

Программирование

Назначение *программирования* –
разработка программ управления
компьютером с целью решения
различных информационных задач.

Программирование

Системное

Прикладное

Системное программирование – разработка системного программного обеспечения: операционных систем, утилит и т.д.



Прикладное программирование – создание прикладных программ: редакторы, табличные процессы, игры, обучающие программы и т.д.

Для составления программ существуют языки разнообразные программирования.

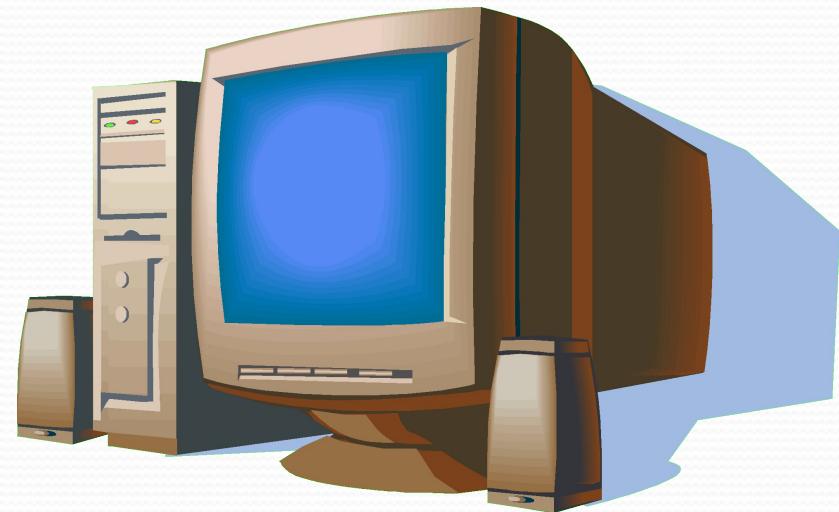
Язык программирования – это фиксированная система обозначений для описания алгоритмов и структур данных.



Для создания и использования на компьютере программы, написанной на языке программирования, используются системы программирования.

Система программирования – это программное обеспечение компьютера, предназначенное для разработки, отладки и исполнения программ, записанных на определенном языке программирования.

Разработка любой программы начинается с построения алгоритма решения задач. Такие алгоритмы называют алгоритмами работы с величинами.



В качестве исполнителя рассматривается – компьютер, оснащенный системой программирования на определенном языке.

Компьютер-исполнитель работает с определенными данными по определенной программе.

Язык
программирования
Паскаль

Язык Паскаль разработан в 1971 году и назван в честь Блеза Паскаля – французского ученого, изобретателя механической вычислительной машины.

Автор языка Паскаль – швейцарский профессор
Никлаус Вирт.



Паскаль – это универсальный язык программирования, позволяющий решать самые разнообразные задачи обработки информации

Процедуры вывода

Write и WriteLn

(переводится – «пиши» и «пиши строку»)

С помощью данных операторов изображают на экране ту или иную информацию, состоящую из символов.

Выводить на экран можно не только числа, но и результаты вычисления арифметических выражений, а также тексты, которые, в отличие от чисел и выражений, нужно брать в одинарные кавычки.

Примеры:

<i>Как пишем</i>	<i>Что видим</i>
Write(-500)	-500
Write($2*2-1$)	3
Write('Хорошо!')	Хорошо!

Один оператор Write может выводить сразу несколько элементов. Элементы нужно отделять друг от друга запятыми.

Все элементы выводятся в одну строку вплотную друг к другу.

На экране отображаются только те пробелы, которые встречаются внутри кавычек.

Примеры:

<i>Как пишем</i>	<i>Что видим</i>
Write('Это',4+4,'Кошек')	Это8Кошек
Write('Это ',4+4,' кошек')	Это 8 кошек
Write('16+17=',16+17)	16+17=33
Write(3+2,' ',4)	5 4
Write(3+2,4)	54
Write('125+1',5+1,'=',120+21)	125+16=141

Правила записи и выполнения оператора WriteLn те же, что и у Write, с одним исключением – после его выполнения следующий оператор Write или WriteLn печатает свою информацию с начала следующей строки, а после выполнения оператора Write продолжает печатать в той же.

Оператор WriteLn можно использовать просто для перевода курсора в начало следующей строки.

Программы на Паскале содержат
следующие «знаки препинания»:

- Служебные слова BEGIN и END;
- Точка с запятой;
- Точка.

BEGIN (переводится – «начало») – ставят в начале программы, чтобы было видно, откуда она начинается.

END (переводится – «конец») – с точкой ставится в конце программы, чтобы было видно, где она заканчивается.

Точкой с запятой отделяют операторы друг от друга.

Служебные слова BEGIN и END от операторов точкой с запятой не отделяются.

Пример:

Программа на Паскале. Результат выполнения

BEGIN

Write('Начали!');

Write(8+1);

Write(5);

END.

Начали!95

Программу можно записывать и в строку,
и в столбец.

Служебные слова и операторы могут
быть записаны любыми буквами
(заглавными или строчными, а также любым
шрифтом).

Программа на Паскале может содержать
комментарии, взятые в фигурные скобки,
которые не влияют на выполнение
программы.

Пример: Программа на Паскале.

```
BEGIN  
Write('Начали!'); {Это приказ печатать!}  
Write(8+1);  
Write(5);  
END.
```

Результат выполнения

Начали!95

Примеры:

Программа: Begin Write('AMa'); Write('ЗОНКА'); End.

Результат: АМАЗОНКА

Программа: Begin Write('AMa'); WriteLn('ЗОНКА'); End.

Результат: АМАЗОНКА

Программа: Begin WriteLn('Ама'); Write('Зонка'); End.

Результат:

Ама

Зонка

Программа: Begin WriteLn('Ама'); WriteLn('Зонка'); End.

Результат:

Ама

Зонка

Задача 1

Определить, что напечатает программа:

Begin

Write(1992);

WriteLn(' Мы начинаем!');

WriteLn(6*8);

WriteLn;

WriteLn('Шестью шесть ',6*6,'.Арифметика:',(6+4)*3);

End.

Оператор присваивания.

При выполнении оператора присваивания компьютер «в уме» вычисляет правую часть и присваивает вычисленное значение переменной, стоящей в левой части.

Обозначение оператора присваивания

\coloneqq

Пример:

Begin

a:=2*3+4;

b:=a;

y:=a+b+1;

Write('y=',y)

End.

a:=10;

b:=10;

y:=10+10+1;

y=21

Замечание. Если переменная принимает новое значение, то старое значение автоматически стирается

Описание переменных

Описание переменных начинается со служебного слова VAR (переводится – «переменная»), которое записывается выше Begin.

После VAR записываются имена всех переменных, встречающихся в программе с указанием через двоеточие типа значений, которые каждая переменная имеет право принимать.

Типы значений переменных

Тип	Перевод	Диапазон принимаемых значений
Integer	целый	целые числа от - 32 768 до 32 767
LongInt	длинное целое	целые числа от - 2 147 483 648 до 2 147 483 647
Byte		целые числа от 0 до 255
Real	Вещественный	целые и дробные числа

Для того, чтобы Паскаль выводил вещественные числа в понятном виде, нужно в оператор вывода WriteLn дописывать формат численного значения переменной:

WriteLn(x:n:m),

где **n** – натуральное число, показывающее сколько символов, включая целую часть, дробную часть, знак и десятичную точку, должно занимать все изображение числа; **m** – натуральное число, показывающее количество символов после десятичной точки.

Пример:

```
Var a,b:Integer;
```

```
    c:Real;
```

```
Begin
```

```
    a:=6;
```

Ответ: $c=1,17$

```
    b:=7;
```

```
    c:=b/a;
```

```
    WriteLn('c=',c:4:2);
```

```
End.
```

СПИСОК ИСТОЧНИКОВ

- Информатика и ИКТ: Учебник для 9 класса / И.Г.Семакин Л.А. Залогова, С.В. Русаков, Л.В. Шестакова.
- <http://informaticweb.altervista.org/images/sistemaoperativi.jpg>
- (немного изменен)
<http://seogad.ru/wp-content/uploads/2011/02/12-02-2011-soft.jpg>
- http://farm4.static.flickr.com/3227/2662210806_c7bbc90786.jpg
- <http://www.terageeks.net/images/computer2.gif>
- <http://informatika.mksat.net/wp-content/uploads/2012/01/virt.jpg>