

# *Алгоритмы*

Алгоритмы, структуры алгоритмов,  
структурное программирование

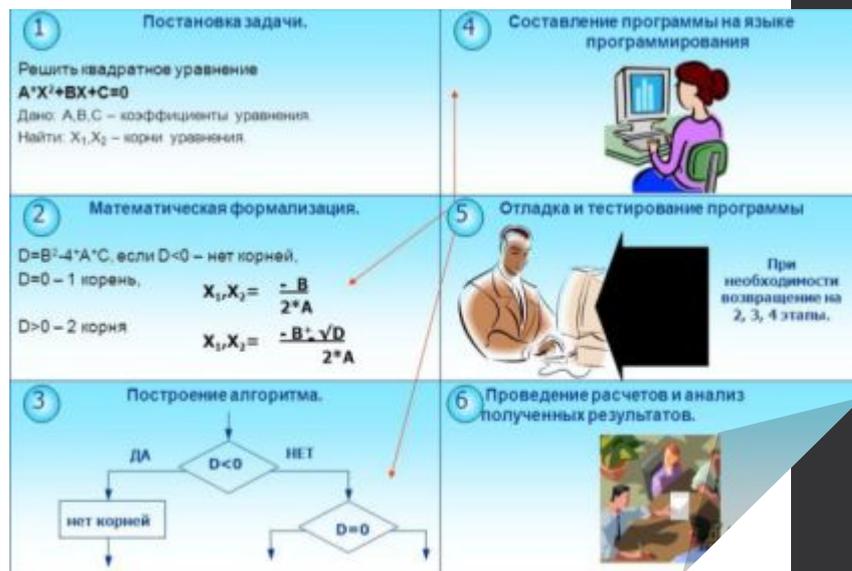
# Алгоритм

Алгоритм — это последовательность команд управления каким-либо исполнителем для достижения некоторого результата.



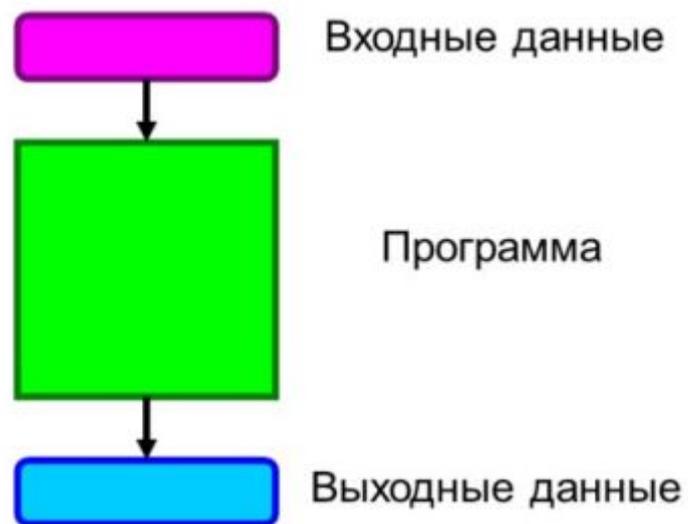
# Этапы решения задачи на компьютере

1. Постановка задачи.
2. Формализация задачи.
3. Построение алгоритма.
4. Составление программы на языке программирования.
5. Отладка и тестирование программы.
6. Проведение расчетов и анализ полученных результатов.



# Данные

- Совокупность величин, с которыми работает компьютер, принято называть данными.
- По отношению к программе данные делятся на исходные, результаты (окончательные данные) и промежуточные данные, которые получаются в процессе вычислений.
- Данные - это множество величин.



# ВЕЛИЧИНЫ

- Всякая величина занимает свое определенное место в памяти компьютера, а значение этой величины определяется двоичным кодом в этой ячейке.
- У всякой величины имеются три основных свойства: имя, значение и тип.
- Величины делятся на константы и переменные.



# Типы данных

- В каждом языке программирования существует своя концепция типов данных, своя система типов.
- Однако в любой язык входит минимально необходимый набор основных типов данных, к которому относятся целый, вещественный, логический и символьный типы.

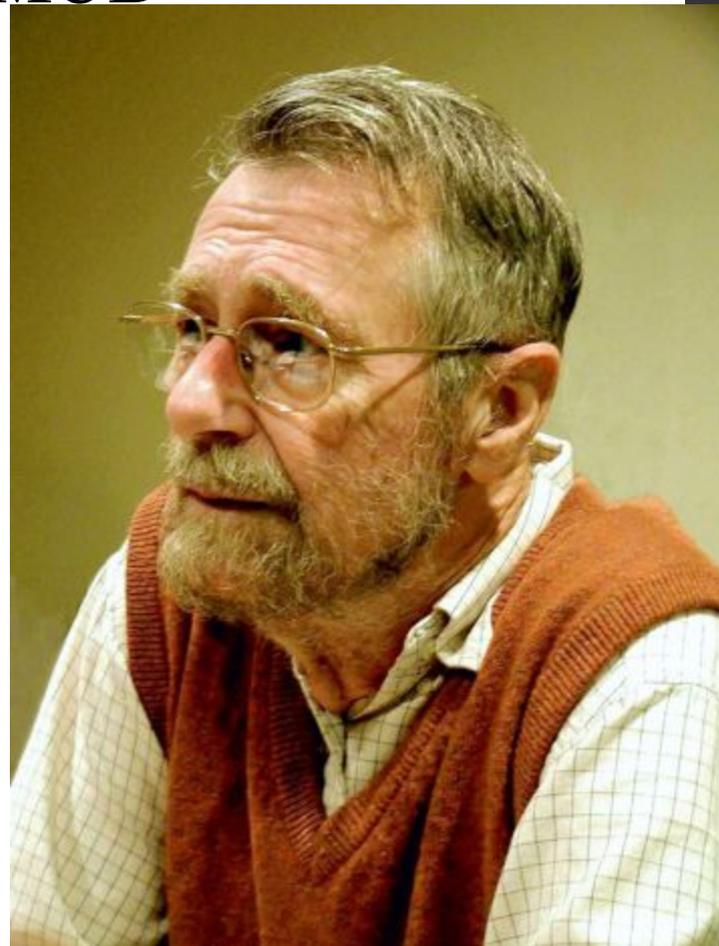


# Типы величин

Тип	Значения	Операции	Внутреннее представление
Целый	Целые положительные и отрицательные числа в некотором диапазоне: -32768...32768. Примеры: 23, -12, 387	Арифметические операции с целыми числами: +, -, *, целочисленное деление и остаток от деления. Операции отношений (<, >, = и др.)	Формат с фиксированной запятой. 2 байта со знаком.
Вещественный	Любые (целые и дробные) числа в некотором диапазоне: $\pm(2.9 * 10^{-39} \dots 1.7 * 10^{+38})$ Примеры: 2.5, -0.01, 45.0, $3.6 * 10^9$	Арифметические операции: +, -, *, /. Операции отношений	Формат с плавающей запятой. 6 байтов.
Логический	true (истина), false (ложь)	Логические операции: И (and), ИЛИ (or), НЕ (not). Операции отношений	1 бит: 1 — true; 0 — false. 1 байт.
Символьный	Любые символы компьютерного алфавита. Примеры: 'a', '5', 'Ч', '\$'	Операции отношений	Коды таблицы символьной кодировки. 1 символ — 1 байт.

# Структура алгоритмов

- В 1969 году известным голландским ученым-программистом Э. В. Дейкстрой было доказано, что алгоритм для решения любой логической задачи можно составить только из структур **следование, ветвление, цикл.**
- Их называют базовыми алгоритмическими структурами.
- Методика программирования, основанная на этой теореме, называется **структурным программированием.**



# Следование

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



# Следование

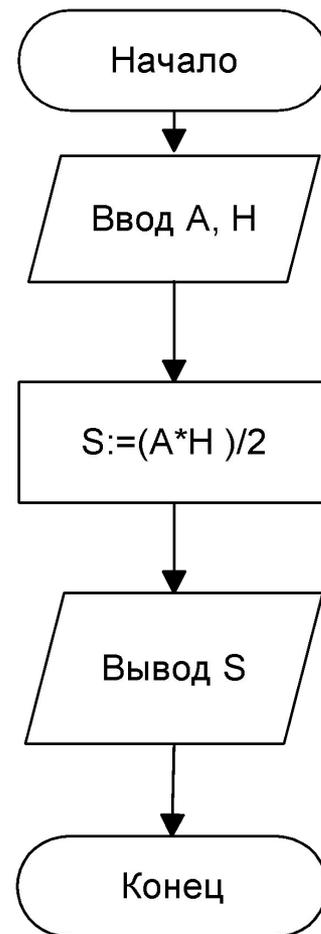
Задача №1. Найдите площадь  
треугольника с основанием  $A$ , высотой  
 $H$ .

алг

нач

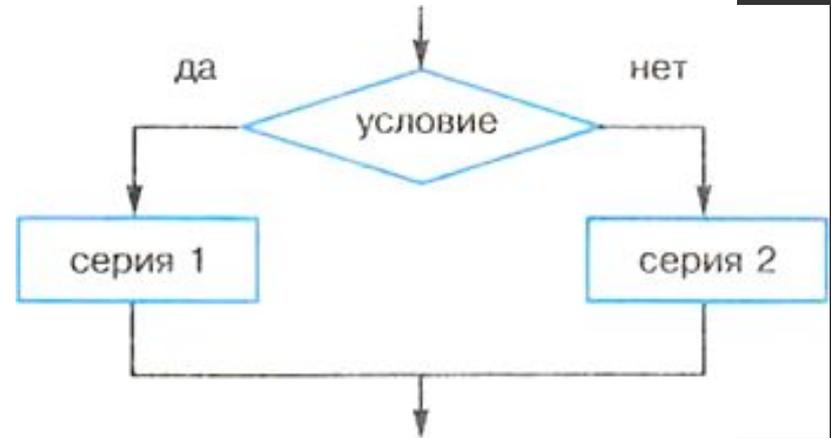
$S := (A * H) / 2$

кон



# Ветвление

- Ветвление — алгоритмическая альтернатива.
- Управление передаётся одному из двух блоков в зависимости от истинности или ложности условия.
- Затем происходит выход на общее продолжение.



# Ветвление

если <У1>

то если <У2>

то <С1>

все

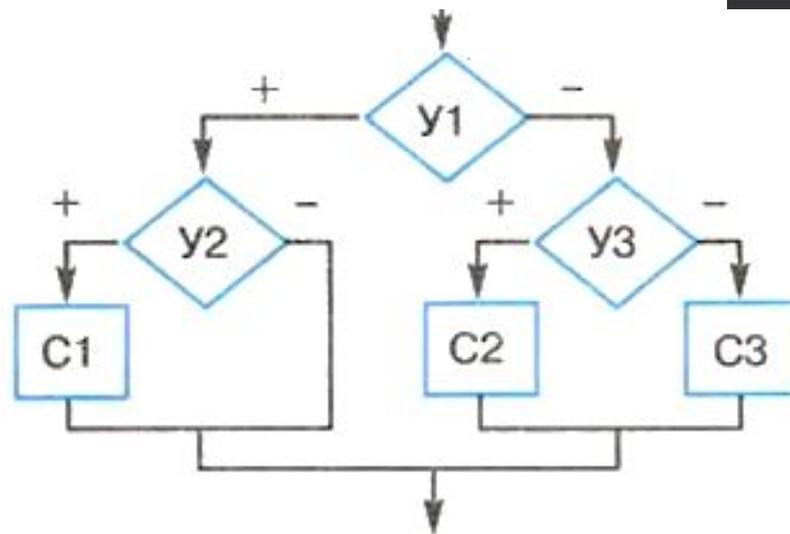
иначе если <У3>

то <С2>

иначе <С3>

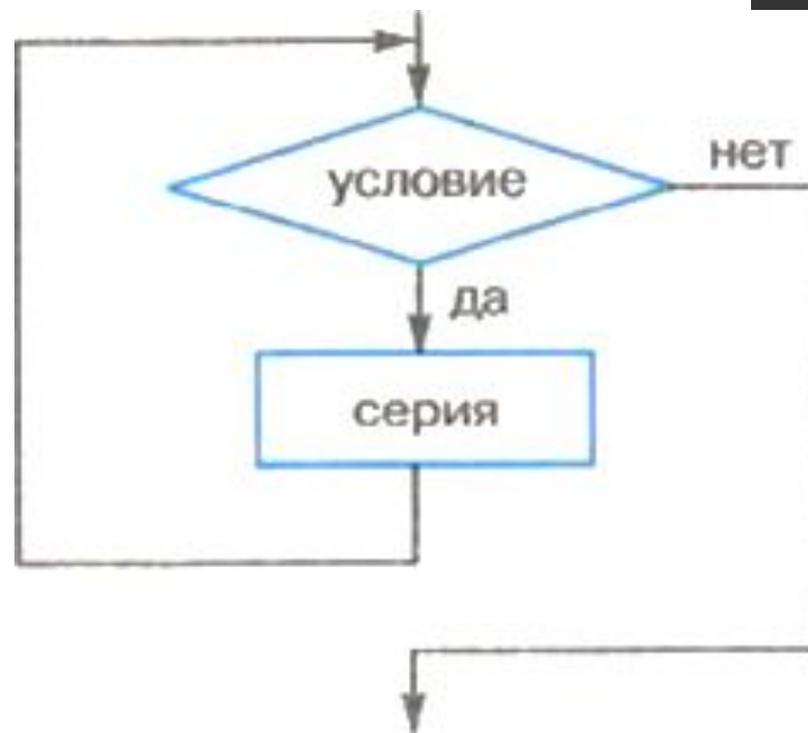
все

все



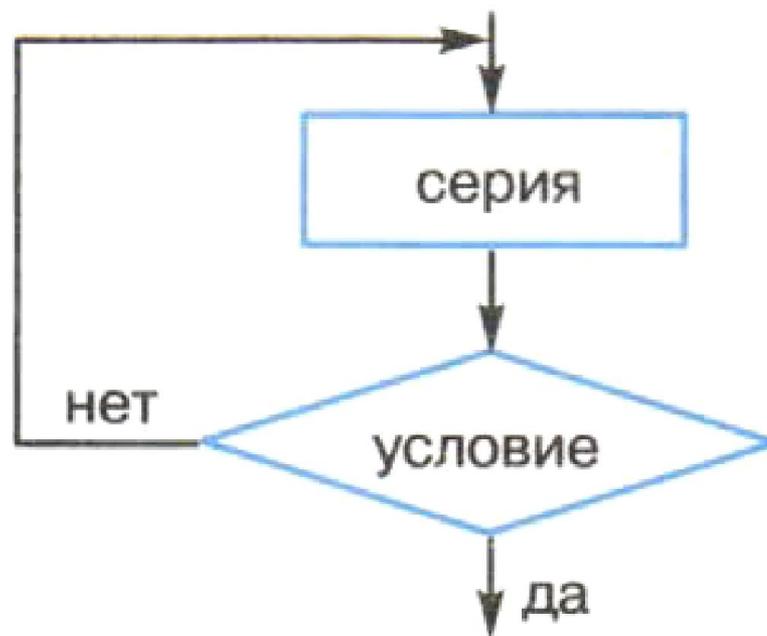
# ЦИКЛ

- Цикл — повторение некоторой группы действий по условию.
- Различают два типа цикла. Первый — цикл с предусловием: **цикл-пока**.
- Пока условие истинно, выполняется серия, образующая тело цикла.



# ЦИКЛ

- Второй тип циклической структуры — цикл с постусловием: **цикл-до**.
- Здесь тело цикла предшествует условию цикла. Тело цикла повторяет свое выполнение, если условие ложно.
- Повторение прекращается, когда условие становится истинным.



# ЦИКЛ

- Если блок, составляющий тело цикла, сам является циклической структурой, то имеют место вложенные циклы.
- Вложенная конструкция записывается смещенной по строке на несколько позиций вправо относительно внешней для нее конструкции

пока <У1>

нц

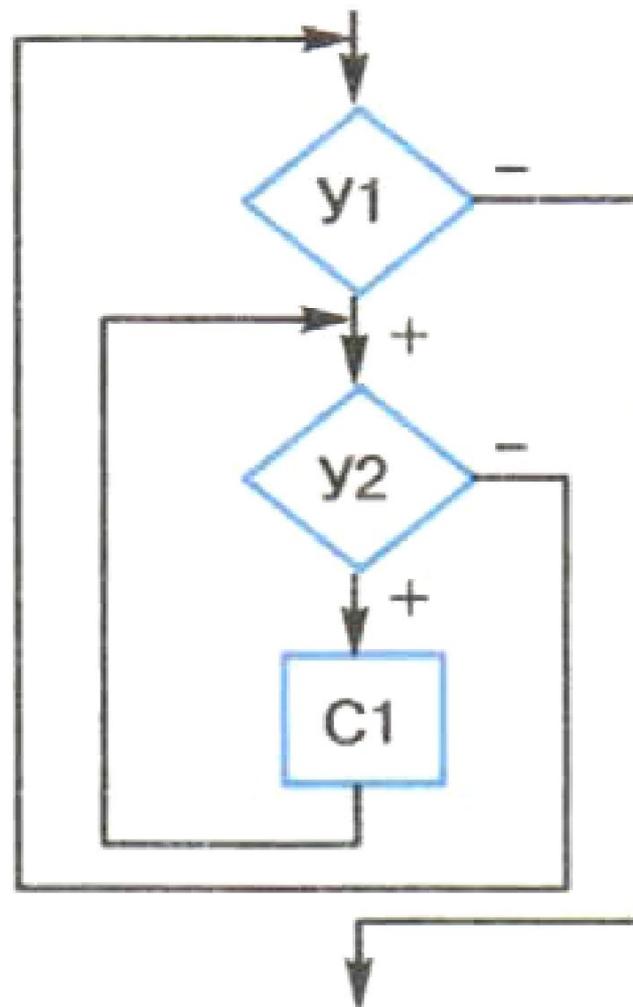
пока <У2>

нц

<С1>

кц

кц



# Структурное программирование

- **Структурное** программирование — это не только форма описания алгоритма и программы, но это еще и способ мышления программиста.
- Размышляя над алгоритмом, нужно стремиться составлять его из стандартных структур.
- Если использовать строительную аналогию, то структурная методика построения алгоритма подобна сборке здания из стандартных секций, в отличие от складывания по кирпичику.



Спасибо за  
внимание!