

Свойства алгоритма

Алгоритм – это последовательность действий, строгое исполнение которых приводит к решению поставленной задачи за конечное число шагов.

- **Результативность и дискретность**
Алгоритм должен обеспечивать преобразование объекта из начального состояния в конечное состояние за определенное число дискретных шагов.
- **Массовость**
Один и тот же алгоритм может применяться к большому количеству однотипных объектов.
- **Детерминированность**
Исполнитель должен выполнять команды алгоритма в строго определенной последовательности.
- **Выполнимость и понятность команд**
Алгоритм должен содержать команды, входящие в систему команд исполнителя и записанные на понятном для исполнителя языке.

Исполнитель алгоритма

Исполнитель – это человек, компьютер, робот, другое механическое, электронное или абстрактное устройство, умеющее выполнять определенный набор команд.

Набор команд исполнителя называется системой команд исполнителя (СКИ).

Формы записи алгоритма:

- словесная;**
- графическая;**
- на языке программирования.**

Словесный способ

Алгоритм, представленный в виде последовательности действий и записанный на естественном языке.

Пример.

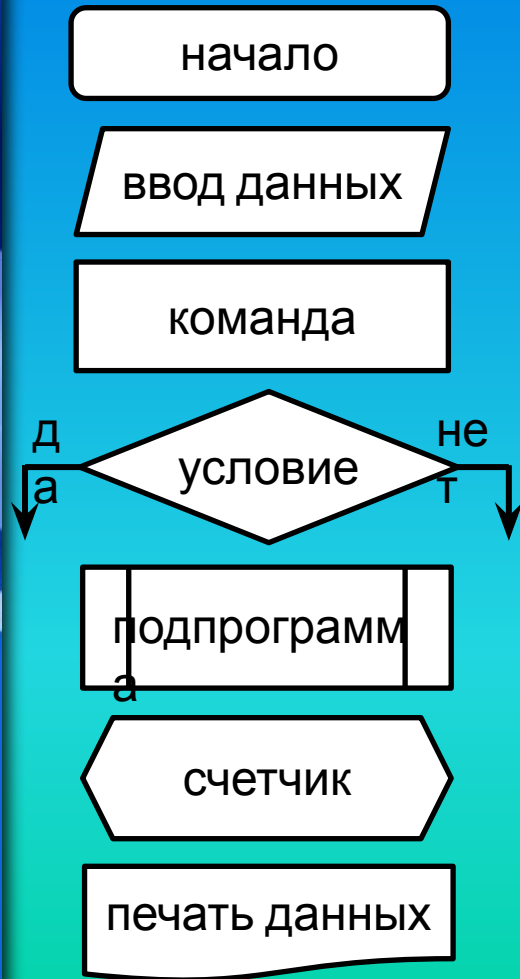
1. Прийти на остановку
2. Дождаться автобуса
3. Зайти в автобус
4. Рассчитаться
5. Выйти на нужной остановке

Недостатки:

- Строго не формален (записан не по строгим правилам)
- Многословен
- Допускает неоднозначность толкования отдельных предписаний

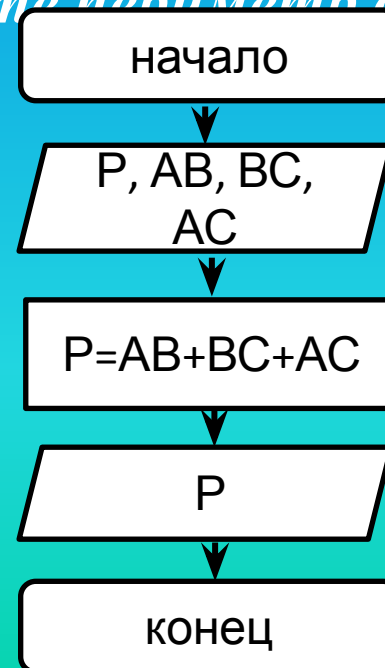
Графический способ

Элементы алгоритма изображаются на блок-схеме с помощью различных геометрических фигур, внутри которых записывается программный код.



Задача.

*Даны длины трёх сторон
треугольника: $AB=3$, $BC=4$, $AC=5$.
Вычислите периметр фигуры.*



Программный способ

Язык для записи алгоритмов должен быть формализован (т.е. по строгим правилам). Такой язык принято называть языком программирования, а алгоритм, записанный на языке программирования, называется программой.

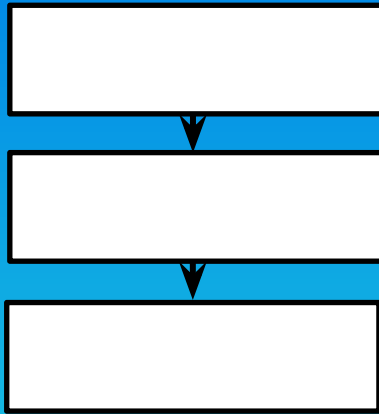
```
program a1;  
var P, a, b, c: integer;  
begin  
write('a='); read(a);  
write('b='); read(b);  
write('c='); read(c);  
P:= a+b+c  
writeln('P=',P)  
end.
```

Задача.

*Даны длины трёх сторон
треугольника: $AB=3$, $BC=4$, $AC=5$.
Вычислите периметр фигуры.*

Линейный алгоритм:

Алгоритмы, в которых все действия выполняются последовательно, одно за другим.



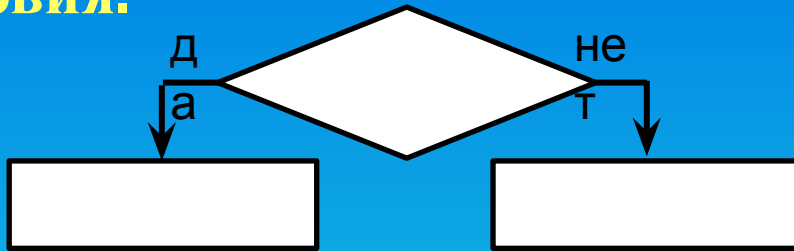
Пример.

1. Установить деталь в измерительное устройство.
2. Измерить диаметр детали.
3. Вывести на печать параметр
4. Конец.



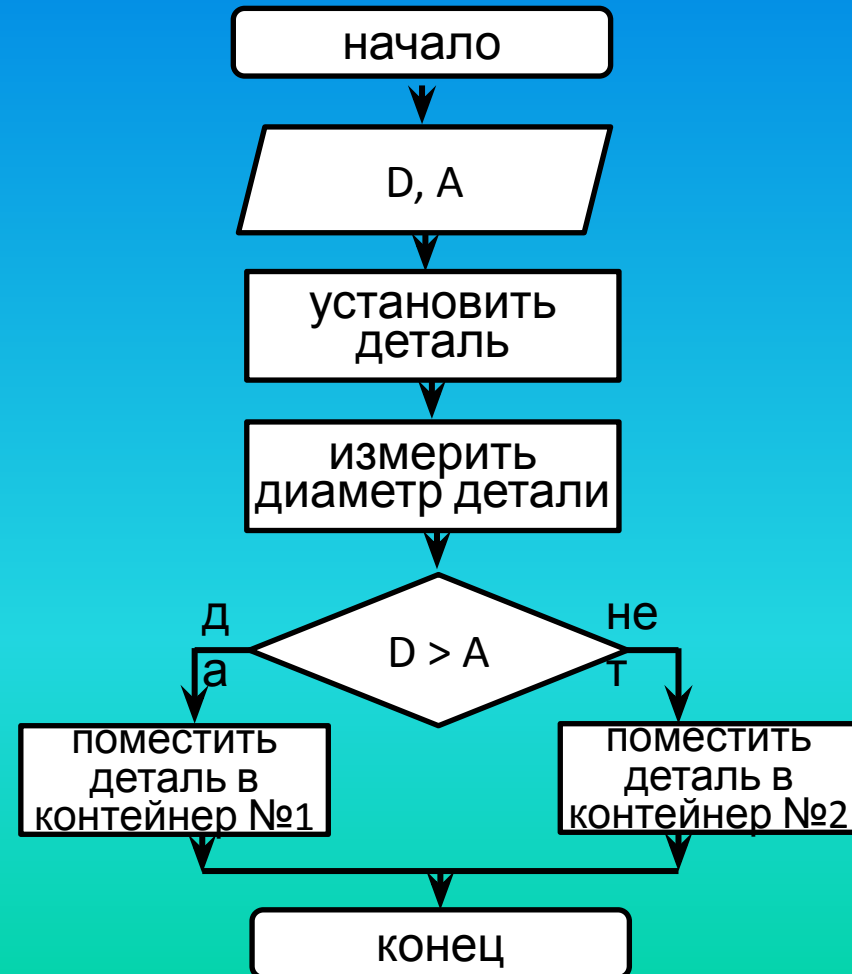
Алгоритм с ветвлением:

Алгоритмы, в которых производится одного из нескольких вариантов действий в зависимости от выполнения некоторого условия.



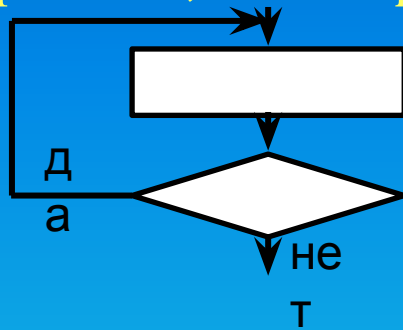
Пример.

1. Установить деталь в измерительное устройство.
2. Измерить диаметр детали.
3. Если диаметр больше заданного, то поместить деталь в контейнер №1.
4. Иначе поместить деталь в контейнер №2.
5. Конец.



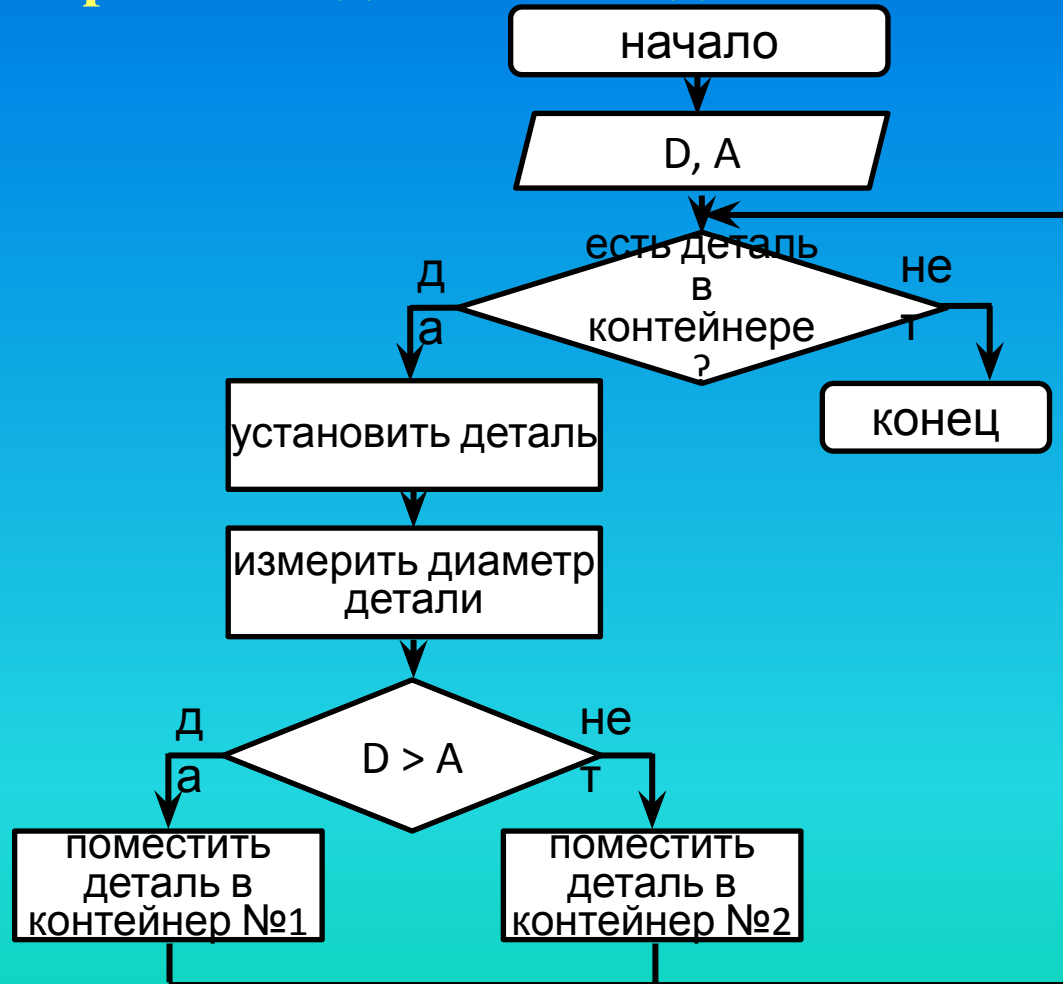
Циклический алгоритм:

Алгоритмы, в которых повторяются одни и те же действия.



Пример.

1. Пока в контейнере есть детали:
2. Установить деталь в измерительное устройство.
3. Измерить диаметр детали.
4. Если диаметр больше заданного, то поместить деталь в контейнер №1.
5. Иначе поместить деталь в контейнер №2.



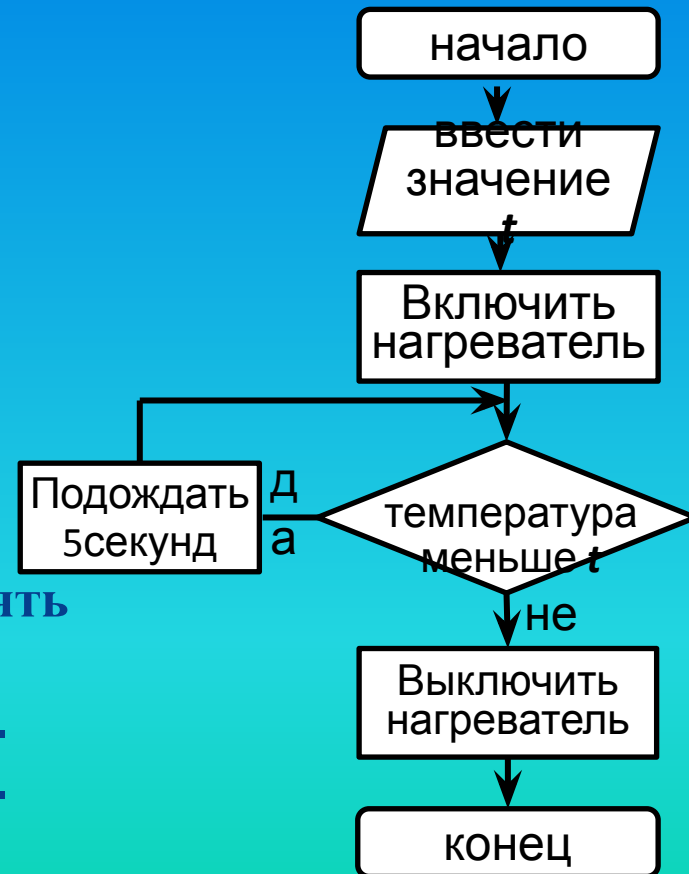
Вспомогательный алгоритм:

Алгоритмы, целиком и многократно используемые в составе других алгоритмов.

Пример.

«Нагрев до t »

1. Ввести значение t .
2. Включить нагреватель.
3. Пока температура меньше t , выполнять действия:
4. Подождать 5 сек.
5. Измерить температуру.
6. Конец цикла.
7. ~~Включить нагреватель.~~



1. Пока счетчик меньше 10 выполнять действия:
2. Налить в резервуар жидкость №1.
3. Налить в резервуар жидкость №2.
4. Перемешать жидкости.
5. Нагрев до 150°C .
6. Открыть заслонку

Вспомогательный алгоритм:

Алгоритмы, целиком и многократно используемые в составе других алгоритмов.

Пример.

«Нагрев до t »

1. Ввести значение t .
2. Включить нагреватель.
3. Пока температура меньше t , выполнять действия:
4. Подождать 5 сек.
5. Измерить температуру.
6. Конец цикла.
7. *«Нагрев жидкости»* нагреватель.

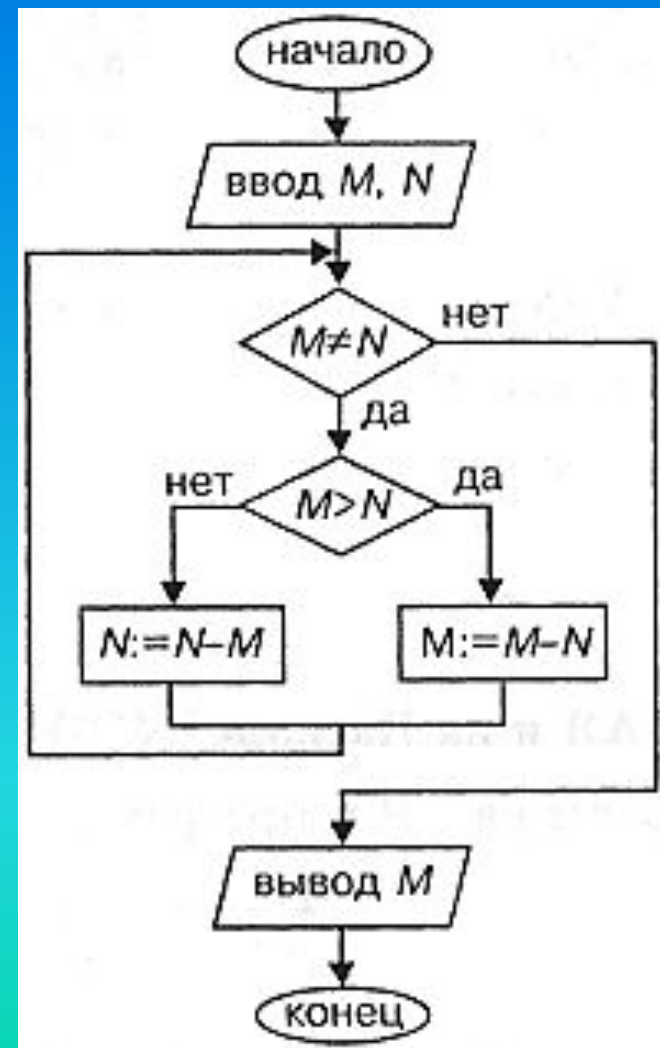
1. Пока счетчик меньше 10 выполнять действия:
2. Налить в резервуар жидкость №1.
3. Налить в резервуар жидкость №2.
4. Перемешать жидкости.
5. Нагрев до 150°C .
6. Открыть заслонку



Алгоритм нахождения НОД (наибольший общий делитель)

Задача.

1. Составить таблицу проверки условий и определить какое значение M будет выведено на экран.
2. Входные данные:
 $M=108, N=48$

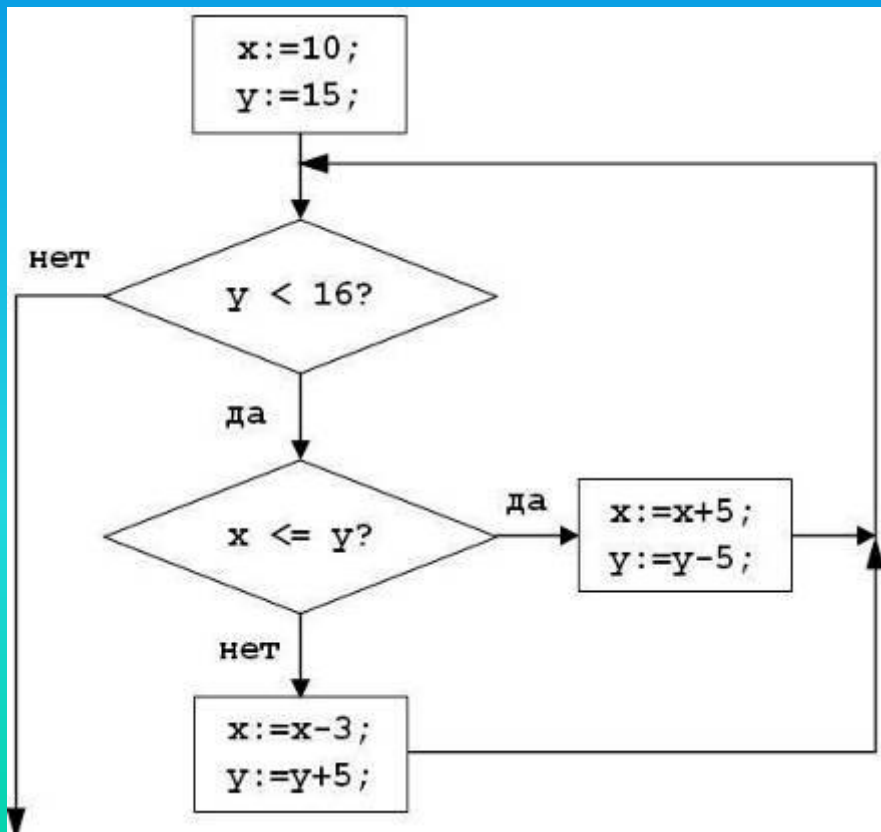


Самостоятельная работа

Определите значение переменных X и Y после выполнения фрагмента алгоритма.

1 вариант

X=13; Y=20



2 вариант

X=30; Y=20

