

Типы данных Java

Целые типы

Основные характеристики целых типов

1. Объем памяти
2. Диапазон возможных значений

Тип	Объем памяти	Диапазон	
		Степень 2	Число
byte	1 байт	$-2^7 \dots + 2^7 - 1$	-128 ... +127
short	2 байта	$-2^{15} \dots + 2^{15} - 1$	-32 768 ... +32 767
int	4 байта	$-2^{31} \dots + 2^{31} - 1$	-2 147 483 648 ... +2 147 483 647
long	8 байт	$-2^{63} \dots + 2^{63} - 1$	-9 223 372 036 854 775 808... +9 223 372 036 854 775 807

Вещественные типы

Основные характеристики вещественных типов

1. Объем памяти
2. Точность

Тип	Объем памяти
float (вещественное число малой точности)	4 байта
double (вещественное число повышенной точности)	8 байт

Другие типы

boolean - логический тип.

char - символьный тип.

void - пустой тип, используется только для указания, что метод не возвращает никакого значения.

Кроме простых типов, в Java поддерживаются объектные типы: Integer, Long, Float, Double, Boolean, Character.

Примеры:

```
byte b1;  
short s1,s2=0;  
int i=34;  
float f=-2.5;
```

Преобразование и приведение типов

Преобразование возможно между переменными родственных типов. Допустимо выполнять преобразование целых типов, которое не приведет к потере данных:

byte -> short -> int -> long

Преобразование выполняется приведением:
(новый тип) переменная старого типа

Пример:

```
double x = 9.997;
```

```
int x1 = (int) x;    => 9 // отбрасывается дробная часть
```

```
int x2 = (int)round( x);    => 10 // перед приведением выполняется округление
```

Константы

Для обозначения констант используется слово *final*. Присвоить значение константам можно только при инициализации и изменить его нельзя:

```
final int maxLen = 8;
```

Комментарии

В тексте программы можно использовать *комментарии*:

1) многострочные

```
/*
```

```
многострочный
```

```
комментарий
```

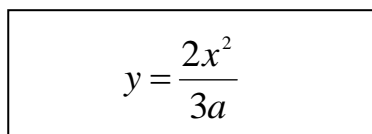
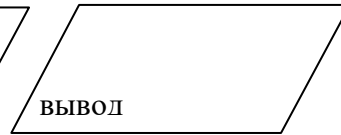
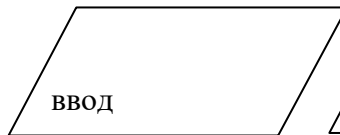
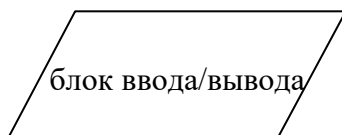
```
*/
```

2) однострочные

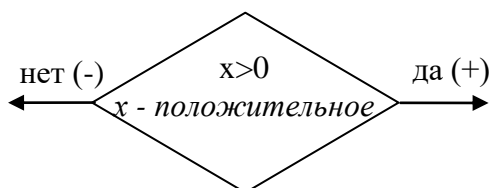
```
//комментарий
```

Блок-схема

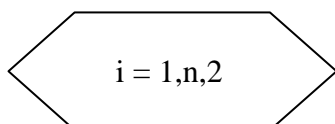
Блок-схема – это графическое изображение алгоритма в виде связанных блоков.



- вычислительный блок



- логический (условный) блок



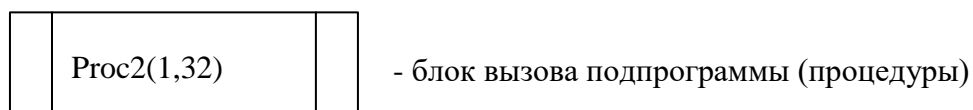
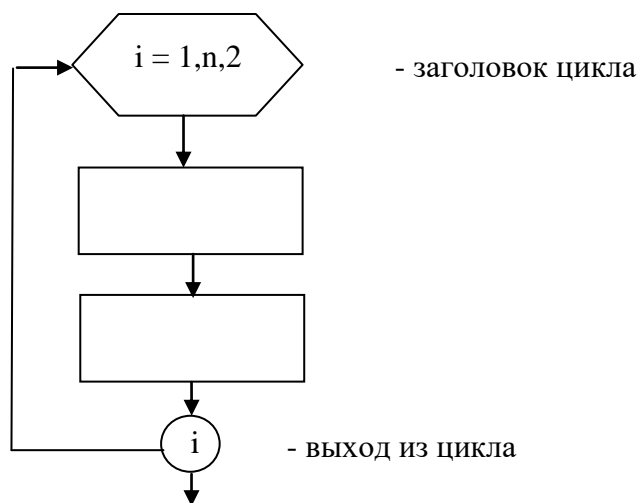
- блок для организации цикла

Примечание к блоку организации цикла

1 – начальное значение

n – конечное значение

2 – шаг изменения параметра (необязательно: если не указан, то шаг = 1)



Примечание к блоку вызова подпрограммы

Proc2 – имя подпрограммы

1 и 32 – фактические параметры

Пример:

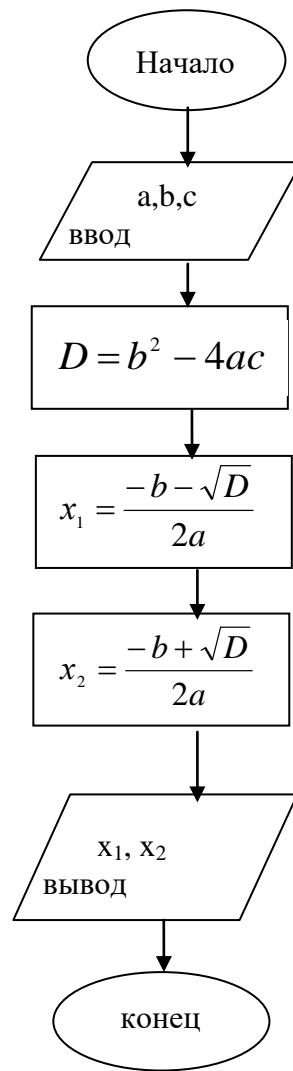
Решение квадратичного уравнения.

$$ax^2 + bx + c = 0$$

$$D = b^2 - 4ac$$

$$x_{1;2} = \frac{-b \pm \sqrt{D}}{2a}$$

Сначала рассмотрим простой вариант блок-схемы:



Теперь рассмотрим более корректный алгоритм, добавив условия, чтобы учесть ситуации, когда корень только один или корней нет вообще.

