

Алгоритм – точная и понятная инструкция (набор команд) исполнителю, определяющая переход от исходных данных к результату.

Исполнитель – это человек или механическое устройство со строго определённым набором возможных действий (операций)

Исполнители: человек, робот, ЭВМ.

Команда - указание выполнить конкретное действие.

Система команд - совокупность всех команд, которые могут быть выполнены некоторым исполнителем.

Свойства алгоритма:

Дискретность. (Алгоритм должен состоять из отдельных шагов).

Понятность. (Указания должны быть понятны исполнителю).

Однозначность. (Единственность толкования правил выполнения действий и порядка их выполнения).

Массовость. (С помощью алгоритма можно решать не одну конкретную задачу, а множество однотипных задач и делать это неоднократно).

Результативность. (Выполнение алгоритма должно приводить к конкретному результату - решению задачи - за конечное число шагов).

Конечность. (Завершение работы алгоритма в целом за конечное число шагов).

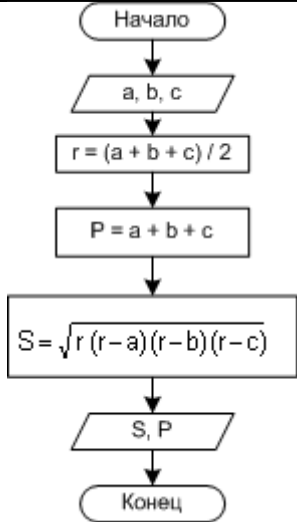
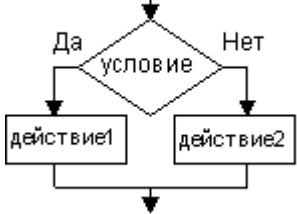
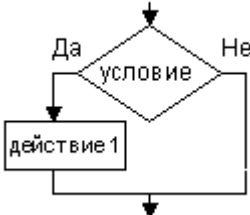
Способы описания алгоритмов

Словесно-пошаговый (на естественном языке).

Графический (на языке блок-схем).

Алгоритмический язык программирования.

Типы алгоритмов

Тип	Графический способ описания алгоритма	Запись примера на алгоритмическом языке (Паскаль)
<p>Линейный – это такой алгоритм, в котором все команды выполняются строго последовательно друг за другом.</p>	 <pre> graph TD Start([Начало]) --> Input[/a, b, c/] Input --> CalcR[r = (a + b + c) / 2] CalcR --> CalcP[P = a + b + c] CalcP --> CalcS[S = √r(r-a)(r-b)(r-c)] CalcS --> Output[/S, P/] Output --> End([Конец]) </pre>	<pre> var a, b, c, P: integer; r, S: integer; begin readln(a, b, c); r := (a + b + c) / 2; P := a + b + c; S := sqrt(r * (r - a) * (r - b) * (r - c)); writeln(S, P); end. </pre>
<p>Разветвляющийся – это такой алгоритм, который содержит команду ветвления.</p> <p>Команда ветвления - это составная команда, в которой та или иная серия команд выполняется после проверки условия.</p> <p>Команда ветвления имеет <i>полную</i> или <i>сокращенную</i> (неполную) форму.</p>	<p>полная форма</p>  <p>неполная форма</p> 	<p>полная форма:</p> <pre> if <условие> then begin <Действие 1>; end else begin <Действие 2>; end; </pre> <p>неполная форма:</p> <pre> if <условие> then begin <Действие 1>; end; </pre>

Тип	Графический способ описания алгоритма	Запись примера на алгоритмическом языке (Паскаль)
<p>Циклический - это такой алгоритм, который содержит команду повторения.</p> <p>Команда повторения—это составная команда, в которой тело цикла выполняется несколько раз.</p>	<pre> graph TD Start([Начало]) --> Input[/Ввод n/] Input --> P1[p = 1] P1 --> Decision{i = 1, n} Decision --> LoopBody[p = p * i] LoopBody --> Decision Decision --> Output[/Вывод p/] Output --> End([Конец]) </pre>	<pre> var i, p, n: integer; begin readln(n); p:=1; for i:=1 to n do begin p:=p*i; end; writeln(p); end. </pre>

Массив – упорядоченное множество однотипных элементов, объединенных общим именем и занимающим в компьютере определенную область памяти. *Моделью массива является таблица.*

Характеристики массива: 1. Имя (формируется по правилам переменных на языке программирования Паскаль). 2. Размер (количество элементов массива). 3. Размерность (количество индексов).

Этапы решения задач с использованием массива	
Название этапа	Запись на алгоритмическом языке (Паскаль)
<p>I. Объявление массива</p>	<pre> var a:array[1..8] of integer; b:array[1..10] of real; </pre>
<p>II. Заполнение массива вывод значений элементов на экран.</p> <p>Способы: с клавиатуры, случайными числами, с помощью формулы</p>	<p>с клавиатуры:</p> <pre> ... begin for i:=1 to 8 do begin readln(a[i]); write(a[i], ' '); end; writeln; ... </pre> <p>случайными числами (например, от 1 до 20): ВМЕСТО <code>readln(a[i]);</code> указать <code>a[i]:=random(20)+1;</code></p> <p>с помощью формулы (заполним массив четными натуральными числами): ВМЕСТО <code>readln(a[i]);</code> указать <code>a[i]:=i*2;</code></p>
<p>III. Обработка массива</p> <p>Задачи анализа – задачи, в которых необходимо найти сумму элементов массива, количество элементов согласно какому-либо условию.</p> <p>Задачи поиска – поиск максимального или минимального элемента массива, а так же их порядковый номер</p>	<pre> ... //Найти сумму элементов массива, кратных трём s:=0; for i:=1 to 8 do if a[i] mod 3=0 then s:=s+a[i]; //Найти минимальный элемент массива и его номер min:=a[1];kmin:=1; for i:=2 to 8 do if a[i]<min then begin min:=a[i]; kmin:=i; end; ... </pre>
<p>IV. Вывод результатов</p>	<pre> ... writeln('Сумма элементов массива, кратных трем равна ',s); writeln('Минимальный элемент массива равен ',min); writeln('Расположен на ',kmin,' месте'); end. </pre>