

# Лекция № 7

## **Модели решения задач**

# Основные этапы решения задач на ЭВМ

Под решением задачи на ЭВМ подразумевается нечто большее, чем то, что делает собственно вычислительная система. Чтобы разобраться, что делает человек, что – ЭВМ, рассмотрим процесс подготовки и решения задачи на ЭВМ.

## Этапы подготовки и решения задач на ЭВМ:

1. постановка задачи,
2. математическая формулировка, разработка модели,
3. выбор (разработка) метода решения,
4. разработка алгоритма,
5. составление программы (кодирование) и подготовка данных,
6. тестирование и отладка,
7. выполнение программы,
8. обработка результатов.

Приведённое разделение является условным. Названы наиболее общие и необходимые этапы. Наличие того или иного этапа определяется решаемой задачей.

В настоящее время существует большое количество программных систем, которые позволяют решить задачу без программирования. Поэтому вместо этапа 5 (в некоторых случаях вместо этапов 4-6) решают задачу средствами готовой программной системы

# I этап. Постановка задачи

- **Первостепенное значение должно быть уделено выяснению конечной цели** и выработке общего подхода к исследуемой проблеме:
  - выяснение, существует ли решение задачи вообще и является ли оно единственным;
  - изучение общих свойств рассматриваемого физического явления или объекта;
  - изучение условий, при которых будет работать объект (система) и т.п.
- **На этом этапе требуется глубокое понимание существа поставленной задачи.** Правильно сформулировать задачу иногда не менее сложно, чем её решить.
- **Этот этап выполняет специалист в конкретной области знаний.** Иногда постановка задачи не вызывает затруднений, но в некоторых случаях это длительный процесс.
- **Пример.** Движение судна в воде (летательного аппарата в воздухе): следует рассмотреть перемещение по трём координатам в зависимости от упора, развиваемого движителями (винтами), скорости и в зависимости от сил сопротивления среды, которые зависят от очень многих факторов (загрузка

# II этап. Математическая формулировка (формализация) задачи

В результате анализа существа задачи определяются:

- . объём и специфика исходных данных,
- . вводятся условные обозначения,
- . **строится модель рассматриваемого объекта или явления,**
- . выбирается математический аппарат.

При этом надо уметь на языке математики сформулировать конкретные задачи физики, механики, медицины, технологии и т.д. То есть, нужны знания из предметной области и хорошие знания математики.

В ряде случаев задачу стремятся упростить, то есть учесть только те факторы и параметры, которые наиболее существенно влияют на конечный результат. Это возможно только при хорошем представлении сути явления и владении аппаратом, позволяющим строить адекватные модели.

# II этап. Математическая формулировка (формализация) задачи

В самых простых случаях этап математической формулировки (формализации) сводится к следующим действиям:

- . вводу обозначений,
- . классификации данных на входные (исходные), выходные (результаты), промежуточные (вспомогательные),
- . записи математических зависимостей и ограничений.

# III этап. Выбор (разработка) метода решения

Разработкой новых методов занимаются только в тех случаях, когда существующие по каким-либо причинам не могут быть применены или отсутствуют. Это крайне редкая ситуация.

**Для большинства задач методы решения разработаны и надо только выбрать такой метод, который отвечает поставленным требованиям.**

При выборе метода решения учитывают:

- . сложность формул и соотношений, лежащих в основе метода,
- . требуемую точность получения результата,
- . объём вычислений,
- . требования к размерам памяти,
- . время решения задачи и т.п.

Во многих случаях для успешной реализации этого этапа необходимы знания в области вычислительной математики (численные методы решения). При этом следует помнить, что при использовании численных методов любой, получаемый на ЭВМ результат, является приближённым.

# IV этап. Разработка алгоритма

Понятие алгоритма является одним из центральных в переработке информации. Термин «алгоритм» является синонимом слов «способ», «рецепт».

**Алгоритм** – это описанный со всеми подробностями способ получения удовлетворяющих поставленным условиям результатов.

В алгоритме всё должно быть определено заранее, так как программа, реализующая алгоритм, выполняет только то, что предусмотрено в алгоритме.

Исполнитель выполняет только то, что ему предписано.

Не всякую последовательность действий можно считать алгоритмом.

# IV этап. Разработка алгоритма

**Действия, выполняемые над данными можно считать алгоритмом, если они обладают свойствами дискретности, детерминированности, массовости и результативности.**

**Дискретность** означает, что алгоритм состоит из последовательности шагов.

**Детерминированность (определённость)** означает, что каждый пункт алгоритма должен быть сформулирован так, чтобы действия при его реализации определялись однозначно: если применить один и тот же алгоритм к одним и тем же данным, то результат будет одинаков.

**Массовость** подразумевает, что алгоритм должен быть пригоден для решения других задач этого же типа, а не только для одной конкретной задачи.

**Результативность** означает, что применение алгоритма должно приводить к получению искомого результата после конечного числа шагов.

# IV этап. Разработка алгоритма

**Алгоритмы бывают численные и нечисленные.**

**Примеры численных алгоритмов:**

- определение корней квадратного уравнения,
- разложение числа на простые множители.

**Примеры нечисленных алгоритмов:**

- преобразование текста в код Морзе,
- деление угла пополам с помощью циркуля и линейки,
- упорядочение слов по алфавиту.

Разработка алгоритмов ведётся с древних времён. Придуманы способы получения площадей фигур и объёма тел, разработаны приёмы вычислений в разных системах счисления.

# IV этап. Разработка алгоритма

**Способов описания алгоритмов достаточно много.** Наиболее часто используются:

- словесное описание,
- структурные схемы (блок-схемы),
- псевдокод.

**Словесное описание.** Применение естественного языка вполне допустимо.

При кажущейся простоте и доступности применение этого способа на практике весьма проблематично. Причина в том, что сложно сформулировать действия так, чтобы не возникало разночтений. В разговорных языках много слов, имеющих разные значения, смысл фраз часто зависит от контекста. Тем не менее, словесное описание алгоритмов решения задач иногда используется.

**При словесном описании действия нумеруют и стараются использовать чёткие формулировки.**

# IV этап. Разработка алгоритма

**Псевдокод** занимает промежуточное положение между естественным языком и языком программирования.

**Псевдокод позволяет формально изображать логику программы, не заботясь о синтаксических тонкостях конкретного языка программирования.**

Псевдокод не ограничен формальными синтаксическими правилами, предписывается только употребление управляющих структур и соблюдение правил ступенчатой записи (для облегчения чтения и понимания алгоритма).

Иногда для наглядности управляющие структуры выделяют шрифтом или подчёркиванием.

Запись алгоритма на псевдокоде достаточно хорошо понимается людьми, не занимающимися программированием.

Никаких стандартов в написании псевдокода нет, но связь с языками программирования существует, так как используются принятые в процедурных языках программирования управляющие структуры: ЕСЛИ–ТО–ИНАЧЕ, ЦИКЛ–ПОКА, ЦИКЛ–ДО, ВЫБОР (IF–THEN–ELSE, DO–REPEAT,

# IV этап. Разработка алгоритма

## Структурные схемы алгоритмов

- **Правила построения структурных схем алгоритмов** и применяемые обозначения раньше регулировались государственными стандартами. Время действия стандартов истекло, новые введены не были. Поэтому формально приводимые ниже сведения следует рассматривать как рекомендации, хотя де-факто они являются общепринятыми правилами.
- **При построении структурных схем алгоритмов** используются условные графические обозначения (УГО), символы, блоки, соединительные линии (линии потока) и краткие текстовые пояснения. Наиболее часто используемые символы приведены в таблице. Текст внутри символов конкретизирует выполняемую операцию.

# IV этап. Разработка алгоритма

- **При изображении блоков** используется два размера –  $a$  и  $b$ , причём,  $b = 1,5 a$ . Базовый размер  $a$  выбирается в зависимости от имеющего места и сложности схемы и может принимать значения, кратные 5 (10, 15, 50 и т. д.).
- **Направление потока** задаётся стрелкой при вхождении потока в блок или в другую линию потока. Направления сверху вниз и слева направо принимаются за основные и, если линии потока не имеют изломов, их можно стрелками не обозначать.
- **Признак условия** (для случая не более трёх исходов) проставляется над выходящей линией потока или справа от неё.
- **Порядковый номер или координаты зоны расположения символа** проставляют слева в верхней части символа в разрыве контурной линии.
- Если пояснение не помещается внутри символа, то применяют **комментарий**. Комментарий помещают на свободном месте данного листа и соединяют с поясняемым символом пунктирной линией. Пунктирная линия может обводить группу

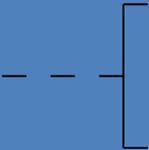
# IV этап. Разработка алгоритма

## Условные обозначения блоков схем алгоритмов

| Обозначение   | Наименование                 | Функция  |
|---|------------------------------|--|
|    | Процесс<br>(вычисления)      | Выполнение операций, в результате которых изменяется значение, форма представления или расположение данных |
|    | Решение<br>(логический блок) | Выбор направления выполнения алгоритма в зависимости от выполнения условия                                 |
|    | Ввод-вывод                   | Ввод данных в форме, пригодной для обработки или отображение результатов обработки (вывод)                 |
|   | Предопределённый процесс     | Использование отдельно написанных программ (подпрограмм)   |
|  | Документ                     | Вывод данных на бумажный носитель  |

# IV этап. Разработка алгоритма

## Условные обозначения блоков схем алгоритмов

| Обозначение  | Наименование              | Функция  |
|--|---------------------------|--|
|   | Пуск-останов              | Начало, конец, прерывание процесса обработки данных  |
|   | Соединитель               | Указание связи между прерванными линиями, соединяющими блоки                                 |
|   | Межстраничный Соединитель | Указание связи между прерванными линиями, соединяющими блоки, расположенные на разных листах |
|  | Комментарий               | Указание связи между пояснениями и элементом, к которому они относятся                       |

# IV этап. Разработка алгоритма

- При разработке алгоритмов часто применяют несколько этапов детализации.
- 
- Сначала разрабатывают **укрупнённую схему**, а затем выполняют **детализацию составных частей**.
- При решении задачи важно понимать, что может существовать несколько алгоритмов. Поэтому нельзя ограничиваться только одним алгоритмом, который первым пришёл в голову. Надо рассмотреть разные варианты и выбрать лучший.

# V этап. Кодирование

- **Кодирование** – это представление алгоритма в форме, допускающей ввод в ЭВМ и последующий перевод на машинный язык.
- Как правило, алгоритм записывается на одном из языков программирования высокого уровня. Создаётся программа.
- **Текст программы** – это полное, законченное и детальное описание алгоритма на языке программирования.
- Затем текст программы специальными служебными приложениями –
- **трансляторами** - переводится в машинный код,
- **Интерпретаторами** - исполняется.
- Чем тщательней проработан алгоритм, тем проще его записать на языке программирования.

# VI этап. Отладка и тестирование

На этом этапе выявляются и исправляются ошибки.

Для выявления ошибок программа выполняется со специально подобранными данными, то есть проверяется на специально подобранных примерах – тестах.

**Тестирование** – это решение специально подобранных примеров (тестов) с целью доказать, что в программе существует хотя бы одна ошибка.

Тестовый прогон считается успешным, если выявлена хотя бы одна ошибка. Однако если тестовый прогон не выявил ошибок, то это не является доказательством того, что в программе вообще нет ошибок.

# VI этап. Отладка и тестирование

**Отладка** – это процесс, следующий за удачным тестовым прогоном. На этапе отладки осуществляют:

- выявление причины ошибки;
- определение места расположения ошибки;
- исправление ошибки.

**Тестирование и отладка требуют значительных ресурсов.**

Поэтому при создании программы стараются использовать технологии, облегчающие проведение тестирования и отладки и уменьшающие затраты на их проведение.

# **VII этап. Выполнение программы**

- К вычислениям приступают только после того, как появится уверенность, что программа обеспечивает правильные результаты.

## **VII этап. Обработка результатов**

- На этом этапе выполняется интерпретация и анализ результатов.
- Полученные результаты соотносятся с поставленной задачей.
- Для анализа и оценки результатов надо разбираться в той предметной области, откуда возникла задача.

# Компьютерное моделирование

## Модели и их классификация

- **Модель** – это некоторая абстракция, представление объектов, процессов или явлений в виде формального описания. Такое формальное описание может быть выполнено с помощью математических зависимостей, словесно, с использованием графиков, таблиц, диаграмм и др.
- **Моделирование** - процесс создания и исследования модели. Никакая модель не может абсолютно точно воспроизвести свойства и поведение объектов. Получаемые на основе модели результаты соответствуют изучаемому объекту или процессу лишь приближённо, с некоторой степенью точности.

# Компьютерное моделирование

## Модели и их классификация

- Создавая модель, учитывают не все свойства объекта, а только те, которые представляют интерес, являются наиболее важными для решаемой задачи.
- Характеристиками объекта, не оказывающими существенного влияния на поведение объекта в рамках поставленной задачи, пренебрегают.
- В зависимости от решаемой задачи, один и тот же объект можно описать разными моделями.
- **Классифицировать модели можно по разным признакам:**
  - по предметной области;
  - в зависимости от поставленной задачи;
  - по применяемым средствам;
  - по используемым методам и др.

# Компьютерное моделирование

## Модели и их классификация

- Существует большое количество классификаций моделей и разные авторы придерживаются только некоторых из них.
- **Используют модели графические, натурные (физические), математические, информационные, концептуальные, логические, эвристические, графовые и др.**
- **Математическая модель** описывает объект или процесс с помощью математических соотношений.
- **Информационно-логическая модель** – это формальное описание объектов, допускающее их представление и обработку средствами ЭВМ.
- Выбор метода моделирования и детализация модели зависит от целей, этапа проектирования, сложности системы и других параметров.

# Компьютерное моделирование

## Модели и их классификация

- Модели характеризуются свойствами: адекватностью, полнотой, детальностью и т.п.
- **Адекватность** – это степень соответствия модели представляемым объектам.
- Первоначально предложенные модели в процессе проектирования могут уточняться, детализироваться.
- Используя созданные модели, при проектировании систем принимаются принципиальные решения, выбираются параметры, выполняется оптимизация.
- Понятно, что ошибки на этапе формирования модели в дальнейшем исправить крайне сложно, а иногда и невозможно.

# Компьютерное моделирование

## Модели и их классификация

- При разработке сложных систем в настоящее время используется системный подход, который построен на переходе от общего к частному.
- При системном подходе к моделированию определяется структура системы – совокупность связей между элементами системы, отражающих их взаимодействие.

# Основные приёмы и методы математического моделирования

**Математическое моделирование** основано на использовании аппарата соответствующих разделов математики.

Это могут быть:

- линейная алгебра и линейное программирование,
- дифференциальное исчисление,
- дифференциальные уравнения,
- численные методы и др.

При проведении компьютерного математического моделирования могут использоваться различные средства:

- языки программирования высокого уровня;
- офисные пакеты – текстовый редактор и табличный процессор;
- математические пакеты.

Выбор математического аппарата и средств для реализации модели, зависит от поставленной задачи, а так же от

# ***Формы представления результатов***

Очень важной при выполнении работ по моделированию является форма представления результатов.

Получаемые с помощью модели результаты могут быть представлены

- в численном виде,
- в виде графиков,
- таблиц,
- диаграмм,
- графов,
- звуков и т.п.

Желательно предусмотреть возможность отображения результатов несколькими способами, чтобы можно было выбрать наиболее удобные для последующего использования. В некоторых случаях форма представления результатов зависит от программных средств, используемых при моделировании.

**Поэтому при выборе программного обеспечения учитывают требования по отображению результатов.**

# Вопросы

1. Перечислите 8 этапов подготовки и решения задач на ЭВМ
2. Чему следует уделять первостепенное внимание при постановке задачи
3. Что такое математическая формулировка (формализация) задачи
4. Что необходимо учитывать при выборе (разработке) метода решения
5. Дайте определение алгоритма
6. Способы описания алгоритмов

**Благодарю  
за внимание !!!**