

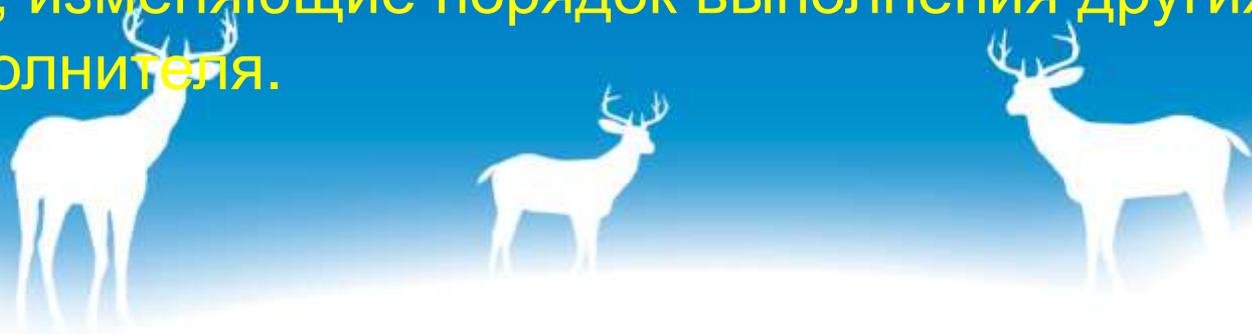
Основные алгоритмические конструкции



- **Алгоритм** — конечная последовательность точных действий, формальное выполнение которых позволяет получить решение задачи для любого допустимого набора исходных данных.
- **Исполнитель** — человек, группа людей или техническое устройство, которые способны правильно выполнять команды алгоритмов.
- Набор команд одного исполнителя называют **системой команд исполнителя**.

Все команды исполнителя делят на группы:

- 1. Команды, которые непосредственно выполняет исполнитель.
- 2. Команды, изменяющие порядок выполнения других команд исполнителя.



Базовые алгоритмические конструкции

1. **Следование** — последовательность команд алгоритма, которые выполняются в том порядке, в котором они записаны. Среди команд, образующих конструкцию *следование*, отсутствуют команды, меняющие порядок выполнения других команд.
2. **Цикл** — последовательность команд, выполняемых многократно.
3. **Ветвление** — последовательность команд, выполнение которых зависит от истинности или ложности проверяемого условия.

Команды цикла и ветвления управляют порядком выполнения других команд в программе и относятся к командам управления (**управляющим конструкциям**).

В 7-м классе, изучая язык Pascal, вы использовали следующие команды: процедуры для ввода и вывода данных; оператор присваивания.



Для **ввода данных** предназначена команда **read()**. В скобках через запятую перечисляются имена переменных, значения которых необходимо ввести.

Для **вывода данных** используют команду **write()**. Она позволяет выводить текстовые сообщения и числовые значения. Текстовые сообщения записываются в апострофах, выводятся в виде последовательности символов так, как записаны, и не анализируются при выполнении.

При использовании команды **writeln()**; после вывода сообщения или числа происходит перевод курсора на следующую строку.

Оператор присваивания предназначен для того, чтобы:
задавать значения переменным;
вычислять значение выражения (результат будет записан как значение переменной).

Формат записи оператора присваивания:

имя переменной:=выражение;

В записи арифметического выражения используются знаки математических действий: сложения (+), вычитания (-), умножения (*), деления (/), а также целочисленного деления (div) и нахождения остатка (mod). Следует помнить, что операция деления (/) используется при вычислениях с данными типа real. Для данных типа integer используются операции div и mod.

Пример 13.3. Даны x, y . Написать программу для вычисления значения выражения

$$a = \frac{2x}{7+y^2} (x-y).$$

Этапы выполнения задания:

I. Определение исходных данных: *переменные x, y .*

II. Определение результатов: *переменная a .*

III. Алгоритм решения задачи.

1. Ввод исходных данных.

2. Вычисление значения выражения.

3. Вывод результата.

IV. Описание переменных: все переменные, определенные для решения задачи, имеют тип `real`.

V. Программа:

```
var x,y,a: real;
```

```
begin
```

```
write('Введите x =');
```

```
read(x);
```

```
write('Введите y =');
```

```
read(y);
```

```
a:= 2 * x * (x-y)/(7 + y * y);
```

```
Writeln ('a =',a);
```

```
End.
```

VI. Тестирование программы: Запустить программу и ввести значения $x = 3.8, y = 2.7$.

VII. Правильность вычислений (проверить).



Математические функции

Запись на языке Pascal

Описание

abs (x)

Находит модуль числа x

sqr (x)

Возводит число x в квадрат

sqrt (x)

Находит корень квадратный из числа x . Результат —

всегда число типа `real`

trunc (x)

Находит целую часть действительного числа x (`real`).

Результат — число типа `integer`

frac (x)

Находит дробную часть действительного числа x (`real`).

Результат — число `real`

sin (x)

Вычисляет синус числа x . Число x задается в радианах

cos (x)

Вычисляет косинус числа x . Число x задается в радианах

RadToDeg (x)

Переводит радианы в градусы

DegToRad (x)

Переводит градусы в радианы



Пример 13.6. Задана длина стороны квадрата a . Написать программу нахождения площади квадрата и длины его диагонали.

Этапы выполнения задания:

I. Исходные данные: **длина стороны, переменная a .**

II. Результат: **переменные S (площадь) и d (длина диагонали).**

III. Алгоритм решения задачи.

1. Ввод исходных данных.

2. Вычисление площади по формуле $S = a^2$.

3. Вычисление длины диагонали по формуле $d = a \sqrt{2}$

4. Вывод результата.

IV. Описание переменных: все переменные, определенные для решения задачи, имеют тип `real`.

V. Программа:

```
var a, S, d: real;
```

```
begin
```

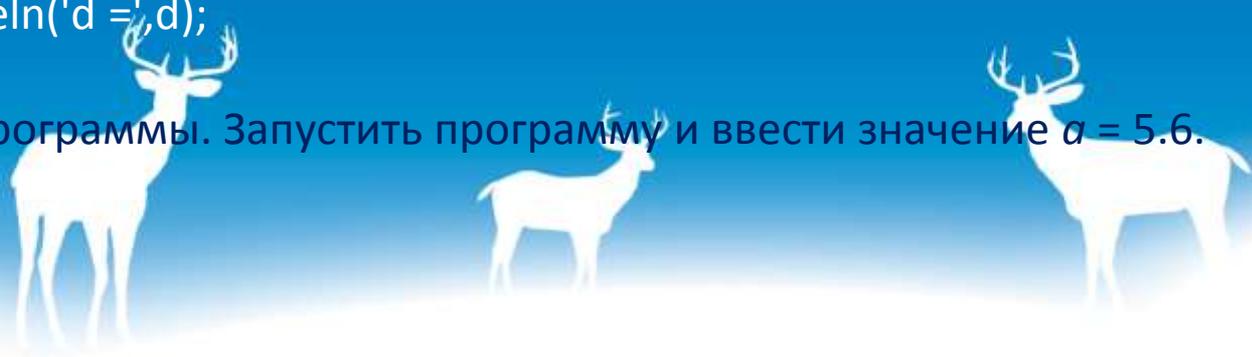
```
write('Введите a ='); read(a);
```

```
S:= sqr(a); d:= a*sqrt(2);
```

```
writeln('S =',s); writeln('d =',d);
```

```
end.
```

VI. Тестирование программы. Запустить программу и ввести значение $a = 5.6$.



Упражнения

1. Расставьте команды программы в правильном порядке так, чтобы можно было вычислить значение выражения $a = \frac{2x}{x^2 + 4}$

1. `writeln('a = ',a);`

2. `write('Введите значение x = ');`

3. **End.**

4. **Var** x,y,a: real;

5. **Begin**

6. `a:=2*x/(sqr(x)+4);`

7. `read(x);`



2. Найдите и исправьте ошибки в программах.

❖ 1.

```
var x, y, z1, z2:integer;  
begin  
write('Введите x =');  
read(x);  
write('Введите y =');  
read(y);  
z1:= int(x/y);  
z2:= frac(x/y);  
write('Целая часть =',z1);  
write('Дробная часть =',z2);  
end.
```

❖ 2.

```
var x, y, z1, z2:real;  
begin  
write('Введите x =');  
read(x);  
write('Введите y =');  
read(y);  
z1:=x div y;  
z2:=x mod y;  
write('Целая часть =',z1);  
write('Остаток =',z2);  
end.
```

