

Модель, алгоритм,  
программа

# Модель

- Это формализованное описание объекта, процесса или явления, выраженное математическими соотношениями, набором чисел и текстов, графиками, таблицами, словесными формулами и т. п.
- **Информационная модель** – это совокупность знаковой информации, характеризующая свойства и состояние объекта, процесса или явления.
- К информационным моделям можно отнести чертежи, схемы, графики, алгоритмы, математические соотношения (формулы, системы уравнений и т.п.)

# Алгоритм

- Это строгая последовательность четких действий, описывающая процесс преобразования объекта из начального состояния в конечное, записанная с помощью понятных исполнителю команд.
- Примеры использования алгоритмов человеком: рецепты, инструкции к техническим устройствам, алгоритмы игр (прятки, например).

# Свойства алгоритма

- **1. Дискретность** - процесс решения задачи по алгоритму разбит на отдельные элементарные операции.
- **2. Однозначность** - правила и порядок выполнения действий алгоритма имеют единственное толкование и исключают произвольность исполнения.
- **3. Результативность** - при выполнении алгоритма за конечное число шагов обязательно получается конечный результат.
- **4. Универсальность (массовость)** - применимость алгоритма к различным наборам исходных данных.
- **5. Понятность** – алгоритм должен содержать только те команды, которые понятны исполнителю.

# Программирование

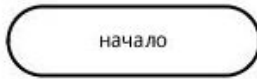
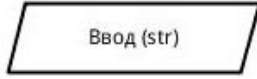
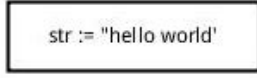
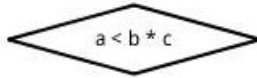
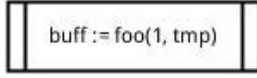
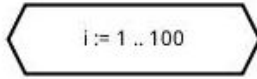
- **Программа** - это последовательность инструкций, предназначенных для выполнения компьютером. В настоящее время программы оформляются в виде текста, который записывается в файлы.
- **Программирование** – это теоретическая и практическая деятельность решения задачи средствами конкретного языка программирования и оформления полученных результатов в виде программы.
- **Язык программирования** – специально разработанный искусственный язык, предназначенный исключительно для записи алгоритмов, исполнение которых поручается ЭВМ.

# Трансляция

- Перевод текста программы в машинные двоичные коды.
- Программы **трансляторы** бывают двух типов:
  - **Интерпретаторы** транслируют текст программы по шагам и сразу же выполняют эти шаги, не создавая .exe-файла.
  - **Компиляторы** транслируют весь текст программы и создают отдельный, готовый к исполнению .exe-файл.

# Схема

- Это абстракция какого-либо процесса или системы, наглядно отображающая наиболее значимые части.
- **Блок-схема** представляет собой совокупность символов, соответствующих этапам работы алгоритма и соединяющих их линий.

|   |   |
|---|---|
|  <p>Терминатор начала и конца работы функции</p>  | <p>Терминатором начинается и заканчивается любая функция. Тип возвращаемого значения и аргументов функции обычно указывается в комментариях к блоку терминатора.</p>  |
|  <p>Операции ввода и вывода данных</p>           | <p>В ГОСТ определено множество символов ввода/вывода, например вывод на магнитные ленты, дисплеи и т.п. Если источник данных не принципиален, обычно используется символ параллелограмма. Подробности ввода/вывода могут быть указаны в комментариях.</p>   |
|  <p>Выполнение операций над данными</p>          | <p>В блоке операций обычно размещают одно или несколько (ГОСТ не запрещает) операций присваивания, не требующих вызова внешних функций.</p>   |
|  <p>Блок, иллюстрирующий ветвление алгоритма</p> | <p>Блок в виде ромба имеет один вход и несколько подписанных выходов. В случае, если блок имеет 2 выхода (соответствует оператору ветвления), на них подписывается результат сравнения — «да/нет». Если из блока выходит большее число линий (оператор выбора), внутри него записывается имя переменной, а на выходящих дугах — значения этой переменной.</p> |
|  <p>Вызов внешней процедуры</p>                | <p>Вызов внешних процедур и функций помещается в прямоугольник с дополнительными вертикальными линиями.</p>   |
|  <p>Подготовка данных</p>                      | <p>Символ «подготовка данных» в произвольной форме (в ГОСТ нет ни пояснений, ни примеров), задает входные значения. Используется обычно для задания циклов со счетчиком.</p>  |

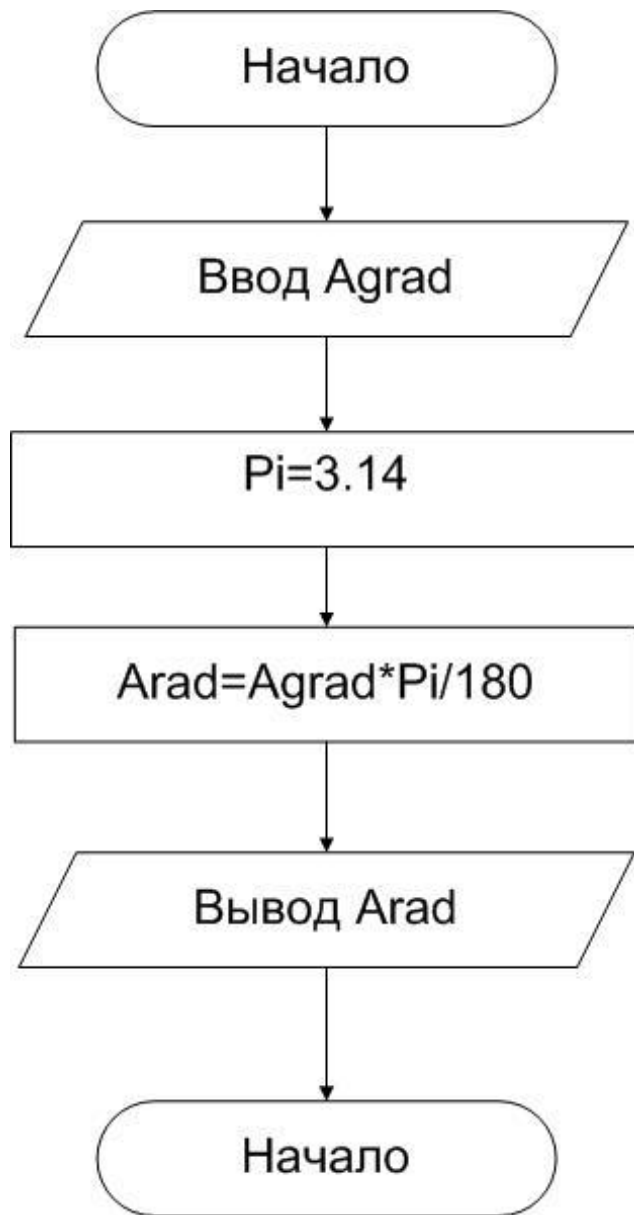


# Решение простых задач

- Определить, как решать задачу – математическая модель
- Выяснить исходные данные
- Если каких-либо данных не хватает, решить как их получить: случайным образом, с клавиатуры, из файла и т.д.
- Перечислить выходные данные
- Построить математическую модель решения
- Построить блок-схему
- Написать программу

# Написать программу для перевода угла в градусах в радианы

- Формула перевода:  $\text{УголРадиан} = \text{УголГрадусов} * \pi / 180$
- Исходные данные:  $\pi$  и значение угла в градусах.
- $\pi = 3.14$
- Угол в градусах можно запросить у пользователя.
- Выходные данные: значение угла в радианах
- Мат Модель:  $A_{\text{rad}} = A_{\text{grad}} * \pi / 180$



# Написать программу для перевода угла в радианах в градусы

- Модель:  $A_{\text{grad}} = A_{\text{rad}} * 180 / \text{Pi}$

Напишите программу, вычисляющую и выводящую на экран число бит в некотором количестве Кбайт.

- Модель:  $\text{Bit} = \text{Kbyte} * 1024 * 8$

Напишите программу, вычисляющую и выводящую на экран число Kbyte в некотором количестве бит.

- Модель:  $Kbyte = Bit / 1024 / 8$

Напишите программу, подсчитывающую сколько долларов и евро можно купить на некоторую сумму в рублях

- Исходные данные: курс евро, курс доллара, сумма в рублях
- Модель:  $K_{dol} = S_{rub} / Kurs_{dol}$

