

---

# ***А Л Г О Р И Т М Ы***



**Алгоритм** - это любая точно определенная последовательность действий (не обязательно математических), необходимых для выполнения некоторой работы или для решения задачи.

Алгоритм, записанный на языке программирования, называется **программой**.

При хорошо составленных алгоритмах можно действовать формально или автоматически, не вникая в смысл, что и делает ЭВМ.

# СВОЙСТВА АЛГОРИТМОВ

- **Дискретность.** Для реализации алгоритма на ПК необходима возможность разбиения задачи на элементарные операции для их выполнения. Алгоритм состоит из отдельных указаний, которые называются *командами*. Команды выполняются одна за другой без пропусков. В алгоритме отражаются не только арифметические действия но и логические связи. Таковую структуру алгоритма оказывается *прерывной*.
- **Определенность.** Строго определенный, понятный алгоритм к достижению однозначного решения, доступный ПК.
- **Массовость.** Алгоритм многократно повторяется при различных исходных данных.
- **Результативность, эффективность.** Алгоритм должен оканчиваться нужным результатом и приносить эффективность. В алгоритме должно быть предусмотрено окончание его работы.

# СРЕДСТВА ЗАПИСИ АЛГОРИТМОВ

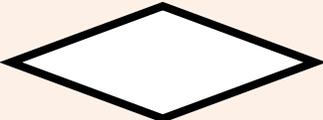
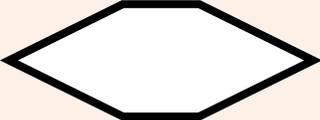
---

- **На естественном языке**
- **Структурная схема алгоритма (блок-схема)**

Структурная схема алгоритма представляет собой графическое изображение последовательности действий при реализации данного алгоритма.

Этапы решения задачи представляются в структурной схеме отдельными блоками, которые изображаются соответствующими символами. Внутри символов структурной схемы указывается содержание соответствующих этапов вычислений. Направление выполнения алгоритма обозначается стрелками – линиями потока.

# ЭЛЕМЕНТЫ БЛОК-СХЕМ

Наименование	Обозначение	Пояснение
Процесс		Вычислительные действия или последовательность вычислительных действий
Решение		Проверка условий
Ввод-вывод		Ввод-вывод данных
Модификация		Начало цикла
Соединитель		Разрыв линий потока
Линии потока		Если поток направлен вниз или направо можно стрелку не ставить
Пуск, останов		Начало, конец, останов, вход и выход в подпрограммах

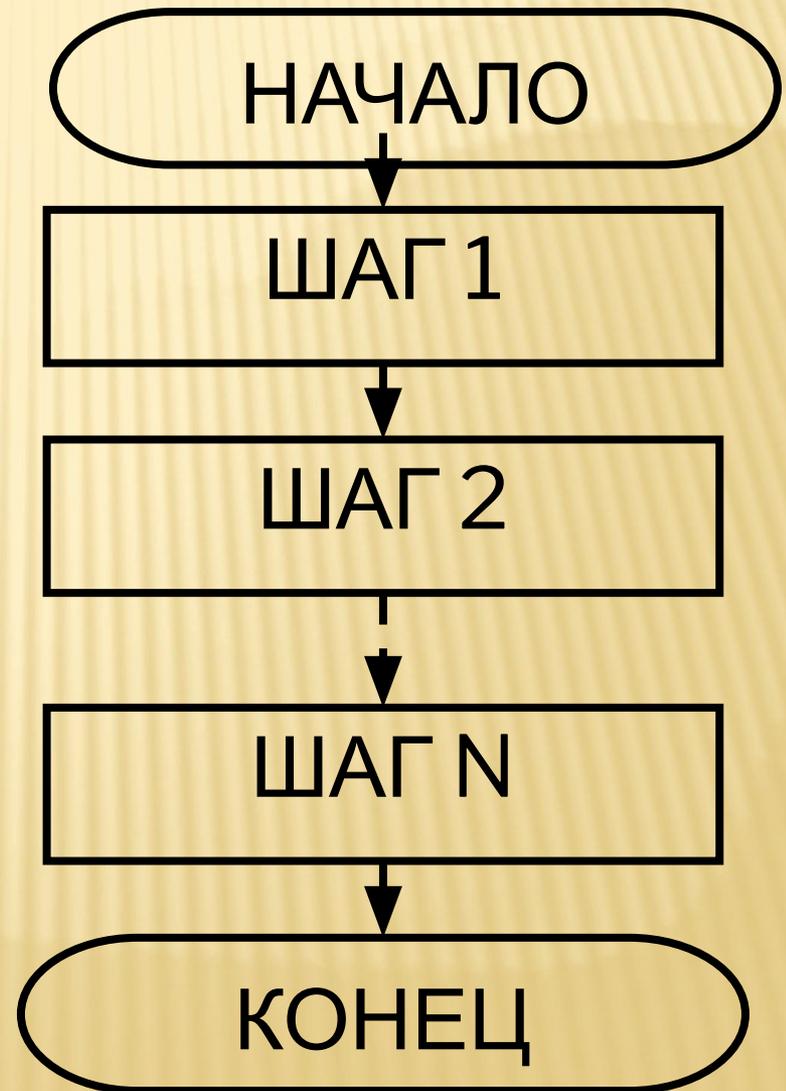
# ПРАВИЛА ОФОРМЛЕНИЯ БЛОК-СХЕМ

---

- Каждый блок имеет единственную точку входа, кроме блока пуска, который не имеет входа.
- Каждый безусловный блок имеет единственную точку выхода, кроме блока останова, который не имеет ни одной точки выхода.
- Условный блок имеет 2 выхода. Выход условного блока необходимо пометить условиями (да, нет;  $>0$ ,  $=0$ ,  $<0$ ,  $+ -$  ;  $0, 1$ ).
- Линии, идущие на вход некоторого блока могут соединяться. Это соответствует переходу на конкретный единственный этап вычислений после нескольких других этапов.
- Линия, исходящая из выходной точки блока не может разветвляться на несколько направлений. Этим исключается неоднозначность перехода между блоками.

# ЛИНЕЙНЫЙ АЛГОРИТМ

Этот алгоритм состоит из простой последовательности действий, которые выполняются только один раз в порядке их следования.



# УСЛОВНЫЙ (РАЗВЕТВЛЯЮЩИЙСЯ) АЛГОРИТМ

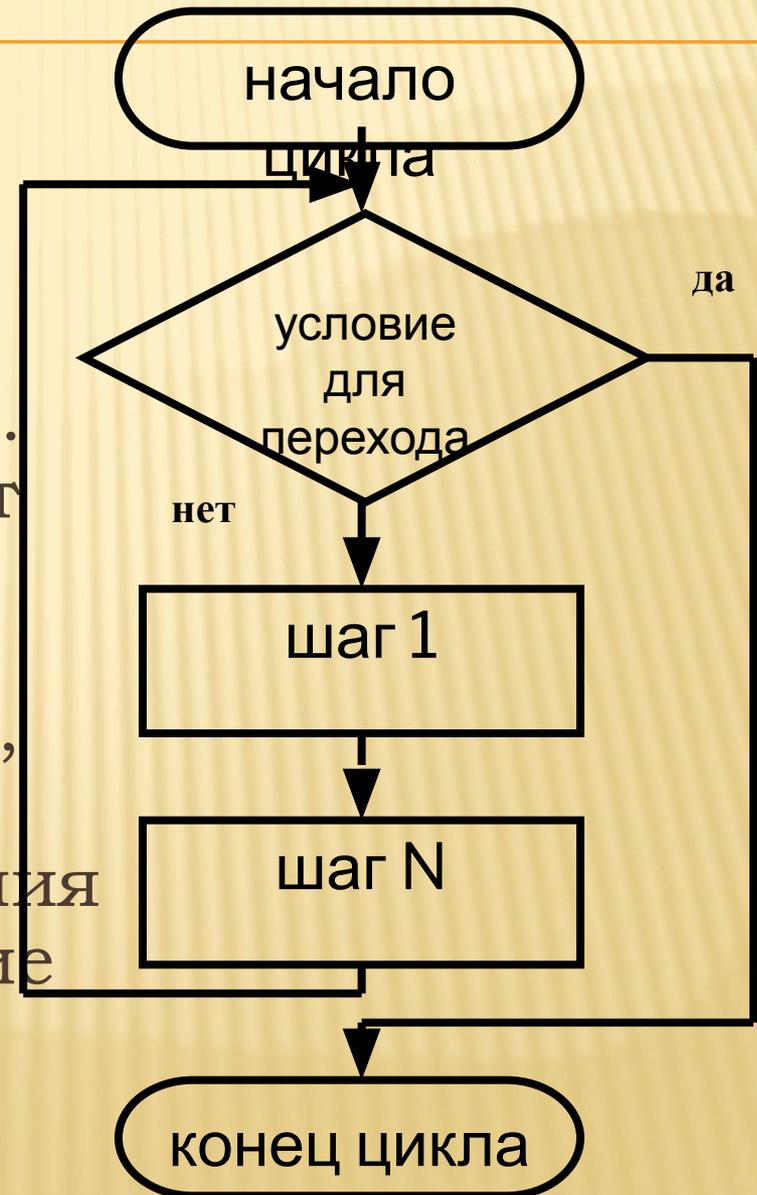
Содержит блок проверки условия.

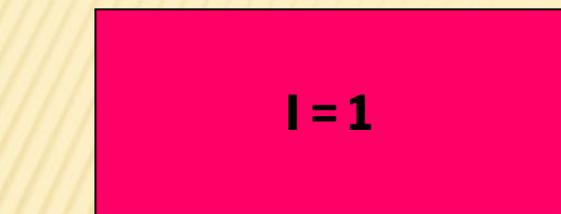
В зависимости от результата проверки выполняется та или иная последовательность действий, называемая ветвью. Эта последовательность выполняется однократно.



# ЦИКЛИЧЕСКИЙ АЛГОРИТМ

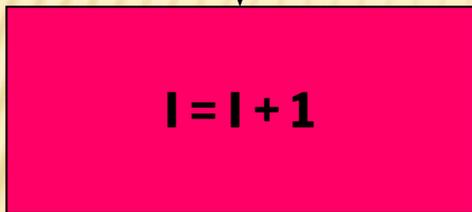
Содержит некоторую последовательность действий, выполнимых многократно. Такая структура содержит несколько типовых блоков. Основной блок производит требуемые вычисления. Остальные блоки имеют вспомогательное значение, они устанавливают начальные и новые значения данных, проверяют условие окончания циклического процесса.





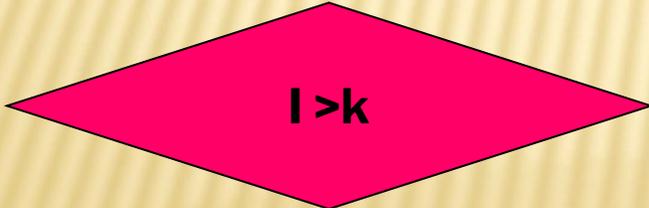
**$l=1$**

Счетчик. Ему присваивается начальное значение 1. (A – имя переменной)



**$l=l+1$**

Значение счетчика увеличивается на 1 после очередного выполнения команды



**$l>k$**

Проверяется сколько раз выполнилась команда. Нужно ее выполнить k раз