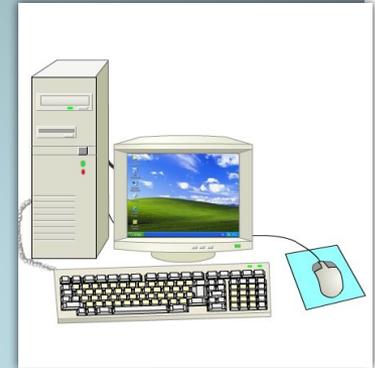


Переменные: тип, имя, значение

- ❑ Переменные, величины. Типы, имена переменных
- ❑ Хранение величин
- ❑ Алгоритм работы с величинами
- ❑ Оператор присваивания

Компьютер - исполнитель



Всякий алгоритм составляется для конкретного исполнителя.

Компьютер - исполнитель работает с определенными данными по определенной программе.

Программа – это алгоритм записанный на каком-либо языке программирования.

Данные – это множество величин.

Величины

Отдельный информационный объект (число, символ, строка, таблица и пр.) называется величиной.

Величины в программировании, как и в математике, делятся



переменные

значение величины
может изменяться



константы

Значение остается
неизменной в течении всей
программы

Переменные

Для того чтобы программа обладала универсальностью, действия в ней должны совершаться не над постоянными, а над **переменными** величинами.

Поэтому важным понятием программирования является понятие переменной.

Переменная в программе представлена именем и служит для обращения к данным определенного типа, конкретное значение которых хранится в ячейке оперативной памяти.

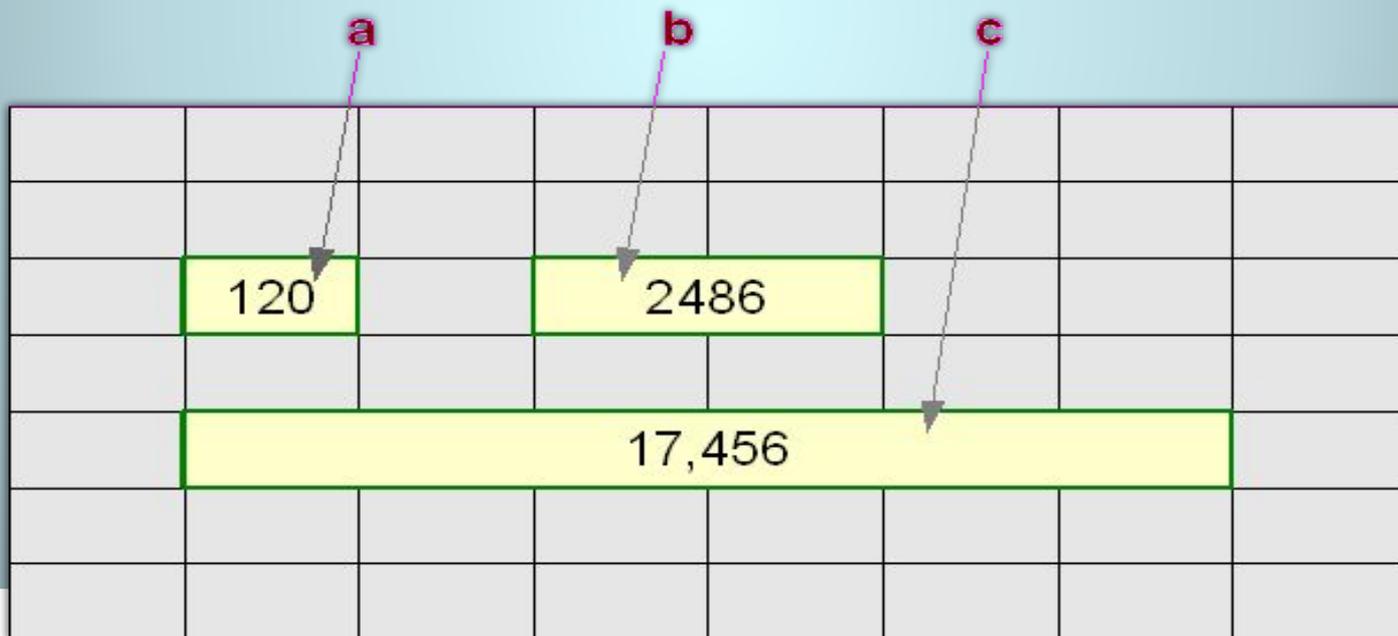
Компьютер работает с информацией, хранящейся в его памяти.



Хранение величин



В зависимости от типа переменной в памяти компьютера будет выделена определенная область. Данные различных типов требуют для своего хранения в оперативной памяти компьютера различное количество ячеек (байтов).

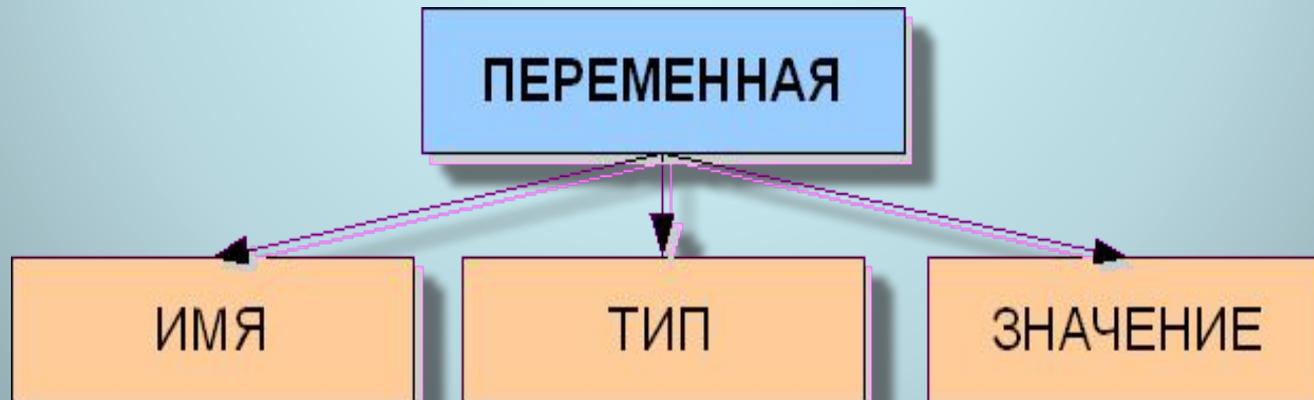


Имена переменных

У каждой переменной есть *имя*, *тип* и *текущее значение*.
Имена переменных называют идентификаторами (от глагола «идентифицировать», что значит «обозначать», «символизировать»). *Имя уникально и не может изменяться в процессе выполнения программы.*

В качестве имен переменных могут быть буквы, цифры и другие знаки. Причем может быть не одна буква, а несколько.

Примеры идентификаторов: a, b5, x, y, x2, summ a, bukva10...
Количество символов не может быть больше 1023.



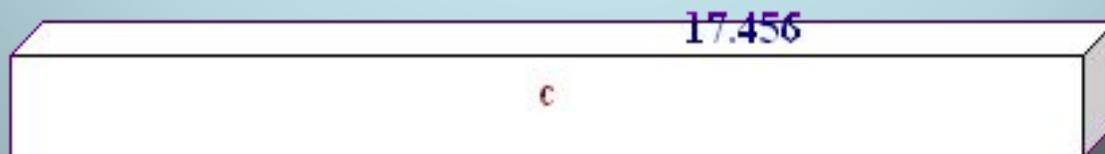
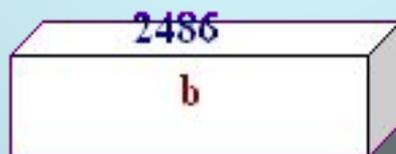
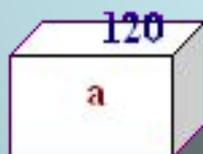
Имя и значение переменной

Наглядно переменную можно представить как коробочку, в которую можно положить на хранение что-либо.

Имя переменной – это надпись на коробочке,

значение – это то, что хранится в ней в данный момент,

а тип переменной говорит о том, что допустимо класть в эту коробочку.



Типы величин

Тип переменной определяется типом данных, которые могут быть значениями переменной.

Существуют три основных типа величин, с которыми работает компьютер (*Visual Basic 2005*):

- **ЧИСЛОВОЙ** (значения переменных типа *Byte, Short Integer, Long, Single, Double* в виде чисел);
- **СТРОКОВЫЙ** (строковый тип *String* – последовательность символов);
- **ЛОГИЧЕСКИЙ** (значения логического типа-«истина» (*True*) или «ложь» (*False*))

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»*.

Оператор присваивания

Например: $z := x + y$

1) Компьютер сначала вычисляет выражение $x + y$,

2) затем результат присваивает переменной z , стоящей слева от знака «:=».

Оператор 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.

Оператор присваивания

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>

или

ввод (<список переменных>)

Схема выполнения приведенной выше команды

1. Память до выполнения команды:

a	b	c
-	-	-

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>

или

ВЫВОД (<СПИСОК ВЫВОДА>)

Например: **ВЫВОД (x1, x2).**

По этой команде значения переменных x_1 и x_2 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

Задание: алгоритм вычисления периметра треугольника.

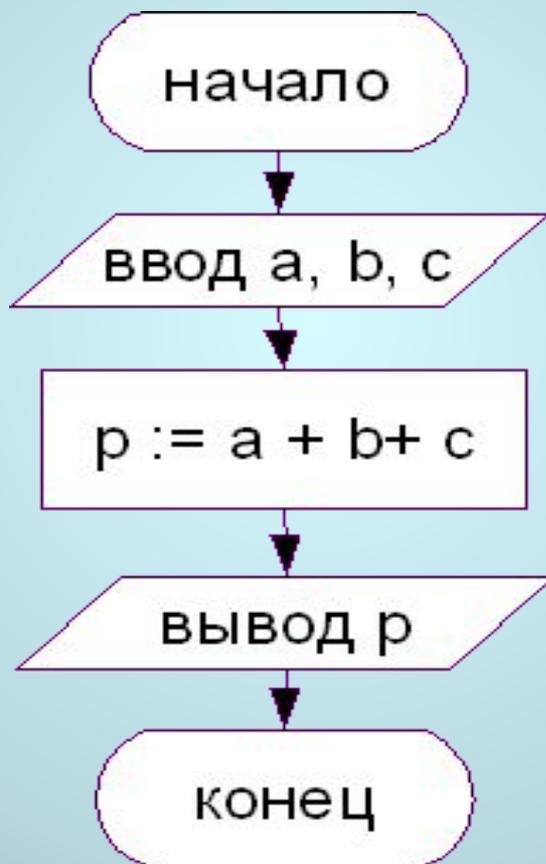
Сначала компьютер запросит значения переменных a , b , c у пользователя,

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 площади трапеции

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

Домашнее задание

Составить алгоритм и линейную структуру любой геометрической формулы

ИТОГ УРОКА

1. *Отдельный информационный объект (число, символ, строка, таблица и пр.) называется величиной: переменные и константы.*
2. *Переменная в программе представлена именем и служит для обращения к данным определенного типа, конкретное значение которых хранится в ячейке оперативной памяти.*
3. *У каждой переменной есть имя, тип и текущее значение. Имя уникально и не может изменяться в процессе выполнения программы. В качестве имен переменных могут быть буквы, цифры и другие знаки. Причем может быть не одна буква, а несколько.*
4. *Существуют три основных типа величин: числовой (обозначение переменных в виде чисел), строковый (последовательность символов), логический (переменные типа «ложь», «истина»).*
5. *Алгоритм работы с величинами, обычно, составляется из следующих команд: присваивание; ввод; вывод.*