

ОСНОВЫ АЛГОРИТМИЗАЦИИ

8 класс

Повторени

е

Алгоритм - это предназначенное для конкретного исполнителя описание последовательности действий, приводящих от исходных данных к требуемому результату, которое обладает свойствами:

- **дискретности**
- **понятности**
- **определённости**
- **результативности**
- **массовости**

Основные способы записи алгоритма

Словесные

Словесное описание

Построчная запись

Графические

Последовательность рисунков

Структурограмма

Блок-схема

На алгоритмических языках

Школьный алгоритмический язык

Язык программирования

Блок-схемы

В **блок-схеме** предписания изображаются с помощью различных геометрических фигур, а последовательность выполнения шагов указывается с помощью линий.



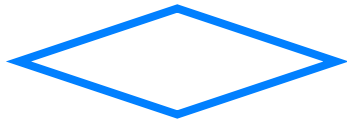
Блок начала или конца алгоритма



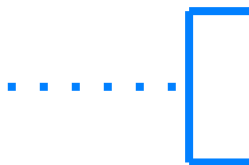
Блок ввода или вывода данных



Блок обработки данных



Блок проверки условия



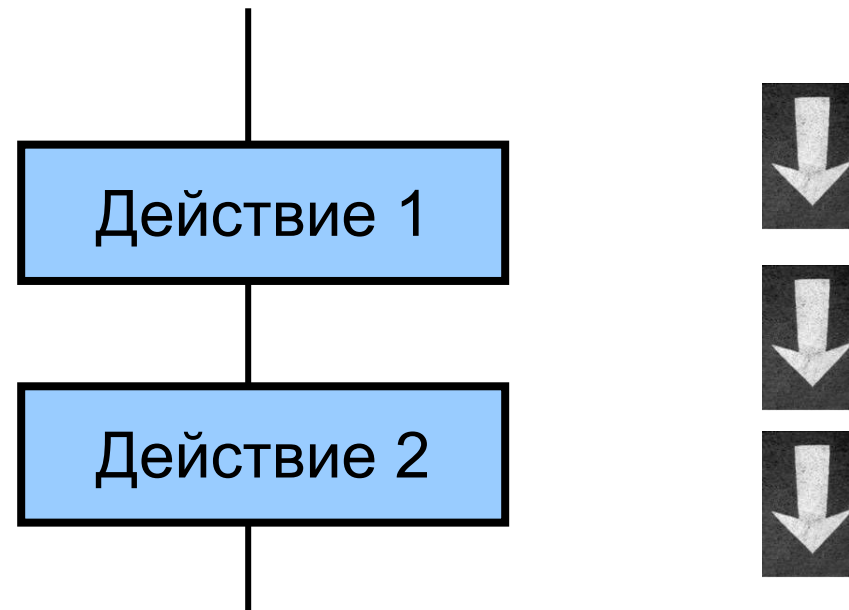
Блок пояснительных записей

Алгоритмы описывают последовательность действий над некоторыми ***информационными объектами***.

Величина в информатике – это отдельный информационный объект.

ОСНОВНЫЕ АЛГОРИТМИЧЕСКИЕ КОНСТРУКЦИИ **Следование**

Следование - алгоритмическая конструкция, отображающая естественный, последовательный порядок действий.
Алгоритмы, в которых используется только структура «следование», называются **линейными алгоритмами**.



Алгоритмическая структура «следование»

ОСНОВНЫЕ АЛГОРИТМИЧЕСКИЕ КОНСТРУКЦИИ

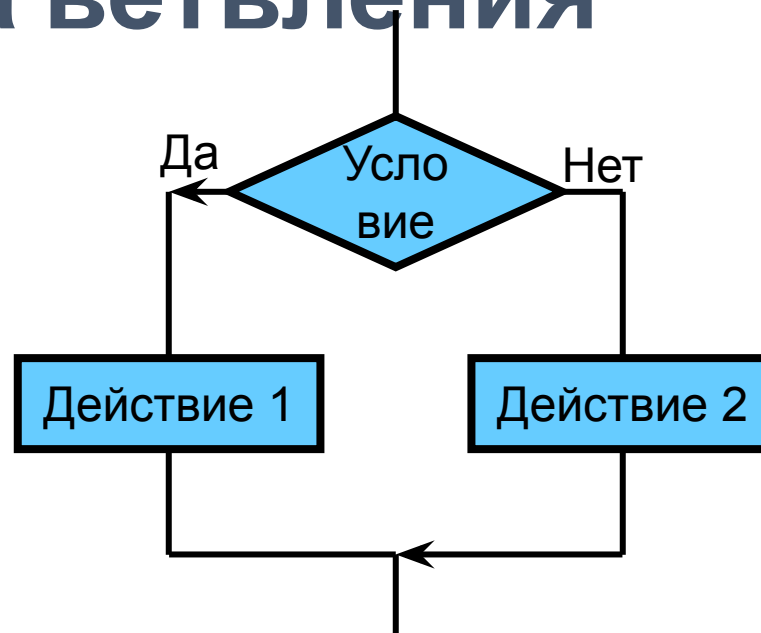
Ветвление

Ветвление - алгоритмическая конструкция, в которой в зависимости от результата проверки условия («да» или «нет») предусмотрен выбор одной из двух последовательностей действий (ветвей).

Алгоритмы, в основе которых лежит структура «ветвление», называют **разветвляющимися**.

Полная форма ветвления

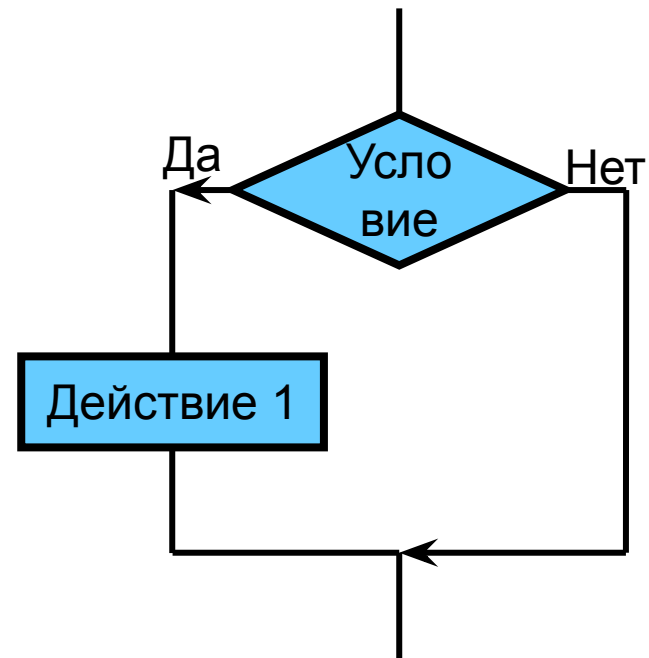
если <условие>
то <действие 1>
иначе <действие 2>
все



ОСНОВНЫЕ АЛГОРИТМИЧЕСКИЕ КОНСТРУКЦИИ

Неполная форма ветвления

если <условие>
то <действие 1>
все



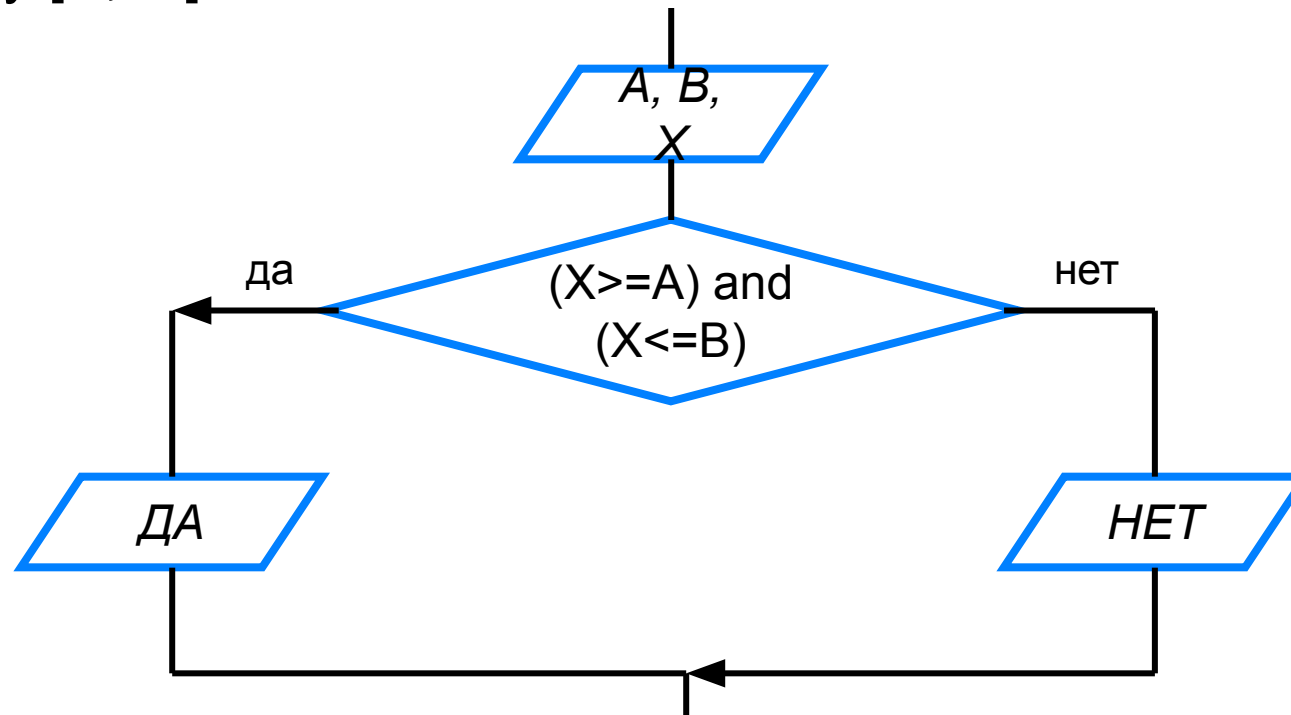
ОСНОВНЫЕ АЛГОРИТМИЧЕСКИЕ КОНСТРУКЦИИ

Простые и составные условия

Простые условия состоят из одной операции сравнения.

Составные условия получаются из простых с помощью логических связок *and* (**и**), *or* (**или**), *not* (**не**).

Пример. Алгоритм определения принадлежности точки X отрезку $[A; B]$.



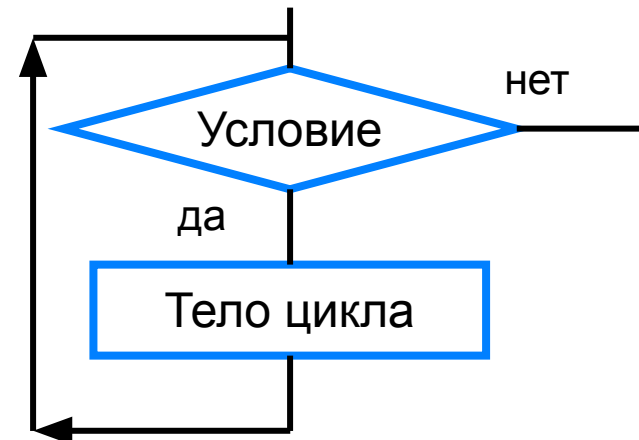
ОСНОВНЫЕ АЛГОРИТМИЧЕСКИЕ КОНСТРУКЦИИ

Цикл с заданным условием продолжения
работы
(цикл-ПОКА, цикл с предусловием)

нц пока <условие>

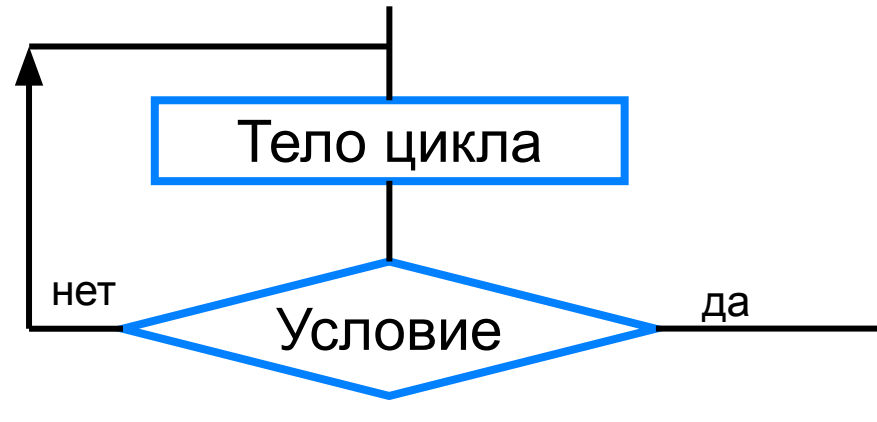
<тело цикла (последовательность действий)>

кц



ОСНОВНЫЕ АЛГОРИТМИЧЕСКИЕ КОНСТРУКЦИИ

Цикл с заданным условием окончания
работы
(цикл-ДО, цикл с постусловием)



Запись на алгоритмическом языке:

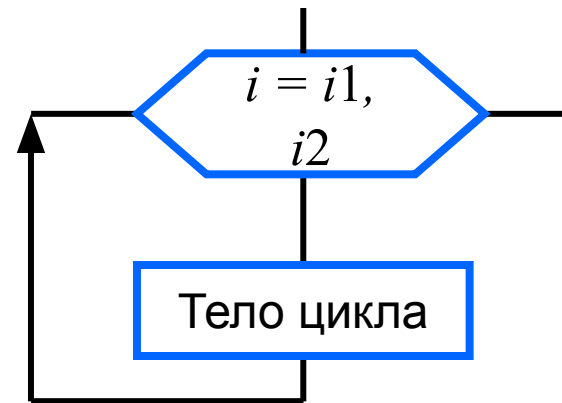
нц

<тело_цикла (последовательность действий)>

кц при <условие>

ОСНОВНЫЕ АЛГОРИТМИЧЕСКИЕ КОНСТРУКЦИИ

Цикл с заданным числом повторений
(цикл-ДЛЯ, цикл с параметром)



Запись на алгоритмическом языке:

```
нц для  $i$  от  $i1$  до  $i2$  шаг  $R$   
  <тело_цикла (последовательность действий)>  
кц
```

Языки программирования.

8 класс

Алфавит и словарь языка

Имена (констант, переменных, программ и других объектов) - любые отличные от служебных слов последовательности букв, цифр и символа подчеркивания, начинающиеся с буквы или символа подчеркивания.

Правильные имена

x

velichina

zzz

polnaja_summa

tri_plus_dva

s25

_k1

a1b88qq

oshibka

Неправильные имена

Ж - буква не латинского алфавита
polnaja summa - содержится символ (пробел), не являющийся буквой, цифрой или знаком подчеркивания.

2as - начинается с цифры

Domby&Son - содержится символ &, не являющийся буквой, цифрой или знаком подчеркивания

Прописные и строчные буквы в именах не различаются.
Длина имени может быть любой.

Простые типы данных

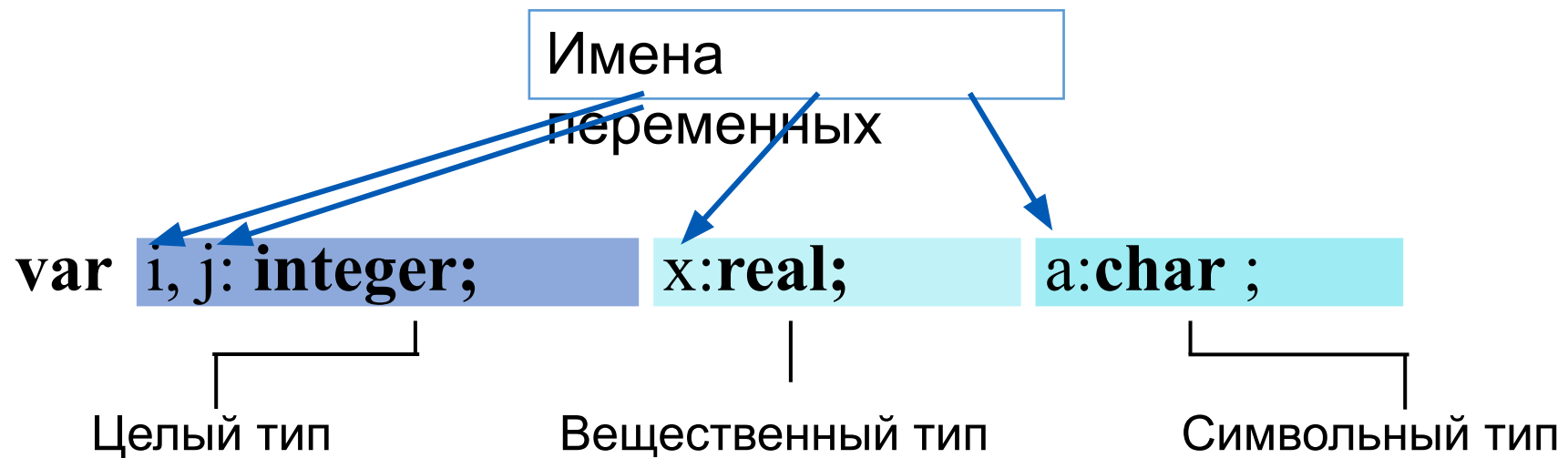
Название	Обозначение	Допустимые значения	Область памяти
Целочисленный	integer	- 32 768 ... 32 767	2 байта со знаком
Вещественный	real	$\pm(2.9 * 10^{-39} \dots 1.7 * 10^{+38})$	6 байтов
Символьный	char	Произвольный символ алфавита	1 байт
Строковый	string	Последовательность символов длиной меньше 255	1 байт на символ
Логический	boolean	True и False	1 байт



Структура программы на языке Паскаль



Раздел описания переменных



Общий вид программы

```
program <имя программы>;  
  const <список постоянных значений>;  
  var <описание используемых переменных>;  
begin <начало программного блока>  
  <оператор 1>;  
  <оператор 2>;  
  . . .  
  <оператор n>  
end.
```

Операторы - языковые конструкции для записи действия, выполняемого над данными в процессе решения задачи.

Оператор присваивания

Основное преобразование данных, выполняемое компьютером, - присваивание переменной нового значения, что означает изменение содержимого области памяти.

Общий вид оператора:

<имя переменной>:=<выражение>

Команда присваивания



Файл "SWF"