

# Алгоритм работы с величинами

- Алгоритм работы с величинами
- Оператор присваивания
- Оператор ввода данных
- Оператор вывода

# Алгоритм работы с величинами

---

Всякий алгоритм строится исходя из системы команд исполнителя, для которого он предназначен.

Независимо от того, на каком языке программирования будет написана программа, алгоритм работы с величинами, обычно, составляется из следующих команд:

- присваивание;
- ввод;
- вывод;

# Оператор присваивания

---

Значения переменным задаются с помощью *оператора присваивания*.

*Команда присваивания – одна из основных команд в алгоритмах работы с величинами.*

При присваивании переменной какого-либо значения старое значение переменной стирается и она получает новое значение.

# Оператор присваивания

---

В языках программирования команда присваивания обычно обозначается либо «:=» (**двоеточие и равно**), либо «=» (**равно**).

Значок «:=» (или «=») читается «*присвоить*».

Язык программирования Qbasic для обозначения оператора присваивания использует знак «=».

# Оператор присваивания

**Например:**      $z = x + y$

- 1) Компьютер сначала вычисляет выражение  $x + y$ ,
- 2) затем результат присваивает переменной  $z$ , стоящей слева от знака «= $\gg$ ».

# Выполнение оператора присваивания

---

Если до выполнения этой команды содержимое ячеек, соответствующих переменным  $x$ ,  $y$ ,  $z$ , было таким:

$x$	$y$	$z$
2	3	-

Прочерк в ячейке  $z$  обозначает, что начальное число в ней может быть любым.

Оно не имеет значения для результата данной команды.

то после выполнения команды  $z := x + y$  оно станет следующим:

$x$	$y$	$z$
2	3	5

# Арифметические выражения

---

Если слева от знака присваивания стоит числовая переменная, а справа – математическое выражение, то такую команду называют арифметической командой присваивания, а выражение – арифметическим.

В частном случае арифметическое выражение может быть представлено одной переменной или одной константой.

*Например:*

$x := 7$

$a := b + 10$

$c := x$

# Ввод переменных в компьютер

---

Значения переменных, являющихся исходными данными решаемой задачи, как правило, задаются вводом.

На современных компьютерах ввод чаще всего выполняется в режиме диалога с пользователем:

- 1) по команде ввода компьютер прерывает выполнение программы и ждет действий пользователя;
- 2) пользователь должен набрать на клавиатуре вводимые значения переменных и нажать клавишу <ВВОД>.
- 3) введенные значения присвоятся соответствующим переменным из списка ввода, и выполнение программы продолжится.



# Ввод переменных в компьютер

---

**Команд ввода** в описаниях алгоритмов обычно выглядит так:

ввод <список переменных>

На языке программирования **Qbasic** оператор ввода выглядит так:

```
input [«пояснительный текст»;] a, b, c
```

[\*] – необязательная часть

# Схема выполнения приведенной выше команды

1. Память до выполнения команды:

<b>a</b>	<b>b</b>	<b>c</b>
-	-	-

2. Компьютер получил команду **ввод (a, b, c)**, прервал свою работу и ждет действий пользователя.

3. Пользователь набирает на клавиатуре:

1 3 5

и нажимает клавишу <ВВОД> (<Enter>).

При выполнении пункта 3 вводимые числа должны быть отделены друг от друга какими-нибудь разделителями. Обычно это пробелы.

4. Память после выполнения команды: вместо «-» записываются 1 3 5.

# Значения величин

---

*Переменные величины получают конкретные значения в результате выполнения команды присваивания или команды ввода.*

Если переменной величине не присвоено никакого значения (или не введено), то она является неопределенной. Иначе говоря, ничего нельзя сказать, какое значение имеет эта переменная.

# Вывод результатов на экран

---

Результаты решения задачи сообщаются компьютером пользователю путем выполнения *команды вывода*.

Команда вывода в описаниях алгоритмов обычно выглядит так:

**ВЫВОД <СПИСОК ВЫВОДА>**

На языке программирования *Qbasic* оператор ввода выглядит так:

```
print [«пояснительный текст»;] x, y
```

[\*] – необязательная часть

*Например: **ВЫВОД (x1, x2).***

По этой команде значения переменных *x1* и *x2* будут вынесены на устройство вывода (чаще всего это экран).

# Задание: алгоритм вычисления периметра треугольника.

---

Составим алгоритм вычисления периметра треугольника.

Нам потребуется 4 переменных для хранения значения длин сторон треугольника и его периметра.

Периметр – это сумма всех сторон.

*Алгоритм Вычисление периметра треугольника*

*переменные  $a, b, c, p$  - целые*

**начало**

**ввод ( $a, b, c$ )**

**$p := a + b + c$**

**вывод ( $p$ )**

**конец**

На языке программирования **Qbasic**

CLS

input «введем длины сторон треугольника»;  $a, b, c$

$P = a + b + c$

print «периметр треугольника равен»;  $p$

## Задание: алгоритм вычисления периметра треугольника.

---

Сначала компьютер запросит значения переменных  $a$ ,  $b$ ,  $c$  у пользователя,

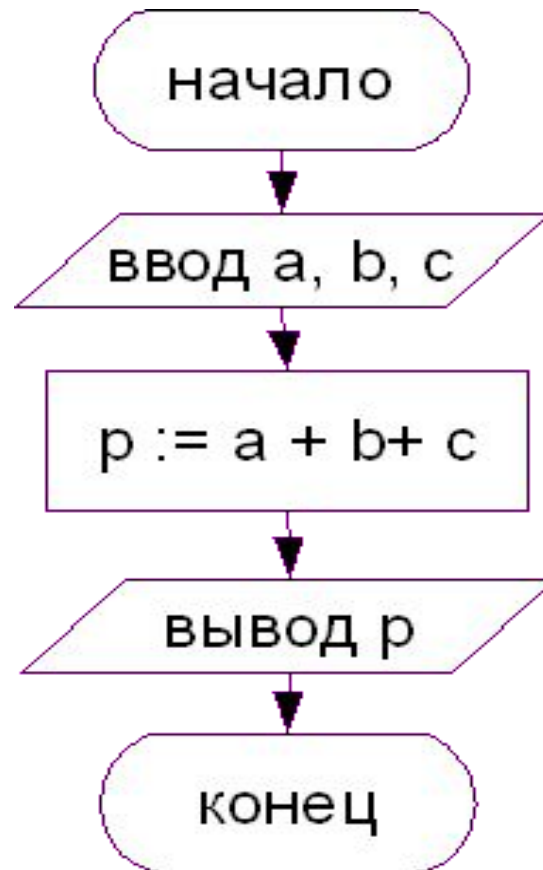
затем произведет вычисления и выведет результат на экран.

Строка *переменные  $a, b, c, p$  - целые* - называется описанием переменных.

Некоторые языки программирования требуют обязательного описания всех переменных до начала их использования в программе, некоторые – относятся более лояльно.

Полученный алгоритм имеет линейную структуру

---



# Задание

---

Составить алгоритм и линейную структуру вычисления площади трапеции

$$S = (a + b)h / 2,$$