

*Введение в языки
программирования*

Основные вопросы:

- **Язык программирования Basic. Компоненты алгоритмического языка (*алфавит, синтаксис, семантика*)**
- **Величины. Константы и переменные. Характеристики переменных.**
- **Запись арифметических выражений на языке программирования.**
- **Организация программ.**
- **Операторы ввода-вывода (Input-Print)**
- **Операторы Rem, Cls.**
- **Составление линейных программ.**

Языки программирования

– это формальные языки специально созданные для общения человека с компьютером.

Машинный язык

– это такой язык, который компьютер воспринимает непосредственно, т.е. это язык машинных команд данной модели компьютера

Процесс описания конкретного алгоритма на языке машинных команд называется *программированием в машинных кодах.*

Поскольку машины «понимает» только свой машинный язык, программа не на алгоритмическом языке перед выполнением переводится на этот язык с помощью специальной программы – *транслятора*, название которой происходит от английского слова *translator* (переводчик).

Транслятор

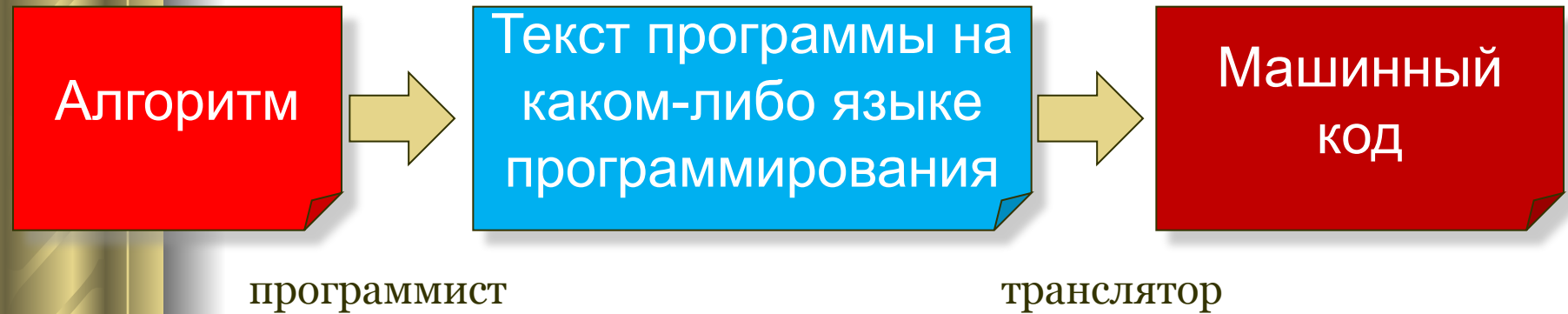
Транслятор — это программа-переводчик для преобразования текстов из одного языка, понятного человеку, в другой язык, понятный компьютеру.



последовательно анализирует и исполняет каждую строку программы

проводит полный анализ написанной программы и формирует уже готовый к исполнению машинный код

Этапы создания программы



Очистить экран
Ввести a
Вычислить площадь
Вывести результат

```
CLS  
INPUT a  
s=a*a  
PRINT s  
END
```

```
001011101010  
010101100101  
010111010110  
101101111010
```



- **Языки программирования** – это формальные языки специально созданные для общения человека с компьютером.
- **Бейсик** – язык программирования, т.е. средство для записи алгоритма решения задачи в виде, понятном для ЭВМ, в виде программы для ЭВМ.

Компоненты алгоритмического языка:

- *Алфавит* – фиксированный для данного языка набор основных символов, допускаемых для составления текста программы на этом языке.
- *Синтаксис* – система правил, определяющих допустимые конструкции языка программирования из букв алфавита.
- *Семантика* - система правил однозначного толкования отдельных языковых конструкций, позволяющих воспроизвести процесс обработки данных.



Алфавит языка Бейсик

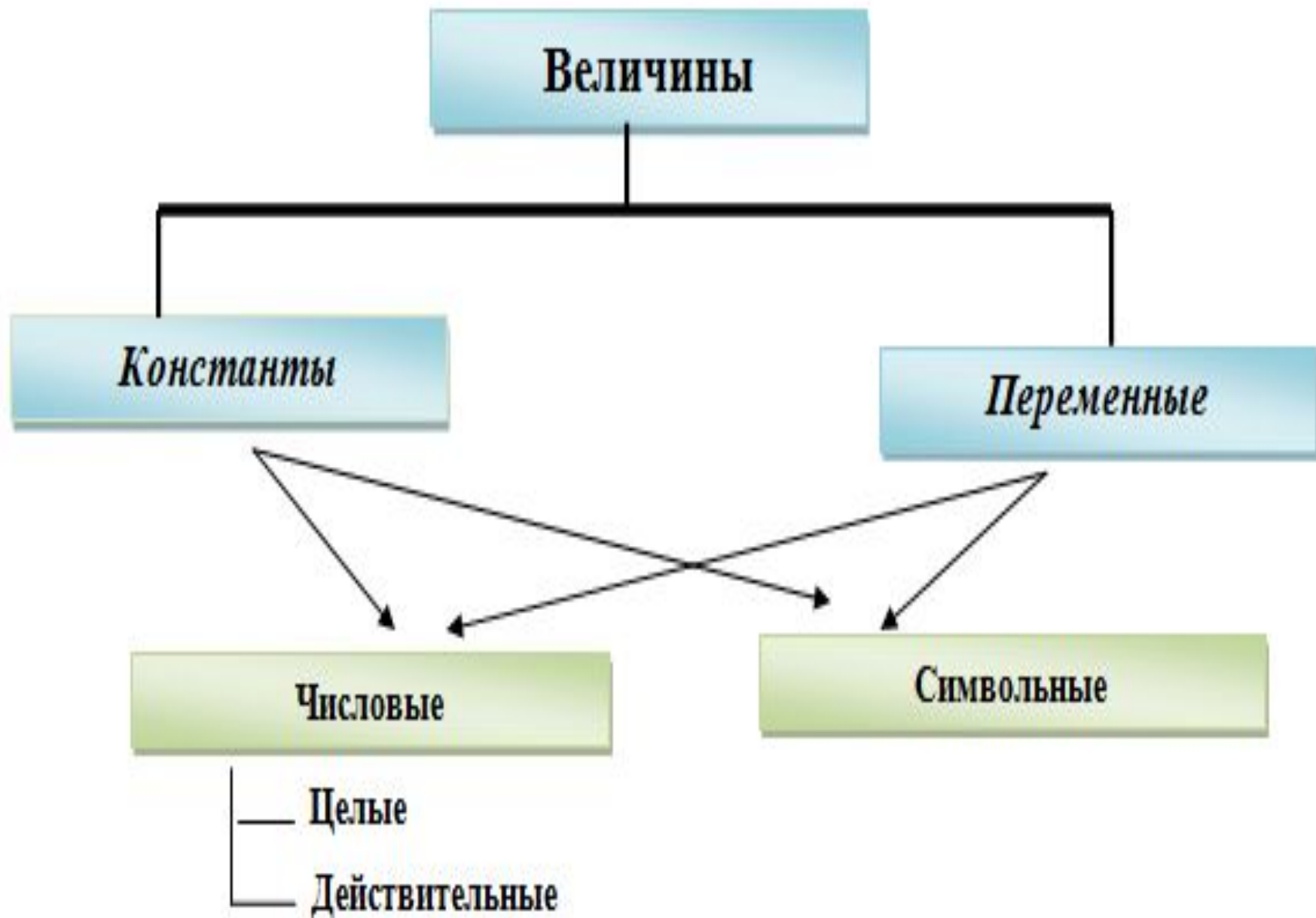
- **заглавные (прописные) буквы латинского алфавита** – А, В, С, D, Е, F, G, H, I, J, K, L, M, N, O, P, Q, R, S, T, U, V, W, X, Y, Z;
- **арабские цифры** – 1,2,3,4,5,6,7,8,9,0;
- **разделители** – запятая (,), точка с запятой (;), точка (.), двоеточие (:), апостроф('), кавычки (“”), открывающаяся скобка (, закрывающаяся скобка) , символ Пробел;
- **знаки арифметических операций**
- **знаки операций отношений**

Арифметические операции на языке Basic.

Операция	Обозначение	Пример	Результат
<i>Сложение</i>	+	2+5	7
<i>Вычитание</i>	-	10-8	2
<i>Умножение</i>	*	3*4	12
<i>Деление</i>	/	15/3	5
		15/4	3.75
<i>Целочисленное деление</i>	\	15\4	3
<i>Возведение в степень</i>	^	2^3	8
<i>Остаток от деления</i>	MOD	13 MOD 5	3

Знаки операций отношения

Значение операции	Знак операции	Выражение в Бейсике
<i>Равенство</i>	=	$x=y$
<i>Неравенство</i>	$\langle \rangle$	$x\langle \rangle y$
<i>Меньше</i>	<	$x<y$
<i>Больше</i>	>	$x>y$
<i>Меньше или равно</i>	\leq	$x\leq y$
<i>Больше или равно</i>	\geq	$x\geq y$



Переменная - это величина, которая может меняться при выполнении программы.

Кроме указанных в таблице, целые числа записывают также в **экспоненциальной записи**, например,
 $2,55 \cdot 10^5 = 2.55E5$; $7,15 \cdot 10^{-7} = 7.15E-7$

Тип переменной	Символ	Описание	Объём	Пример
Целые числа	%	integer	2 байта	17; 123
Действит. числа	нет	real	4 байта	3.1415
Символьный	\$	string	min 1бай	язык
Пользовательский	нет	type		
Целый 2 точности	#	double	8 байт	1.2543786

Идентификаторы

- это произвольный набор символов, который может содержать от 1 до 40 символов, причём **первый символ должен быть латинской буквой**, а остальные – латинские буквы или цифры или символы типа @, #, % и т.д.

Пример:

A, Ds, SodRan, k1, n123, dlina!

Тип идентификатора в Бейсике опознаётся по последнему символу в имени переменной.

% - целое число;

& - длинное целое число;

! – вещественное число обычной точности;

- вещественное число двойной точности;

\$ - символьный тип.

- *Значение переменной* – это содержимое тех ячеек памяти, в которых хранится переменная.
- *Имя переменной* – это «где хранится», а значение переменной – «что хранится».
- Присваивая новое значение переменной, мы теряем старое. Рекомендуется имена переменных давать по смыслу содержащихся в них значений.

- ***Константами*** называются величины, которые имеют постоянные значения, **не изменяющиеся во время выполнения операций.**
- В Basic существует два типа констант – ***числовые*** и ***символьные*** (или литерные).

- ***Символьные (литерные) константы*** – последовательность любых отображений символов, заключается в кавычках. В цепочке символов могут быть любые символы языка (запятая, точки, пробелы и т.п.), кроме кавычек.

Примеры:

«Введите два разных числа»

«10.03.1994год»

- Допускается случай, когда ***литерная константа не содержит ни одного символа***; соответствующий текст (пусто) обозначается двумя кавычками, следующими друг за другом: “ ”.

Арифметические выражения

*- представляют собой
совокупность констант,
переменных, стандартных
функций, соединенных знаками
арифметических операций и
круглыми скобками.*



АРИФМЕТИКА В БЕЙСИКЕ

Элементарные арифметические выражения необходимо представить в виде, понятном для компьютера:

1. выражение должно быть записано в одну строку
2. для записи арифметических действий использовать знаки: + - * / ^ ()
3. недопустим пропуск знака умножения между *коэффициентом и переменной* (нельзя $2x$, надо $2*X$);
4. дробная часть отделяется от целой точкой, а не запятой (нельзя $3,14$, надо 3.14);
5. допустимо опускать в записи десятичной дроби ноль, стоящий перед точкой (вместо 0.123 можно $.123$);
6. необходимо соблюдать приоритет выполнения действий –
 - действия в скобках
 - вычисление функций
 - возведение в степень
 - умножение и деление
 - сложение и вычитание
 - действия одинаковой очередности выполняются слева направо.

Математические функции на языке Basic.

Название функции	Запись в математике	Запись в Бейсике	Пример в Бейсике
Абсолютная величина	$ x $	ABS(X)	ABS(-5)
Экспонента	e^x	EXP(X)	EXP(5)
Логарифм натуральный	$\ln x$	LOG(X)	LOG(3)
Остаток целочисленного деления (MOD)	-	x MOD y	19 MOD 6.7
Квадратный корень	\sqrt{x}	SQR(X)	SQR(4)
Синус	$\sin x$	SIN(X)	SIN(3.14)
Косинус	$\cos x$	COS(X)	COS(1)
Тангенс	$\operatorname{tg} x$	TAN(X)	TAN(2)
Арктангенс	$\operatorname{arctg} x$	ATN(X)	ATN(3)
Знак числового выражения	-	SGN(X)	SGN(-5)

Правила записи функций

- Стандартные функции могут входить в **арифметические выражения**. В качестве аргументов также можно использовать арифметические выражения. Аргумент обязательно записывается в скобках **SIN (3), ABS (X)**

Примеры правильной записи стандартных функций

Алгебра	Бейсик
$\sin 3 + \cos 6 + \operatorname{tg} 4$	<code>SIN(3) +COS(6) +TAN(4)</code>
$\sqrt{ \sin 45^{\circ} - \cos^2 30^{\circ} }$	<code>SQR(ABS(SIN(45*3.14/180))-COS(30*3.14/180)^2)</code>

Тригонометрические функции

- ◆ 30° записывается как $30*3.14/180$ так как $1^{\circ} = \pi/180$
- ◆ Таким образом
 $\operatorname{SIN} 30^{\circ}$ запишется как $\operatorname{SIN} (30*3.14/180)$

Практическая часть: Запись математических выражений

$$I = \frac{U}{R}$$

$$I=U/R$$

$$T = 2\pi \sqrt{\frac{l}{g}}$$

$$T=2*3.14*SQR(L/G)$$

$$s = v_0 t + \frac{at^2}{2}$$

$$s=v0*t+(a*t^2)/2$$

Переведите выражения на Basic

Математическое выражение

$$\sqrt{X + a}$$

SQR(X+A)

$$\frac{a + 4,5}{2,8 - 3k}$$

(A+4.5) / (2.8-3*k)

$$\sqrt[4]{b^3 - 12}$$

(B^3-12)^(1/4)

$$7\frac{5}{2} - 3\frac{13}{5}$$

(7+5/2) - (3+13/5)


$$|\sin^2(\beta + \alpha)|$$


ABS(SIN(B+A)^2)

$$\sqrt[4]{\operatorname{tg}|x - y|}$$

TAN(ABS(X-Y))^(1/4)

Организация программ

 Оператор- это приказ машине выполнить какое-либо действие, например: изменить значение переменной, изменить порядок выполнения операторов в программе , вывести на экран текущее значение.

 Программа, составленная на языке программирования, представляет собой последовательность операторов-команд, выполнение которой приводят к искомому результату.

Оператор присваивания

$$Z = (324 * x - 2 * y^3) + 525$$

Имя переменной, в которую
будет занесён результат

Выражение, результат которого
Вычисляется (может стоять
конкретное число)

Принцип работы оператора:

1. Вычисляется значение выражения
2. При работе с арифметическими типами результат вычисления, если это необходимо, преобразуется к типу переменной
3. Полученный результат присваивается переменной, на прежнее её значение теряется.

Оператор **INPUT**

Оператор **INPUT** вводит значение переменной с клавиатуры в память компьютера.

INPUT “подсказка”; список переменных

INPUT в переводе с английского языка означает *вставлять, вводить*.

При встрече с оператором **INPUT** программа приостанавливает своё действие; на экране появляется знак вопроса **?**, после которого *необходимо набрать на клавиатуре значение переменной, входящий в состав оператора **INPUT**, нажать клавишу **Enter***.

Оператор **INPUT** можно использовать для присваивания значения как числовым, так и строковым переменным

.

Примеры записи оператора

1. **INPUT X**
2. **INPUT A,B,C**
3. **INPUT «Каков Ваш рост в см»;X**
4. **INPUT «Введите длины двух катетов»;A,B**

- Оператор позволяет *запрашивать ввод нескольких переменных через запятую*, позволяет записать подсказку для ввода переменных, но только одну.

- Если вы случайно ввели больше или меньше переменных, после нажатия клавиши ENTER появится сообщение **«Ввод сначала»**, т.е. вы должны осуществить ввод заново.

Алгоритм записи программы с оператором **INPUT**

Задача: запросить у пользователя год рождения и вывести на экран число лет пользователя.

1. Очищаем экран
2. Запрашиваем год рождения (переменная N)
3. Вычисляем количество лет
4. Выводим полученное число на экран с соответствующими комментариями.

1. **CLS**
2. **INPUT “введите год рождения”; N**
3. **L=2010-N**
4. **PRINT “Вам ”;L;” лет”**

Оператор PRINT

- Выводить текстовую информацию, заключенную в кавычки, на экран монитора;

Например:

? «Привет»

- Вычислять выражения арифметических выражений ;

Например:

? 5*4-5.6^2

- Выводить значения переменных на экран монитора.

Например:

DAY\$=«понедельник»

? DAY\$

Вместо слова *PRINT* можно набирать знак ?

PRINT в переводе с английского языка означает *печатать*.

Оператор CLS очищает экран монитора.

Название оператора CLS произошло от английских слов Clear Screen, что в переводе означает *очистить экран*.

Например:

CLS

? «Привет»

На экране монитора в левом верхнем углу будет выведено слово *Привет*.

REM – оператор комментария. Все что следует после этого оператора до конца строки игнорируется компилятором и предназначено исключительно для человека. Т.е. здесь можно писать что угодно. Удобно использовать комментарий в начале программы для указания её названия и назначения.

Пример:

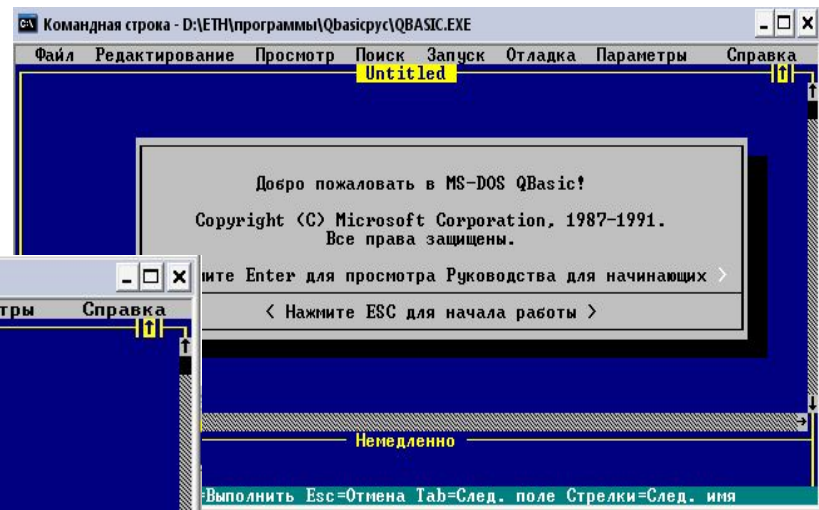
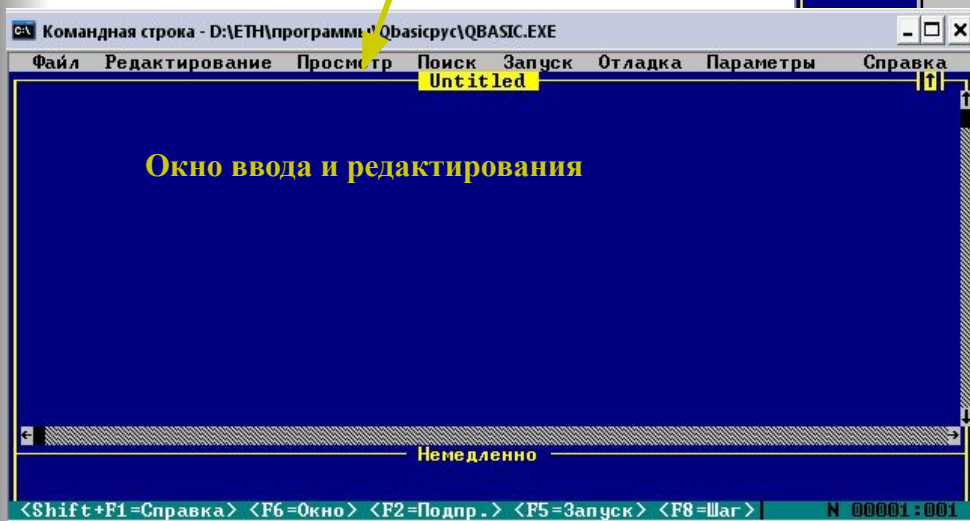
REM Это комментарий

МОЖНО И ТАК:

‘ **Это тоже комментарий**

Запуск Qbasic и начало работы

1. Найдите на рабочем столе ярлык **QuickBasic** и 2 ЛК по нему.
2. На экране появляется инструкция. Нажмите **Esc** для выхода в окно редактора.



Запуск Qbasic и начало работы

3. Для работы предоставляется два окна:
верхнее – **окно редактирования**;
нижнее – окно немедленного выполнения
(в нем вы можете сразу видеть результаты работы
набранных команд после нажатия клавиши **Enter**).
4. В окне редактора мигает курсор – он указывает, где
будет вводиться текст программы при
наборе его с клавиатуры.
5. **Если в программе была допущена ошибка**, то после
запуска программы на экране появится окно с
сообщением
об ошибке; при нажатии на **Enter** или **Esc** вы
вернетесь в окно редактирования, где курсор будет
находиться в том месте, где компьютер нашел
ошибку.
6. **Запуск программы на выполнение** – нажать клавишу
F5, если нет ошибок – увидите результат ее работы и
сообщение внизу экрана
«Чтобы продолжить, нажмите любую клавишу».



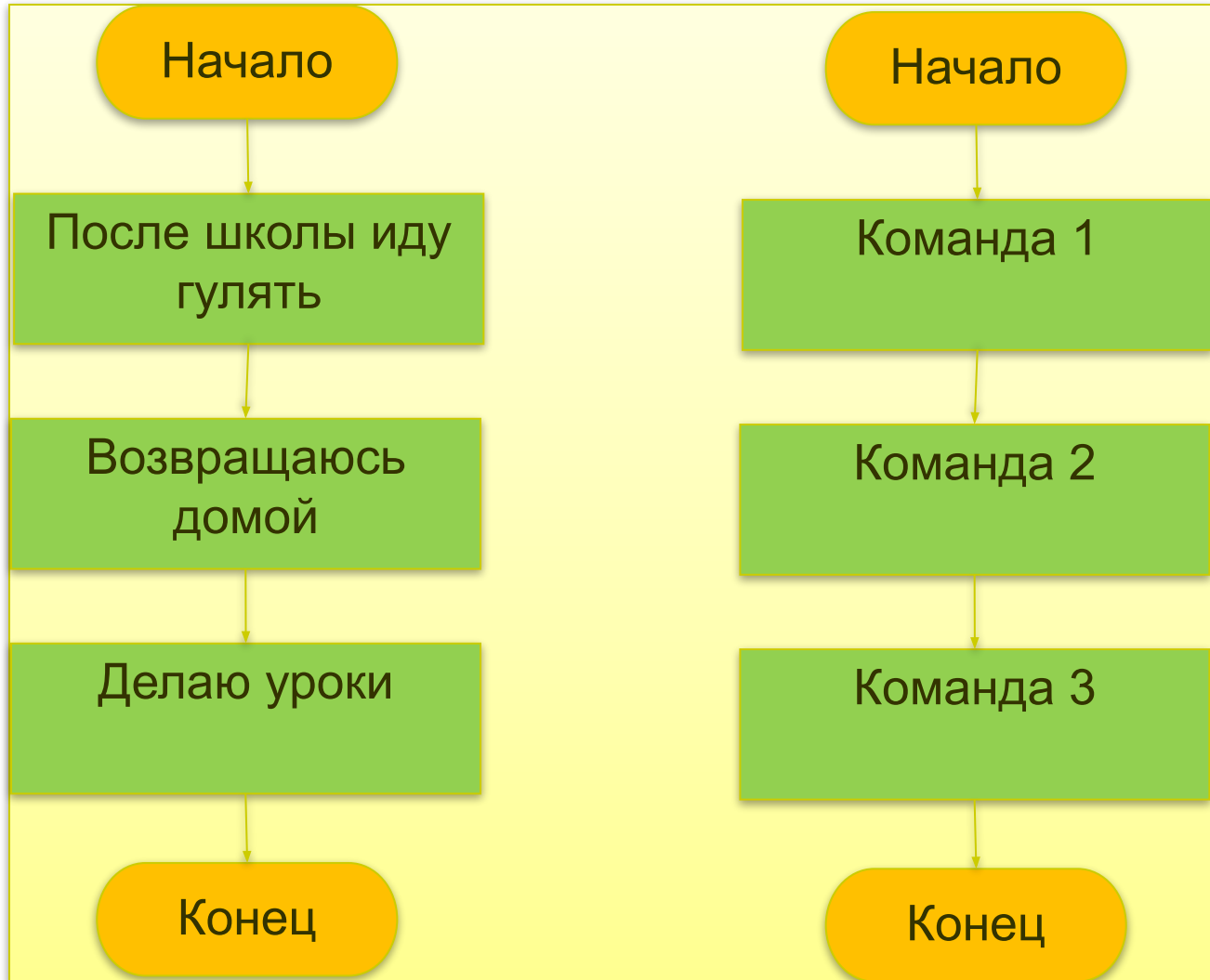
Линейная структура программы



Линейная структура программы

Программа имеет *линейную структуру*, если все операторы (команды) выполняются последовательно друг за другом.

Линейная структура программы



Примеры задач.

ЗАДАЧА 1. Найти периметр и площадь прямоугольника:

Вариант №1	Вариант №2	Вариант №3
<pre>rem периметр и площадь прямоугольника input a, b p=2*(a+b): s=a*b print "P="; p print "S="; s end</pre>	<pre>rem периметр и площадь прямоугольника input "a,b="; a,b p=2*(a+b): s=a*b print "P="; p: print "S="; s end</pre>	<pre>rem периметр и площадь прямоугольника input "a="; a input "b="; b print "P="; 2*(a+b) print "S="; a*b end</pre>

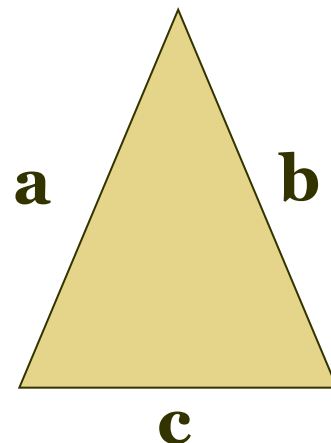
Задача 2.

Рассмотрим для примера программу вычисления площади треугольника по 3 сторонам.

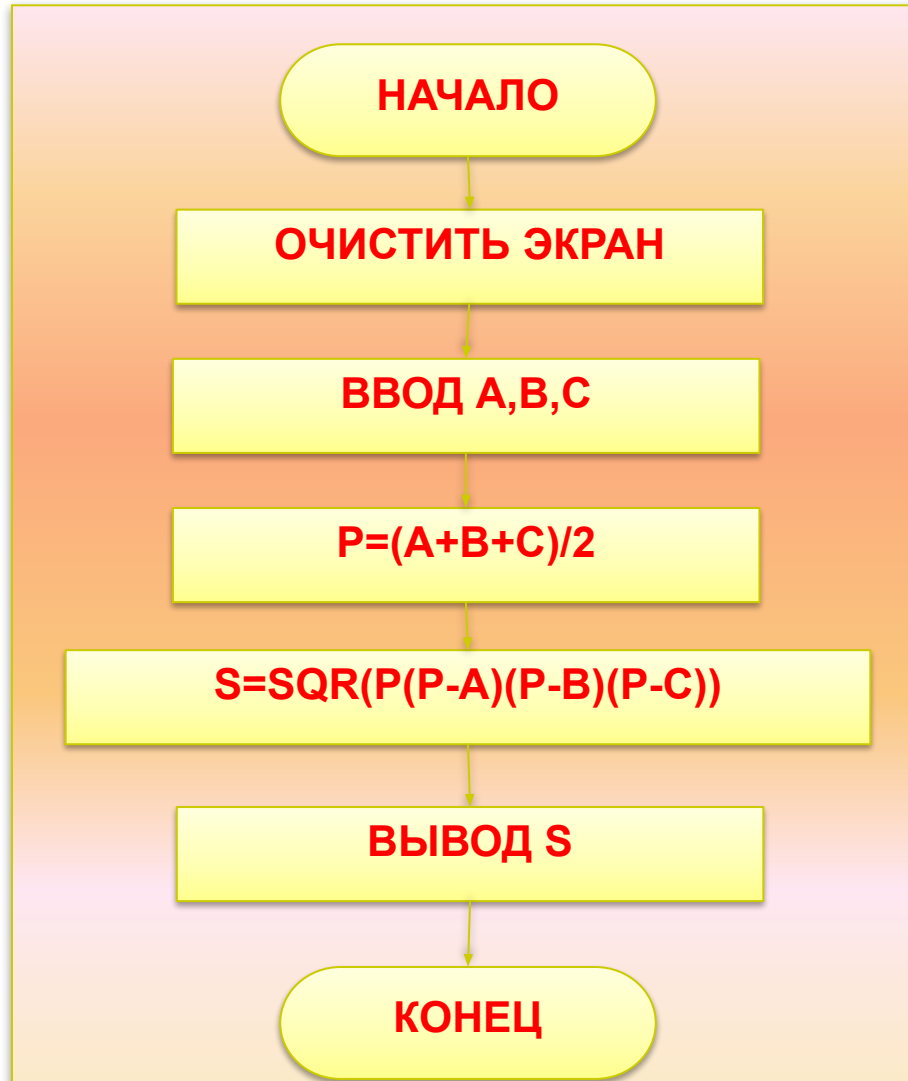
$$s = \sqrt{p(p-a)(p-b)(p-c)}$$

, где

$$p = \frac{1}{2}(a+b+c)$$



Составим алгоритм и запишем его в виде блок-схемы:



Программа может иметь вид: *вычисление площади треугольника по 3 сторонам.*

REM Вычисление *площади*
треугольника по 3 сторонам

Cls

INPUT "Введите a: ", a

INPUT "Введите b: ", b

INPUT "Введите c:",c

$p=(a+b+c)/2$

$s=\text{SQR}(p(p-a)(p-b)(p-c))$

PRINT " площадь треугольника: ", s

END

Задача 3. Программа для вычисления перемещения

```
REM ВЫЧИСЛЕНИЕ ПЕРЕМЕЩЕНИЯ  
CLS  
INPUT "ВВЕДИТЕ НАЧ. СКОРОСТЬ ", v0  
INPUT "ВВЕДИТЕ УСКОРЕНИЕ ", a  
INPUT "ВВЕДИТЕ ВРЕМЯ ДВИЖЕНИЯ ", t  
s=v0*t+(a*t^2)/2  
PRINT "ПЕРЕМЕЩЕНИЕ РАВНО: ",s  
END
```

Задача 4. Составить программу калькулятор. Ввод чисел А,В осуществить с клавиатуры. Найти сумму этих чисел; разность; произведение; частное двух чисел А, В; среднеарифметическое; среднегеометрическое. Вывести все результаты арифметических операций над ними с пояснениями.

```
10 Cls
```

```
20 rem калькулятор
```

```
30 Input"введите значения a,b ";a,b
```

```
40 print" сумма чисел a,b="; a+b
```

```
50 print" произведение чисел a,b="; a*b
```

```
60 print" разность чисел a,b="; a -b
```

```
70 print" частное чисел a,b="; a/b
```

```
80 print" среднеарифметическое чисел a,b="; (a+b)/2
```

```
90 print" среднегеометрическое чисел a,b="; sqr(a*b)
```

```
100 end
```

```
F5
```



Решение задач

- № 1 Напишите программу из двух операторов (не считая CLS), в результате работы которой на экране появилось бы сообщение:
Здравствуйте, <имя>.
Имя должно быть введено с клавиатуры во время выполнения предыдущего оператора.
- № 2. Напишите программу запрашивающую ваш год рождения, год рождения вашей мамы и печатающую, во сколько лет она вас родила.
- № 3. Напишите программу, запрашивающую 3 стороны треугольника и вычисляющие площадь по формуле Герона.

Проверь себя

1. **CLS**
INPUT «Ведите имя.»; A\$
PRINT «Здравствуйте, »; A\$
2. **CLS**
INPUT «Ведите год рождения мамы и свой год рождения через запятую»; A, B
L=2010 - A
M=2010 – B
K=L – M
PRINT K
3. **CLS**
INPUT «Ведите длины сторон треугольника»; A, B, C
P=(A+B+C)/2
S=SQR(P*(P-A)*(P-B)*(P-C))
PRINT S

Домашнее задание



- 1. Выучить состав языка программирования.**
- 2. Знать типы переменных и их обозначения.**
- 3. Программу нахождения периметра треугольника уметь воспроизводить и пояснять.**

Самостоятельно

Задача 1. Напишите программы, которые в зависимости от введенного числа либо вычисляют функцию, либо выдают сообщение, что функция не определена:

а) $y = \frac{b}{x}$

б) $y = \sqrt{x^2 - 1}$

Задача 2. Составьте блок – схему и программу для нахождения площади параллелограмма, если известны: длины его смежных сторон a, b и угол α между ними. Значения длин сторон и угла ввести с поясняющим текстом .