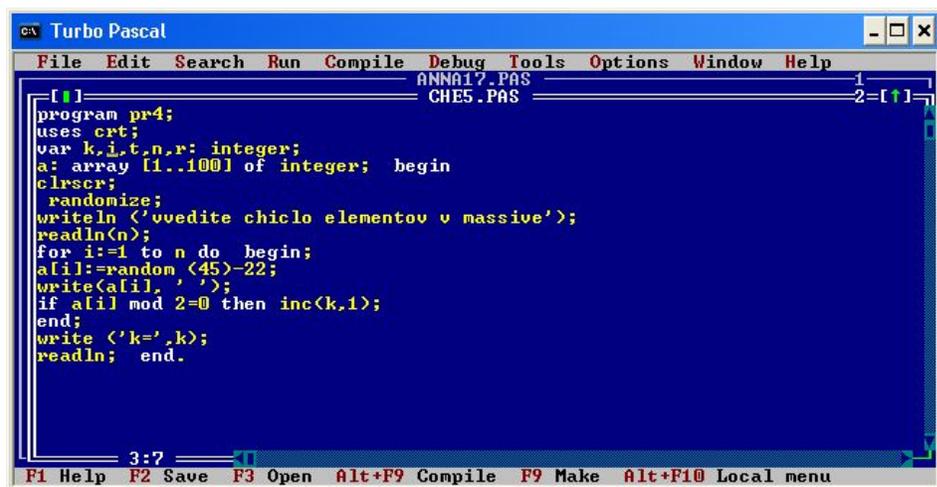


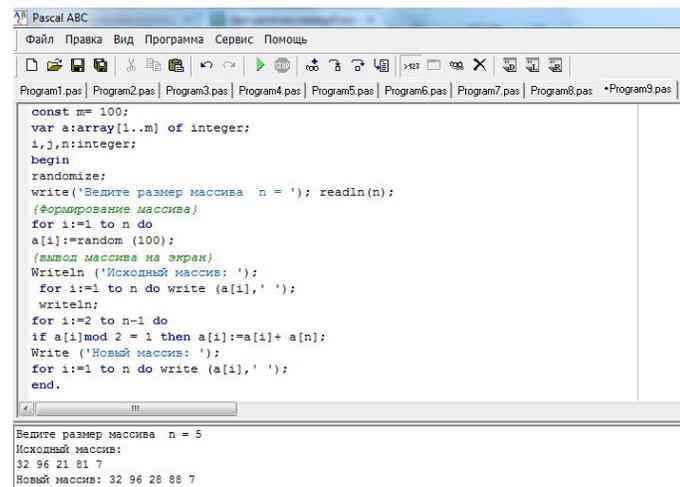
Pascal

Язык программирования Pascal (Паскаль). Общие сведения



```
File Edit Search Run Compile Debug Tools Options Window Help
ANNA17.PAS 1
CHE5.PAS 2
[ ]
program pr4;
uses crt;
var k,i,t,n,r: integer;
a: array [1..100] of integer; begin
clrscr;
randomize;
writeln ('vvedite chiclo elementov v massive');
readln(n);
for i:=1 to n do begin;
alil:=random (45)-22;
write(alil, ' ');
if alil mod 2=0 then inc(k,1);
end;
write ('k=',k);
readln; end.
```

F1 Help F2 Save F3 Open Alt+F9 Compile F9 Make Alt+F10 Local menu



```
Pascal ABC
Файл Правка Вид Программа Сервис Помощь
Program1.pas | Program2.pas | Program3.pas | Program4.pas | Program5.pas | Program6.pas | Program7.pas | Program8.pas | *Program9.pas
const m= 100;
var a:array[1..m] of integer;
i,j,n:integer;
begin
randomize;
write ('Ведите размер массива n = '); readln(n);
(#формирование массива)
for i:=1 to n do
a[i]:=random (100);
(вывод массива на экран)
Writeln ('Исходный массив: ');
for i:=1 to n do write (a[i], ' ');
writeln;
for i:=2 to n-1 do
if a[i] mod 2 = 1 then a[i]:=a[i]+ a[n];
Write ('Новый массив: ');
for i:=1 to n do write (a[i], ' ');
end.
```

Ведите размер массива n = 5
Исходный массив:
32 96 21 81 7
Новый массив: 32 96 28 88 7

ОСНОВНЫЕ ПОНЯТИЯ

Язык программирования – это формальный язык, предназначенные для записи алгоритмов, исполнителем которых будет компьютер.

Алгоритм – набор инструкций, описывающих порядок действий исполнителя для достижения некоторого результата.

Исполнитель алгоритма – это некоторая абстрактная или реальная (техническая, биологическая или биотехническая) система, способная выполнить действия, предписываемые алгоритмом. В информатике универсальным исполнителем алгоритмов является *компьютер*.

Записи алгоритмов на языках программирования называются **программами**.

ИСТОРИЯ

Язык Паскаль был создан швейцарским учёным *Никлаусом Виртом* в 1968—1969 годах. Назван в честь французского математика, физика, литератора и философа *Блеза Паскаля*, который создал первую в мире механическую машину, складывающую два числа.

Цель создания языка Pascal – построение небольшого и эффективного языка, способствующего хорошему стилю программирования, использующему структурное программирование и структурированные данные.



Никлаус Вирт (1934 г.р.) - специалист в области информатики, один из известнейших теоретиков в области разработки языков программирования, профессор информатики (компьютерных наук). Разработчик многих языков программирования.

РЕАЛИЗАЦИИ ЯЗЫКА PASCAL

В 1983 году появилась *первая версия* интегрированной среды разработки *Turbo Pascal* фирмы *Borland*, основывавшаяся на одноимённой реализации Паскаля.

В 1989 году *объектное расширение* языка было добавлено в *Turbo Pascal версии 5.5*.

Последняя версия (7.0) была переименована в *Borland Pascal*.

Дальнейшее развитие реализации Паскаля от *Borland* породило вариант *Object Pascal* от *Borland*, впоследствии, в ходе развития среды программирования *Delphi*.

Важным шагом в развитии языка является появление свободных реализаций языка Паскаль *Free Pascal* и *GNU Pascal*, которые вобрали в себя черты множества других диалектов языка и обеспечили широкую переносимость написанных на нём программ (GNU Pascal поддерживает более 20 различных платформ, под более чем 10 различными операционными системами, Free Pascal обеспечивает специальные режимы совместимости с различными распространёнными диалектами языка: Turbo Pascal, Delphi и др.)

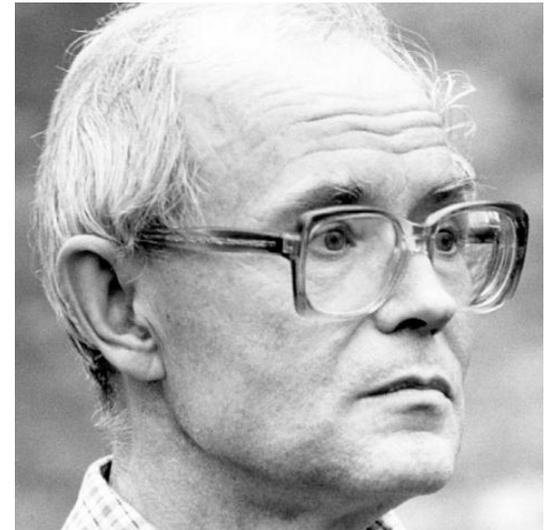
PascalABC.NET — язык программирования Паскаль, включающий большинство возможностей языка *Object Pascal*, а также ряд собственных расширений (разработан в Южном федеральном университете). Он основан на платформе *Microsoft.NET* и содержит практически все современные языковые средства: классы, перегрузку операций, интерфейсы, обработку исключений, обобщенные классы и подпрограммы, сборку мусора, лямбда-выражения.

ОСОБЕННОСТИ ЯЗЫКА PASCAL

Особенностями языка являются *строгая типизация* и *наличие средств структурного (процедурного) программирования*. По мнению Вирта, язык должен способствовать дисциплинированному программированию, в Паскале *сведены к минимуму возможные синтаксические неоднозначности*, а сам синтаксис автор постарался сделать интуитивно понятным даже при первом знакомстве с языком.

К 1980-м годам Паскаль стал основой для многочисленных учебных программ (на его основе были созданы специализированные обучающие языки программирования). В начале 1980-х годов в СССР для обучения школьников основам информатики и вычислительной техники *Андрей Ершов* разработал алголо-паскалеподобный «учебный алгоритмический язык».

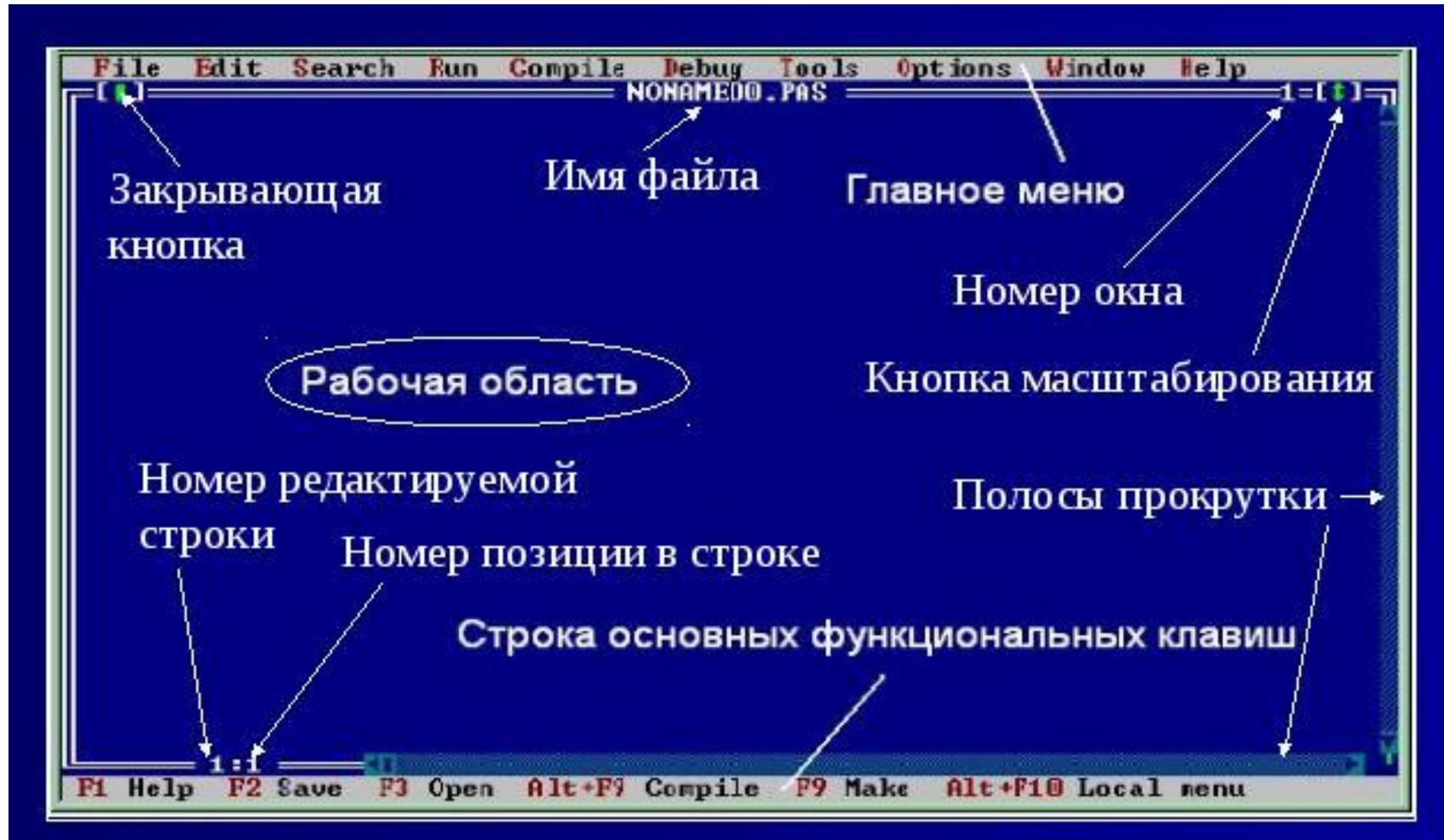
Наиболее известная реализация Паскаля – *Turbo Pascal* фирмы *Borland*, выросшая затем в объектный Паскаль для DOS и Windows и, далее, в *Delphi*, в которой были внедрены значительные расширения языка.



Андрей Петрович Ершов
(19.04.1931 - 8.12.1988)

советский учёный, один из пионеров теоретического и системного программирования, создатель Сибирской школы информатики, академик АН СССР

СРЕДА ПРОГРАММИРОВАНИЯ TURBO PASCAL 7.0



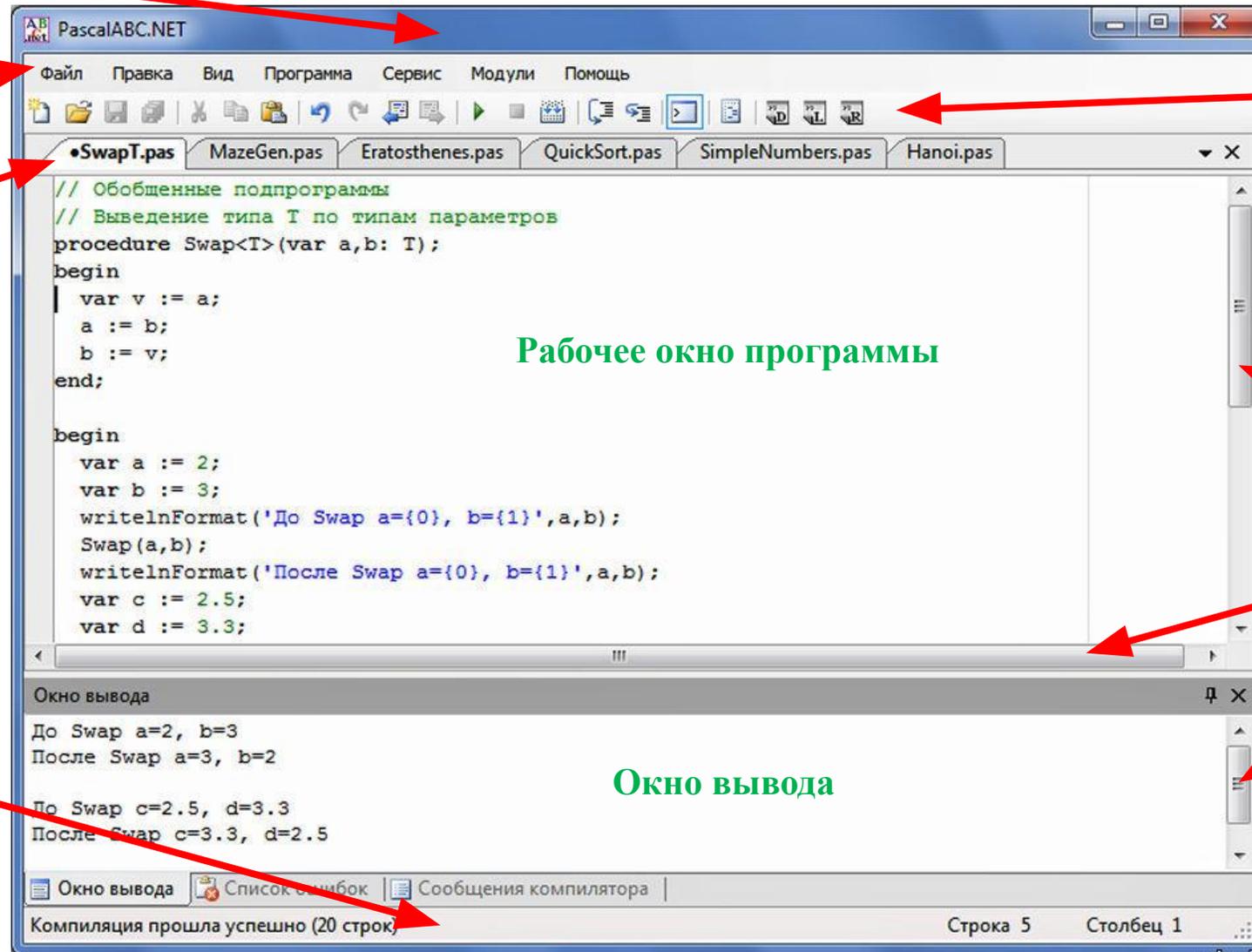
СРЕДА ПРОГРАММИРОВАНИЯ PascalABC.NET

Строка заголовка окна

Строка меню

Вкладки

Строка состояния



Рабочее окно программы

Окно вывода

Панель инструментов

Полосы прокрутки

Окно вывода | Список ошибок | Сообщения компилятора

Строка 5 Столбец 1

ОСНОВНЫЕ СОСТАВЛЯЮЩИЕ ЯЗЫКА PASCAL

Основными элементами любого языка программирования являются:

Алфавит – совокупность символов, отображаемых на устройствах печати и экранах и/или вводимых с клавиатуры. Иногда в это множество включаются неотображаемые символы с указанием правил их записи (комбинирование в лексемы). **Лексика** – совокупность правил образования цепочек символов (*лексем*), образующих идентификаторы (переменные и метки), операторы, операции и другие лексические компоненты языка.

Синтаксис – совокупность правил образования языковых конструкций, или предложений языка программирования – блоков, процедур, составных операторов, условных операторов, операторов цикла и пр.

Семантика – смысловое содержание конструкций, предложений языка, семантический анализ – это проверка смысловой правильности конструкции.

Задача синтаксиса и семантики состоит в том чтобы, правильно написать раздел программы и правильно составить ее структуру, в противном случае наша программа будет работать неправильно или вообще не работать.

АЛФАВИТ ЯЗЫКА PASCAL

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.*

Зарезервированные слова имеют фиксированное написание и навсегда определенный смысл. Они не могут изменяться программистом, и их нельзя использовать в качестве имен для обозначения величин.

Идентификатор – имя (identification – установление соответствия объекта некоторому набору символов). Для обозначения определенных разработчиками языка функций, констант и т. д. служат стандартные идентификаторы, например, **Sqr** (вызывает функцию, которая возводит в квадрат данное число), **Sqrt** (корень квадратный из заданного числа) и т. д.

Идентификаторы пользователя – это те имена, которые дает сам программист.

При записи программ нужно соблюдать **общие правила написания идентификаторов**:

- идентификатор начинается только с буквы (исключение – специальные идентификаторы меток)
- идентификатор может состоять из букв, цифр и знака подчеркивания
- максимальная длина – 127 символов
- при написании идентификаторов можно использовать прописные и строчные буквы
- между двумя идентификаторами должен стоять хотя бы один пробел

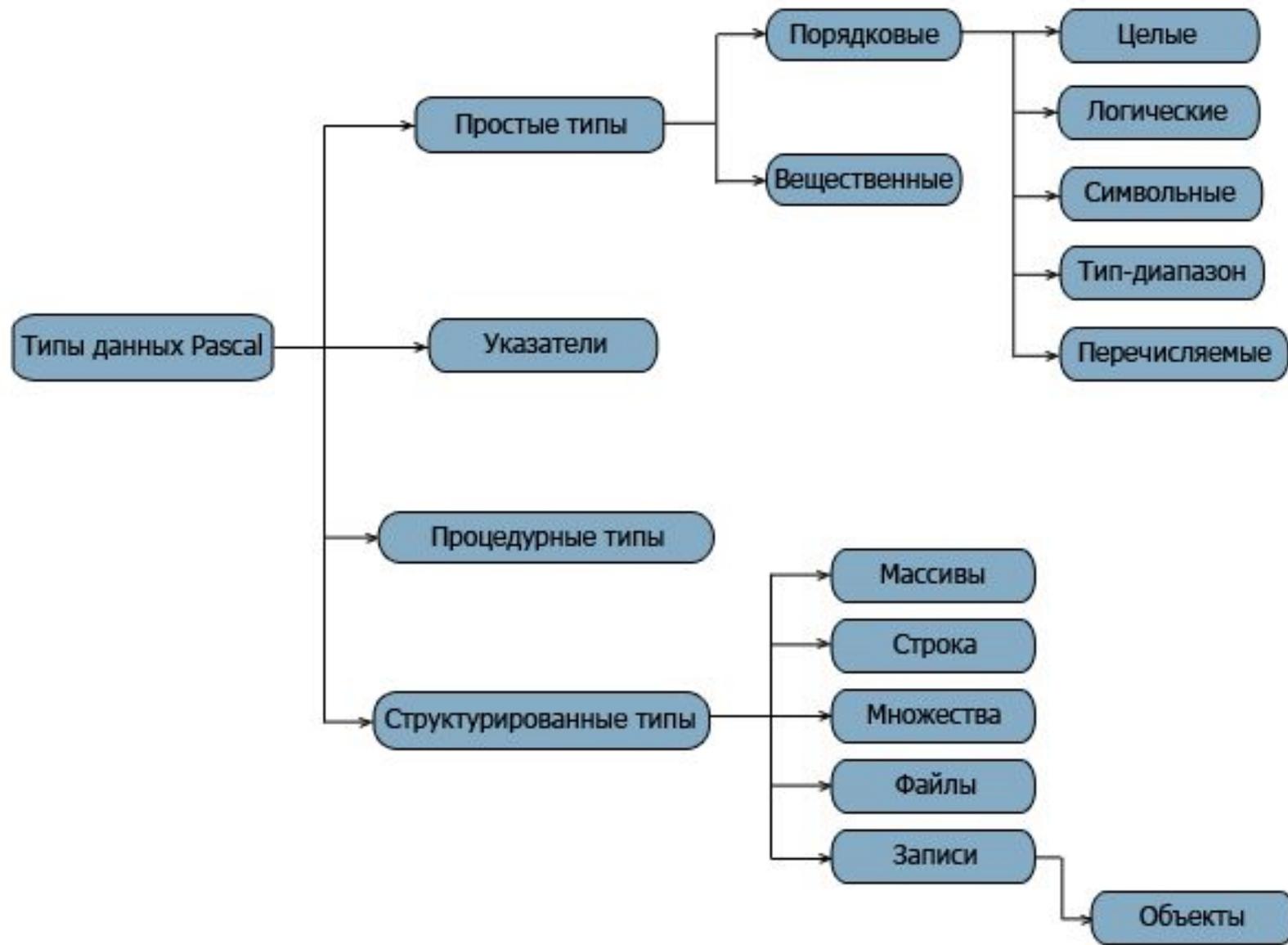
ЗАРЕЗЕРВИРОВАННЫЕ СЛОВА

<i>Absolute</i>	Абсолютный	<i>Library</i>	Библиотека
<i>And</i>	Логическое И	<i>Mod</i>	Остаток от деления
<i>Array</i>	Массив	<i>Not</i>	Логическое НЕ
<i>Begin</i>	Начало блока	<i>Or</i>	Логическое ИЛИ
<i>Case</i>	Вариант	<i>Of</i>	Из
<i>Const</i>	Константа	<i>Object</i>	Объект
<i>Div</i>	Деление нацело	<i>Procedure</i>	Процедура
<i>Goto</i>	Переход на	<i>Program</i>	Программа
<i>Do</i>	Выполнять	<i>Repeat</i>	Повторять
<i>Downto</i>	Уменьшить до	<i>String</i>	Строка
<i>Else</i>	Иначе	<i>Then</i>	То
<i>End</i>	Конец блока	<i>To</i>	Увеличивая до
<i>File</i>	Файл	<i>Type</i>	Тип
<i>For</i>	Для	<i>Until</i>	До тех пор, пока не выполнится
<i>Function</i>	Функция	<i>Uses</i>	Использовать
<i>If</i>	Если	<i>Var</i>	Переменная
<i>Interrupt</i>	Прерывание	<i>While</i>	Пока
<i>Interface</i>	Интерфейс	<i>With</i>	С
<i>Label</i>	Метка	<i>Xor</i>	Исключающее ИЛИ

ТИПЫ ДАННЫХ

Исходные данные, которые вводятся с клавиатуры или считываются из файла, хранятся в *переменных*, а они, в свою очередь, хранятся в оперативной памяти. В языке *Паскаль* переменные характеризуются своим *типом*.

Тип – это свойство переменной, по которому переменная может принимать множество значений, допустимых этим типом, и участвовать во множестве операций, допустимых над данным типом. Тип данных определяет, какие именно данные можно хранить, и сколько они займут оперативной памяти.



ПРОСТЫЕ ТИПЫ ДАННЫХ (ПОРЯДКОВЫЕ)

Название	Диапазон значений	Объём памяти	Допустимые операции
Byte (целый тип)	0 ... 255	1 байт без знака	+, -, /, *, Div, Mod, >=, <=, =, <, >
ShortInt (целый тип)	-128 ... 127	1 байт со знаком	+, -, /, *, Div, Mod, >=, <=, =, <, >
Word (целый тип)	0 ... 65535	2 байта без знака	+, -, /, *, Div, Mod, >=, <=, =, <, >
Integer (целый тип)	-32786...32787	2 байта со знаком	+, -, /, *, Div, Mod, >=, <=, =, <, >
LongInt (целый тип)	-214748364 ... 2147483647	4 байта со знаком	+, -, /, *, Div, Mod, >=, <=, =, <, >
Char (символьный тип)	Произвольный символ алфавита (все символы кода ASCII)	1 байт	+, >=, <=, =, <, >
String (строковый тип)	последовательность символов, заключенных в апострофы (не более 255 символов)	1 байт на символ	+, >=, <=, =, <, >
Boolean (логический тип)	true, false	1 байт	Not, And, Or, Xor, >=, <=, =, <, >

ПРОСТЫЕ ТИПЫ ДАННЫХ (ВЕЩЕСТВЕННЫЕ)

Тип	Диапазон	Память, байт	Количество значащих цифр	Допустимые операции
<i>Single</i>	1.5e-45 ... 3.4e38	4	7-8	+, -, /, *, >=, <=, =, <>, <, >
<i>Real</i>	2.9e-39 ... 1.7e38	6	11-12	+, -, /, *, >=, <=, =, <>, <, >
<i>Double</i>	5.0e-324 ... 1.7e308	8	15-16	+, -, /, *, >=, <=, =, <>, <, >
<i>Extended</i>	3.4e-4932 ... 1.1e4932	10	19-20	+, -, /, *, >=, <=, =, <>, <, >
<i>Comp</i>	-9.2e63 ... (9.2e63)-1	8	19-20	+, -, /, *, >=, <=, =, <>, <, >

СТРУКТУРА ПРОГРАММЫ



ОБЩИЙ ВИД ПРОГРАММЫ

program имя_программы; {Заголовок программы, необязателен}
const {Раздел описания констант}
label {Раздел описания меток}
type {Раздел описания типов}
var {Раздел описания переменных}
{Описания подпрограмм}
Procedure {Описание процедуры}
begin {Начало тела процедуры}
{Тело процедуры}
end; {Конец тела процедуры}

Function {Описание функции}
begin {Начало тела функции}
{Тело функции}
end; {Конец тела функции}

ЗАКЛЮЧЕНИЕ

Pascal – один из самых известных языков программирования. Благодаря своей четкости, логичности и другим особенностям Паскаль надолго занял свою нишу, являясь прекрасным языком для обучения программированию.

Паскаль – самый удобный язык для изучения основ профессионального программирования. Среди них вычисления и обработка информации, использование объектно-ориентированного программирования – словом, те задачи, с которыми приходится сталкиваться профессиональному программисту.

Паскаль – достаточно «старый» программный продукт. Следует заметить, однако, что Паскаль – это живой язык. Известны, используются или находятся в стадии разработки компиляторы и среды разработки программ на Паскале для различных операционных систем.

В следующих презентациях я постараюсь изложить основные алгоритмические конструкции и команды языка Паскаль.