

ЛЕКЦИЯ 4

ТЕМА: «МОДЕЛИРОВАНИЕ И ПРОЕКТИРОВАНИЕ. МОДЕЛИ ИНФОРМАЦИОННЫХ ПРОЦЕССОВ».

План

1. Математические модели. Моделирование, прогнозирование, проектирование в человеческой деятельности.
2. Имитационное моделирование.
3. Использование инструментов автоматизированного проектирования.
4. Системы, графы взаимодействия.

1. Математические модели. Моделирование, прогнозирование, проектирование в человеческой деятельности.

Математические модели.

Модель — это объект-заместитель объекта-оригинала, обеспечивающий изучение некоторых свойств оригинала. Замещение одного объекта другим с целью получения информации о важнейших свойствах объекта-оригинала с помощью объекта-модели называется **моделированием**. Под **математическим моделированием** будем понимать процесс установления соответствия данному реальному объекту некоторого математического объекта, называемого **математической моделью**, и исследование этой модели, позволяющее получать характеристики рассматриваемого реального объекта. Вид математической модели зависит как от природы реального объекта, так и задач исследования объекта и требуемой достоверности и точности решения этой задачи.

Математическая модель — *приближенное описание объекта моделирования, выраженное с помощью математической символики.*

Математические модели появились вместе с математикой много веков назад. Огромный толчок развитию математического моделирования придало появление ЭВМ. Применение вычислительных машин позволило проанализировать и применить на практике многие математические модели, которые раньше не поддавались аналитическому исследованию. *Реализованная на компьютере математическая модель* называется **компьютерной математической моделью**, а *проведение целенаправленных расчетов с помощью компьютерной модели* называется **вычислительным экспериментом**.

Этапы компьютерного математического моделирования изображены на рисунке.

Первый этап — *определение целей моделирования*. Эти цели могут быть различными:

1. модель нужна для того, чтобы понять, как устроен конкретный объект, какова его структура, основные свойства, законы развития и взаимодействия с окружающим миром (понимание);
2. модель нужна для того, чтобы научиться управлять объектом (или процессом) и определить наилучшие способы управления при заданных целях и критериях (управление);
3. модель нужна для того, чтобы прогнозировать прямые и косвенные последствия реализации заданных способов и форм воздействия на объект (прогнозирование).

Поясним на примерах. Пусть объект исследования — взаимодействие потока жидкости или газа с телом, являющимся для этого потока препятствием. Опыт показывает, что сила сопротивления потоку со стороны тела растет с ростом скорости потока, но при некоторой достаточно высокой скорости эта сила скачком уменьшается с тем, чтобы с дальнейшим увеличением скорости снова возрасти. Что же вызвало уменьшение силы сопротивления? Математическое моделирование позволяет получить четкий ответ: в момент скачкообразного уменьшения сопротивления вихри, образующиеся в потоке жидкости или газа позади обтекаемого тела, начинают отрываться от него и уносятся потоком.

Пример совсем из другой области: мирно сосуществовавшие со стабильными численностями популяции двух видов особей, имеющих общую кормовую базу, "вдруг" начинают резко менять численность. И здесь математическое моделирование позволяет (с известной долей достоверности) установить причину (или по крайней мере опровергнуть определенную гипотезу).

Выработка концепции управления объектом — другая возможная цель моделирования. Какой режим полета самолета выбрать для того, чтобы полет был безопасным и экономически наиболее выгодным? Как составить график выполнения сотен видов работ на строительстве большого объекта, чтобы оно закончилось в максимально короткий срок? Множество таких проблем систематически возникает перед экономистами, конструкторами, учеными.

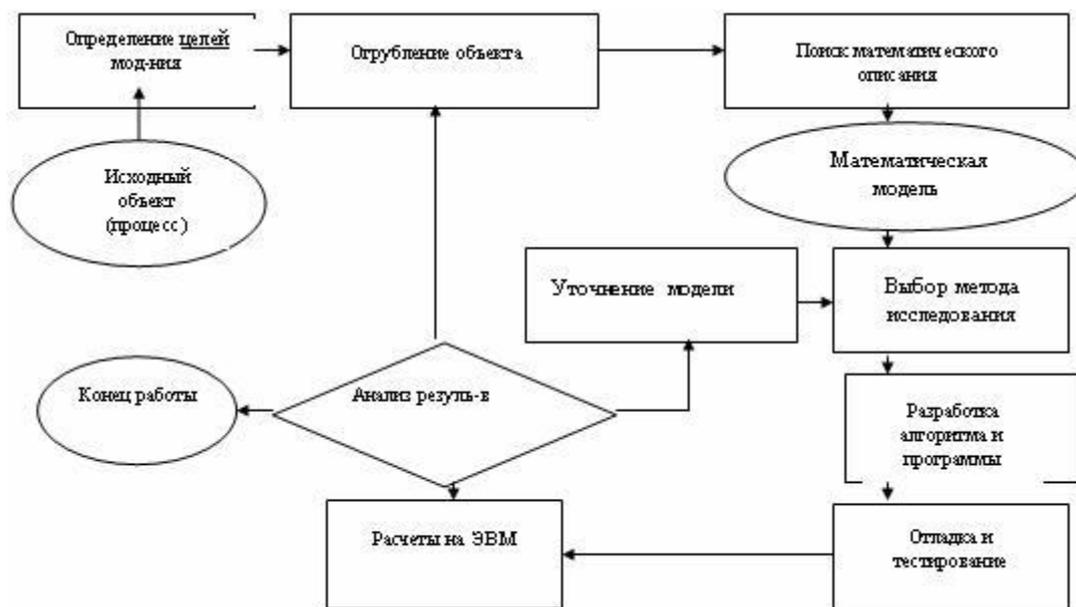


Рис. 1 – Математическое моделирование

Наконец, прогнозирование последствий тех или иных воздействий на объект может быть как относительно простым делом в несложных физических системах, так и чрезвычайно сложным — на грани выполнимости — в системах биолого-экономических, социальных. Если ответить на вопрос об изменении режима распространения тепла в тонком стержне при изменениях в составляющем его сплаве относительно легко, то проследить (предсказать) экологические и климатические последствия строительства крупной ГЭС или социальные последствия изменений налогового законодательства несравненно труднее. Возможно, и здесь методы математического моделирования будут оказывать в будущем более значительную помощь.

Второй этап: определение входных и выходных параметров модели; разделение входных параметров по степени важности влияния их изменений на выходные. Такой процесс называется ранжированием, или разделением по рангам (см. *"Формализация и моделирование"*).

Третий этап: построение математической модели. На этом этапе происходит переход от абстрактной формулировки модели к формулировке, имеющей конкретное математическое представление. Математическая модель — это уравнения, системы уравнений, системы неравенств, дифференциальные уравнения или системы таких уравнений и пр.

Четвертый этап: выбор метода исследования математической модели. Чаще всего здесь используются численные методы, которые хорошо поддаются программированию. Как правило, для решения одной и той же задачи подходит несколько методов, различающихся точностью, устойчивостью и т.д. От верного выбора метода часто зависит успех всего процесса моделирования.

Пятый этап: разработка алгоритма, составление и отладка программы для ЭВМ — трудно формализуемый процесс. Из языков программирования многие профессионалы для математического моделирования предпочитают FORTRAN: как в силу традиций, так и в силу непревзойденной эффективности компиляторов (для расчетных работ) и наличия написанных на нем огромных, тщательно отлаженных и оптимизированных библиотек стандартных программ математических методов. В ходу и такие языки, как PASCAL, BASIC, C, — в зависимости от характера задачи и склонностей программиста.

Шестой этап: тестирование программы. Работа программы проверяется на тестовой задаче с заранее известным ответом. Это — лишь начало процедуры тестирования, которую трудно

описать формально исчерпывающим образом. Обычно тестирование заканчивается тогда, когда пользователь по своим профессиональным признакам сочтет программу верной.

Седьмой этап: собственно вычислительный эксперимент, в процессе которого выясняется, соответствует ли модель реальному объекту (процессу). Модель достаточно адекватна реальному процессу, если некоторые характеристики процесса, полученные на ЭВМ, совпадают с экспериментально полученными характеристиками с заданной степенью точности. В случае несоответствия модели реальному процессу возвращаемся к одному из предыдущих этапов.

Классификация математических моделей

В основу классификации математических моделей можно положить различные принципы. Можно классифицировать модели по отраслям наук (математические модели в физике, биологии, социологии и т.д.). Можно классифицировать по применяемому математическому аппарату (модели, основанные на применении обыкновенных дифференциальных уравнений, дифференциальных уравнений в частных производных, стохастических методов, дискретных алгебраических преобразований и т.д.). Наконец, если исходить из общих задач моделирования в разных науках безотносительно к математическому аппарату, наиболее естественна такая классификация:

- дескриптивные (описательные) модели;
- оптимизационные модели;
- многокритериальные модели;
- игровые модели.

Поясним это на примерах.

Дескриптивные (описательные) модели. Например, моделирование движения кометы, вторгшейся в Солнечную систему, производится с целью предсказания траектории ее полета, расстояния, на котором она пройдет от Земли, и т.д. В этом случае цели моделирования носят описательный характер, поскольку нет никаких возможностей повлиять на движение кометы, что-то в нем изменить.

Оптимизационные модели используются для описания процессов, на которые можно воздействовать, пытаясь добиться достижения заданной цели. В этом случае в модель входит один или несколько параметров, доступных влиянию. Например, меняя тепловой режим в зернохранилище, можно задаться целью подобрать такой режим, чтобы достичь максимальной сохранности зерна, т.е. оптимизировать процесс хранения.

Многокритериальные модели. Нередко приходится оптимизировать процесс по нескольким параметрам одновременно, причем цели могут быть весьма противоречивыми. Например, зная цены на продукты и потребность человека в пище, нужно организовать питание больших групп людей (в армии, детском летнем лагере и др.) физиологически правильно и, одновременно с этим, как можно дешевле. Ясно, что эти цели совсем не совпадают, т.е. при моделировании будет использоваться несколько критериев, между которыми нужно искать баланс.

Игровые модели могут иметь отношение не только к компьютерным играм, но и к весьма серьезным вещам. Например, полководец перед сражением при наличии неполной информации о противостоящей армии должен разработать план: в каком порядке вводить в бой те или иные части и т.д., учитывая и возможную реакцию противника. Есть специальный раздел современной математики — теория игр, — изучающий методы принятия решений в условиях неполной информации.

Методические рекомендации

В школьном курсе информатики начальное представление о компьютерном математическом моделировании ученики получают в рамках базового курса. В старших классах математическое моделирование может глубоко изучаться в общеобразовательном курсе для классов физико-математического профиля, а также в рамках специализированного элективного курса.

Основными формами обучения компьютерному математическому моделированию в старших классах являются лекционные, лабораторные и зачетные занятия. Обычно работа по созданию и подготовке к изучению каждой новой модели занимает 3—4 урока. В ходе изложения материала ставятся задачи, которые в дальнейшем должны быть решены учащимися

самостоятельно, в общих чертах намечаются пути их решения. Формулируются вопросы, ответы на которые должны быть получены при выполнении заданий. Указывается дополнительная литература, позволяющая получить вспомогательные сведения для более успешного выполнения заданий.

Формой организации занятий при изучении нового материала обычно служит лекция. После завершения обсуждения очередной модели учащиеся имеют в своем распоряжении необходимые теоретические сведения и набор заданий для дальнейшей работы. В ходе подготовки к выполнению задания учащиеся выбирают подходящий метод решения, с помощью какого-либо известного частного решения тестируют разработанную программу. В случае вполне возможных затруднений при выполнении заданий дается консультация, делается предложение более детально проработать указанные разделы в литературных источниках.

Наиболее соответствующим практической части обучения компьютерному моделированию является метод проектов. Задание формулируется для ученика в виде учебного проекта и выполняется в течение нескольких уроков, причем основной организационной формой при этом являются компьютерные лабораторные работы. Обучение моделированию с помощью метода учебных проектов может быть реализовано на разных уровнях. Первый — проблемное изложение процесса выполнения проекта, которое ведет учитель. Второй — выполнение проекта учащимися под руководством учителя. Третий — самостоятельное выполнение учащимися учебного исследовательского проекта.

Результаты работы должны быть представлены в численном виде, в виде графиков, диаграмм. Если имеется возможность, процесс представляется на экране ЭВМ в динамике. По окончании расчетов и получению результатов проводится их анализ, сравнение с известными фактами из теории, подтверждается достоверность и проводится содержательная интерпретация, что в дальнейшем отражается в письменном отчете.

Если результаты удовлетворяют ученика и учителя, то работа *считается* завершенной, и ее конечным этапом является составление отчета. Отчет включает в себя краткие теоретические сведения по изучаемой теме, математическую постановку задачи, алгоритм решения и его обоснование, программу для ЭВМ, результаты работы программы, анализ результатов и выводы, список использованной литературы.

Когда все отчеты составлены, на зачетном занятии учащиеся выступают с краткими сообщениями о проделанной работе, защищают свой проект. Это является эффективной формой отчета группы, выполняющей проект, перед классом, включая постановку задачи, построение формальной модели, выбор методов работы с моделью, реализацию модели на компьютере, работу с готовой моделью, интерпретацию полученных результатов, прогнозирование. В итоге учащиеся могут получить две оценки: первую — за проработанность проекта и успешность его защиты, вторую — за программу, оптимальность ее алгоритма, интерфейс и т.д. Учащиеся получают отметки и в ходе опросов по теории.

Существенный вопрос — каким инструментарием пользоваться в школьном курсе информатики для математического моделирования? Компьютерная реализация моделей может быть осуществлена:

- с помощью табличного процессора (как правило, MS Excel);
- путем создания программ на традиционных языках программирования (Паскаль, Бейсик и др.), а также на их современных версиях (Delphi, Visual Basic for Application и т.п.);
- с помощью специальных пакетов прикладных программ для решения математических задач (MathCAD и т.п.).

На уровне основной школы первое средство представляется более предпочтительным. Однако в старшей школе, когда программирование является, наряду с моделированием, ключевой темой информатики, желательно привлекать его в качестве инструмента моделирования. В процессе программирования учащимся становятся доступными детали математических процедур; более того, они просто вынуждены их осваивать, а это способствует и математическому образованию. Что же касается использования специальных пакетов программ, то это уместно в профильном курсе информатики в качестве дополнения к другим инструментам.

2. Имитационное моделирование.

Имитационное моделирование (ситуационное моделирование) — метод, позволяющий строить модели, описывающие процессы так, как они проходили бы в действительности. Такую модель можно «проиграть» во времени как для одного испытания, так и заданного их множества. При этом результаты будут определяться случайным характером процессов. По этим данным можно получить достаточно устойчивую статистику.

Имитационное моделирование — это метод исследования, при котором изучаемая система заменяется моделью, с достаточной точностью описывающей реальную систему, с которой проводятся эксперименты с целью получения информации об этой системе. Экспериментирование с моделью называют имитацией (имитация — это постижение сути явления, не прибегая к экспериментам на реальном объекте).

Имитационное моделирование — это частный случай математического моделирования. Существует класс объектов, для которых по различным причинам не разработаны аналитические модели, либо не разработаны методы решения полученной модели. В этом случае аналитическая модель заменяется имитатором или имитационной моделью.

Имитационным моделированием иногда называют получение частных численных решений сформулированной задачи на основе аналитических решений или с помощью численных методов^[1].

Имитационная модель — логико-математическое описание объекта, которое может быть использовано для экспериментирования на компьютере в целях проектирования, анализа и оценки функционирования объекта.

К имитационному моделированию прибегают, когда:

- дорого или невозможно экспериментировать на реальном объекте;
- невозможно построить аналитическую модель: в системе есть время, причинные связи, последствие, нелинейности, стохастические (случайные) переменные;
- необходимо симитировать поведение системы во времени.

Цель имитационного моделирования состоит в воспроизведении поведения исследуемой системы на основе результатов анализа наиболее существенных взаимосвязей между её элементами или другими словами — разработке симулятора (англ. simulation modeling) исследуемой предметной области для проведения различных экспериментов.

При **имитационном моделировании** реализующий модель алгоритм воспроизводит процесс функционирования системы во времени. Имитируются элементарные явления, составляющие процесс, с сохранением их логической структуры и последовательности протекания во времени.

Основным преимуществом имитационных моделей по сравнению с аналитическими является возможность решения более сложных задач. Имитационные модели позволяют легко учитывать наличие дискретных или непрерывных элементов, нелинейные характеристики, случайные воздействия и др. Поэтому этот метод широко применяется на этапе проектирования сложных систем. Основным средством реализации имитационного моделирования служит ЭВМ, позволяющая осуществлять цифровое моделирование систем и сигналов.

В связи с этим определим словосочетание «**компьютерное моделирование**», которое все чаще используется в литературе. Будем полагать, что **компьютерное моделирование** - это математическое моделирование с использованием средств вычислительной техники. Соответственно, технология компьютерного моделирования предполагает выполнение следующих действий:

- 1) определение цели моделирования;
- 2) разработка концептуальной модели;
- 3) формализация модели;
- 4) программная реализация модели;
- 5) планирование модельных экспериментов;
- 6) реализация плана эксперимента;
- 7) анализ и интерпретация результатов моделирования.

Содержание первых двух этапов практически не зависит от математического метода, положенного в основу моделирования (и даже наоборот - их результат определяет выбор метода). А вот реализация остальных пяти этапов существенно различается для аналитического и имитационного моделирования.

При **имитационном моделировании** используемая ММ воспроизводит алгоритм («логику») функционирования исследуемой системы во времени при различных сочетаниях значений параметров системы и внешней среды. Примером простейшей аналитической модели может служить уравнение прямолинейного равномерного движения. При исследовании такого процесса с помощью имитационной модели должно быть реализовано наблюдение за изменением пройденного пути с течением времени.

Виды имитационного моделирования.

· Агентное моделирование — относительно новое (1990-е-2000-е гг.) направление в имитационном моделировании, которое используется для исследования децентрализованных систем, динамика функционирования которых определяется не глобальными правилами и законами (как в других парадигмах моделирования), а наоборот, когда эти глобальные правила и законы являются результатом индивидуальной активности членов группы. Цель агентных моделей — получить представление об этих глобальных правилах, общем поведении системы, исходя из предположений об индивидуальном, частном поведении её отдельных активных объектов и взаимодействии этих объектов в системе. Агент — некая сущность, обладающая активностью, автономным поведением, может принимать решения в соответствии с некоторым набором правил, взаимодействовать с окружением, а также самостоятельно изменяться.

· Дискретно-событийное моделирование — подход к моделированию, предлагающий абстрагироваться от непрерывной природы событий и рассматривать только основные события моделируемой системы, такие, как: «ожидание», «обработка заказа», «движение с грузом», «разгрузка» и другие. Дискретно-событийное моделирование наиболее развито и имеет огромную сферу приложений — от логистики и систем массового обслуживания до транспортных и производственных систем. Этот вид моделирования наиболее подходит для моделирования производственных процессов. Основан Джеффри Гордоном в 1960-х годах.

· Системная динамика — парадигма моделирования, где для исследуемой системы строятся графические диаграммы причинных связей и глобальных влияний одних параметров на другие во времени, а затем созданная на основе этих диаграмм модель имитируется на компьютере. По сути, такой вид моделирования более всех других парадигм помогает понять суть происходящего выявления причинно-следственных связей между объектами и явлениями. С помощью системной динамики строят модели бизнес-процессов, развития города, модели производства, динамики популяции, экологии и развития эпидемии. Метод основан Джейм Форрестером в 1950 годах.

Этапы имитационного моделирования

1. Формулировка цели моделирования
2. Построение концептуальной модели
3. Выбор аппарата моделирования (система, язык программирования и т.д.)
4. Планирование эксперимента
5. Выполнение эксперимента
6. Обработка, анализ и интерпретация данных эксперимента

3. Использование инструментов автоматизированного проектирования.

Система автоматизированного проектирования — автоматизированная система, реализующая информационную технологию выполнения функций проектирования, представляет собой организационно-техническую систему, предназначенную для автоматизации процесса проектирования, состоящую из персонала и комплекса технических, программных и других средств автоматизации его деятельности. Также для обозначения подобных систем широко используется аббревиатура **САПР**.

Классификация САПР.

Прошло то время, когда инженеры реализовывали свои идеи с помощью кульмана и карандаша.

Сейчас конструктора и технологи (а также архитекторы, исследователи, программисты и т.д.) повсеместно применяют системы автоматизированного проектирования (или САПР): от самых простых "чертилок" до навороченных программ типа Unigraphics NX.

Все САПР можно условно разделить на 3 категории (см. рисунок 2):

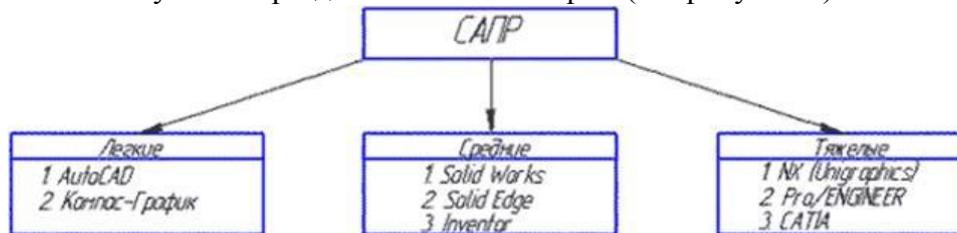


Рис. 2 – Категории САПР

Классификация САПР:

1. Легкие (AutoCAD, Компас-График)
2. Средние (Solid Works, Solid Edge, Компас-3D)
3. Тяжелые (САТІА, Pro/ENGINEER, NX)

Вполне возможно, что ваша работа будет (или может быть уже) связано с проектированием в какой-либо из этих программ.

Рассмотрим виды САПР более подробно.

1. Легкие САПР применяют, в основном, вместо кульмана. Можно сказать, что 2D черчение на компьютере легче, чем за кульманом, ведь программы настроены специальным образом так, чтобы чертить было максимально легко и комфортно. Здесь не нужно следить за качеством графики, все рисует компьютер. Можно без проблем выполнять чертежи любой сложности и размеров (что немаловажно, когда выполняешь сборки формата А1 и А0).

2. Эти САПР используются для 3D моделирования и построения чертежей по 3D моделям. Естественно, увидев 3D модель двигателя вы поймете намного больше, чем по чертежу также как и то, что деталь выполненная станком с ЧПУ по 3D модели будет точнее, чем рабочим по 2D чертежу.

3. Это даже не программы, а целые комплексы программ для крупного предприятия. В одной вы выполняете 3D модель детали (САD-программа), во второй - рассчитываете ее на прочность (САЕ-программа), в третьей - проектируете инструмент для ее изготовления, в четвертой - разрабатываете управляющую программу для станков с ЧПУ (САМ-программа). Ну и стоимость у них соответствующая количеству функций (прибавьте еще пару нулей к сумме, о которой сейчас подумали).

Поэтому для многих компаний по соотношению цена/качество наиболее оптимальной выглядит категория средних САПР, куда входит и программа Компас 3D.

Система автоматизированного проектирования (САПР, САD) повышает эффективность процесса конструирования изделий за счет автоматизации процессов, которые раньше осуществлялись вручную, например традиционного процесса черчения. Но дело не только в удобстве. Многие возможности программного обеспечения САD позволяют пользователям повысить качество изделий и сократить сроки вывода изделий на рынок.

Программы 3D-САПР используются для создания трехмерных представлений конструкций, при этом обычно используется твердотельное моделирование. Исторически программы 2D-САПР использовались для двухмерных представлений конструкции, таких как чертежи, созданные с помощью традиционного черчения. С помощью этих и других функций программного обеспечения САD помогает исследовать первоначальные концепции конструкции, создавать конструкции изделий, выполнять симуляции и анализ, а также проводить инженерные расчеты.

Мы гордимся тем фактом, что компания РТС изобрела и впервые внедрила программное обеспечение САD для параметрического моделирования почти 30 лет назад. Сегодня наше флагманское ПО САD, Creo, является одним из наиболее широко используемых пакетов САD среди специалистов по разработке изделий.

4. Системы, графы взаимодействия.

Когда проводим исследование, чтобы понять, как действует тот или иной объект, очень часто бывает необходимо рассмотреть этот объект во взаимосвязи с окружающим миром, возможно, представить его состоящим из более простых объектов.

Система - совокупность взаимосвязанных между собой объектов. Составные части системы называются элементами или компонентами.

В системе важно выделить объекты, описать их свойства, понять и указать связи между каждым объектом системы. В информатике понятие "система" чаще используют относительно набора технических средств и программ. Системой также называют аппаратную часть компьютера. Когда мы дополняем понятие "система" словом "информационная", то хотим подчеркнуть цель ее создания и функционирования.

Информационная система - взаимосвязанная совокупность средств, методов и персонала, используемая для сохранения, обработки и выдачи информации с целью решения конкретной задачи.

В рамках современного понимания информационной системы компьютер является основным инструментом обработки информации.

В работе информационной системы можно выделить следующие этапы:

1. Зарождение данных - формирование первичных сообщений, которые фиксируют результаты определенных операций, свойства объектов и субъектов управления, параметры процессов, содержание нормативных и юридических актов и т.п..

2. Накопление и систематизация данных. - организация такого их размещения, которое обеспечивало бы быстрый поиск и отбор нужных сведений, методическое обновление данных, защита их от искажений, потери, деформирования целостности и др.

3. Обработка данных - процессы, вследствие которых на основании прежде накопленных данных формируются новые виды данных: обобщающие, аналитические, рекомендательные, прогнозные. Производные данные тоже можно обрабатывать, получая более обобщенные сведения.

4. Отображение данных - представление их в форме, пригодной для восприятия человеком. Прежде всего - это вывод на печать, то есть создание документов на так называемых твердых (бумажных) носителях. Широко используют построение графических иллюстративных материалов (графиков, диаграмм) и формирование звуковых сигналов.

Главное свойство любой системы - это "возникновение системного эффекта" или "принцип эмерджентности". Принцип эмерджентности заключается в том, что при объединении элементов в систему у системы появляются новые свойства, которыми не обладал ни один из элементов в отдельности.

Граф — абстрактный математический объект, представляющий собой множество *вершин* графа и набор *рёбер*, то есть соединений между парами вершин. Например, за множество вершин можно взять множество аэропортов, обслуживаемых некоторой авиакомпанией, а за множество рёбер взять регулярные рейсы этой авиакомпании между городами.

Графы — это специальная математическая абстракция, которая позволяет обсуждать и анализировать широкий круг вещей в реальном мире. Это такая же абстракция, как, например, числа. Числа естественно возникают при подсчёте яблок у Пети, массы сахара в чайной ложке и температуре воздуха за окном. Изучая числа, вы сталкиваетесь с новыми определениями: вам объясняют, что называется квадратным корнем числа, что такое делители числа. Далее, про числа вам рассказывают в школе разные теоремы: например, формулу квадрата суммы двух чисел, теорему о единственности разложения числа на простые множители. Наконец, на программировании вы изучаете алгоритмы, связанные с числами: алгоритм Евклида нахождения НОД, алгоритм проверки числа на простоту. Всё это ждёт нас и при изучении графов.

Начнём с выяснения, зачем же нам нужны графы, какие вещи в реальном мире они позволяют изучать. Посмотрим на участок Москвы с автомобильными дорогами (скриншот сделан с сайта Яндекс.Карты).



Рис. 3 – Фрагмент карты Москвы

Можно описывать сеть дорог как набор перекрёстков, некоторые из которых соединены участками дорог.

Наконец, посмотрим на пример цепи питания в биологии.

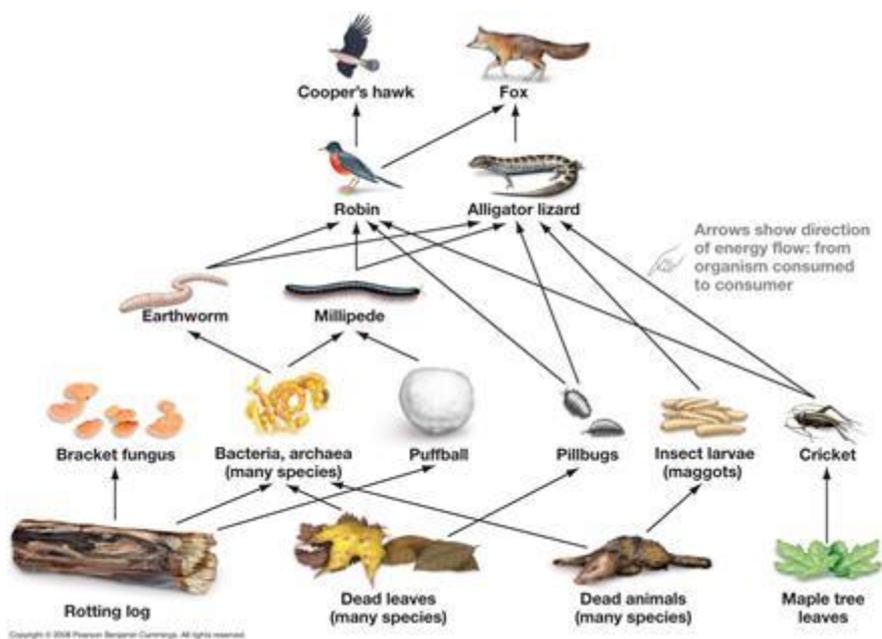


Рис. 4 – Цепь питания

Что общего у всех этих картинок? Главное, что на них изображено — это объекты и связи между ними. В теории графов все такие картинки называются графами. Графы состоят из вершин и рёбер. Так, в графе киевского метрополитена станции считаются вершинами, а перегоны между ними — рёбрами. В графе цепи питания биологические виды являются вершинами, и направленное ребро проведено от одного вида к другому тогда, когда первый вид является пищей для второго.

Итак, графом называется набор вершин и набор рёбер. Каждое ребро соединяет две вершины.

Степенью вершины называется количество рёбер, концом которых она является. Например, в графе метрополитенов большинство станций имеют степень 2, а конечные станции имеют степень 1. В графе славянской языковой группы вершина «западославянский язык» имеет степень 4.



Рис. 5 – Пример графа путь

На рисунке выше вершина А имеет степень 3, вершина В имеет степень 4. Вершина Н имеет степень 0 и называется изолированной вершиной.

Виды графов и пути в графах

Для представления разных объектов и связей между ними используются разные виды графов. Например, графы бывают ориентированные и неориентированные. Ориентированные графы — это графы, в котором у каждого ребра есть начало и конец. Такие рёбра рисуют стрелками. В наших примерах граф цепи питания ориентированный, а граф метрополитена — неориентированные.

Подумаем, нужно ли считать граф дорог ориентированным. Пусть мы пишем программу, которая по графу дорог находит автомобильный маршрут между двумя точками в городе. Поскольку в городе бывают улицы с односторонним движением, то наша программа должна это учитывать. Значит, на каждом ребре нужно хранить направление — возможное направление проезда по ребру. Если по дороге можно проехать в обе стороны, то рисуют два ребра со стрелками в разные стороны.

Путём в графе называется любая последовательность вершин, в которой каждые две соседние вершины соединены ребром. На рисунке выше $A \rightarrow C \rightarrow B \rightarrow G$ — это путь из вершины А в вершину G. Есть и более короткий путь из А в G: путь $A \rightarrow B \rightarrow G$. Длиной пути называется количество рёбер в нём. Таким образом, кратчайший путь из А в G имеет длину 2.

Циклом в графе называют путь, у которого начальная и конечная вершина совпадают. На рисунке выше путь $A \rightarrow C \rightarrow B \rightarrow D \rightarrow A$ является циклом.

Если вы внимательно посмотрите на определение пути и цикла, то увидите, что путём так же можно считать последовательность $A \rightarrow D \rightarrow B \rightarrow D \rightarrow B$, а последовательность $F \rightarrow E \rightarrow F$ удовлетворяет определению цикла. Чтобы исключить такие патологические ситуации из рассмотрения, обычно вводят понятия простого пути и простого цикла. Простой путь — это путь, в котором нет повторяющихся рёбер. Простой цикл это цикл, который является простым путём.

(Осторожно, сейчас мы введём очень сложное понятие.) Компонентой связности неориентированного графа называется любой набор его вершин, который удовлетворяет следующим двум свойствам:

1. между любыми двумя вершинами набора существует путь;
2. набор нельзя расширить, добавив в него ещё хотя бы одну вершину, чтобы при этом осталось верным свойство 1.

Всякий неориентированный граф разбивается на свои компоненты связности. На рисунке выше имеются три компоненты связности: $\{A, B, C, D, G\}$, $\{E, F\}$ и $\{H\}$. Граф, у которого только одна компонента связности, называется связным. Граф, у которого более одной компоненты связности, называется несвязным. На рисунке выше изображён несвязный граф.

В ориентированном графе путём называется любая последовательность вершин, в которой соседние вершины соединены ребром, и это ребро идёт «слева направо» (в нужную сторону). Например, на рисунке ниже $A \rightarrow B \rightarrow C \rightarrow D$ является путём, а $A \rightarrow D \rightarrow C \rightarrow B$ — не является (потому что в графе нет рёбер $A \rightarrow D$ и $C \rightarrow B$).

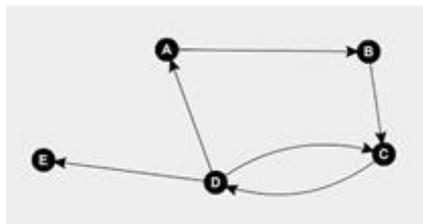


Рис. 6 – Пример графа цикл

В ориентированном графе некоторые понятия, которые мы ввели для неориентированных графов, имеют свои аналоги. Например, наряду с понятием «степень вершины», в ориентированных графах используются понятия полустепень захода (количество рёбер, входящих в вершину) и полустепень исхода (количество рёбер, исходящих из вершины). На рисунке выше вершина D имеет полустепень захода 1 и полустепень исхода 3.

Наконец, отметим, что в некоторых графах допустимы ситуации, изображённые на следующей картинке.

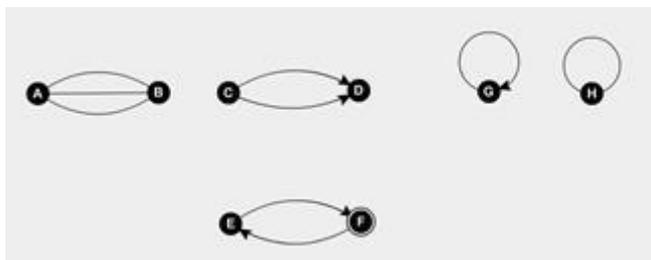


Рис. 7 – Пример графа

Между вершинами A и B, а также между вершинами C и D проведены кратные рёбра. Заметим, что между вершинами E и F кратных рёбер нет, поскольку ориентированные рёбра считаются кратными, только если у них совпадают начала и концы с учётом ориентации. Рёбра, исходящие из вершин G и H, называются петлями. Про некоторые графы специально говорят «графы без петель и кратных рёбер», чтобы подчеркнуть, что в них не встречаются ситуации, аналогичные изображённым выше.

Деревья

На практике часто встречаются графы, которые обладают какими-нибудь особенностями строения. Один из часто встречающихся видов графов — это деревья. Дерево — это связный неориентированный граф без петель и кратных рёбер, в котором нет циклов. Типичный пример дерева изображён на рисунке ниже.

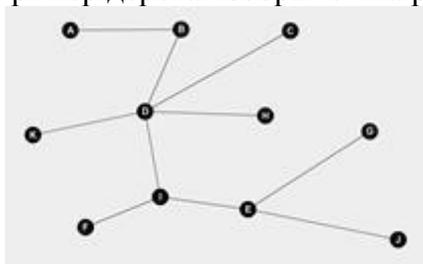


Рис. 8 – Пример графа дерево

Деревья обладают рядом особых свойств. Например, в дереве между любыми двумя вершинами существует единственный простой путь. Действительно, если бы между какими-нибудь двумя вершинами существовало более одного простого пути, то отсюда бы следовало, что в графе есть простой цикл.

Ещё одно удивительное свойство деревьев — это связь между количеством вершин и количеством рёбер. Договоримся обозначать буквой V количество вершин (от англ. vertex «вершина»), а буквой E — количество рёбер (от англ. edge «ребро»). Например, у дерева на рисунке выше $V = 11$, $E = 10$. Мы видим, что для графа на рисунке $E = V - 1$.

Чтобы понять, всегда ли это будет верно, рассмотрим висячие вершины. Висячей вершиной называется вершина степени 1. На рисунке выше висячими являются вершины А, С, F, G, H, J и K. Заметим, что в дереве, в котором есть хотя бы две вершины, всегда есть хотя бы одна висячая вершина. Действительно, выберем произвольную вершину дерева и пойдём из неё гулять по рёбрам дерева в произвольном направлении, не возвращаясь назад. Поскольку циклов в дереве нет, то с каждым шагом мы будем посещать всё новые и новые вершины и в какой-то момент придём в вершину, из которой никуда пойти нельзя. Эта вершина и будет висячей.

Правда ли, что если в дереве есть хотя бы две вершины, то в нём есть хотя бы две висячие вершины? А правда ли, что если в дереве есть хотя бы три вершины, то в нём есть хотя бы три висячие вершины?

Теорема. В любом дереве $E = V - 1$.

Доказательство. Как мы выяснили, если в дереве хотя бы две вершины, то в нём есть хотя бы одна висячая вершина. Выберем её и удалим из графа её и ребро, за которое она присоединена к графу. При этом количество вершин и рёбер уменьшится на единицу. С новым графом сделаем ту же операцию. В конце концов, когда мы удалим всё, что можно, мы получим граф из одной вершины. Для него $V = 1$, $E = 0$, т.е. $E = V - 1$. Значит, и в исходном дереве выполнялось $E = V - 1$.

Контрольные вопросы

1. Что представляет собой модель?
2. Перечислите этапы компьютерного математического моделирования.
3. Как классифицируются математические модели?
4. Что представляет собой имитационное моделирование?
5. Назовите виды имитационного моделирования?
6. Что такое «САПР»?
7. Как классифицируется САПР?
8. Дайте определение понятию система.
9. Что представляет собой граф?
10. Какие бывают виды графов?

Список использованной литературы

1. Цветкова М.С., Великович Л.С. Информатика и ИКТ: учебник. – М.: Издательский центр «Академия», 2012, 1.1, 1.3.
2. Бондаренко Е. А., Журиной А. А., Милютин И. А. Технические средства обучения в современной школе: Пособие для учителя и директора школы. / Под ред. А. А. Журиной. – М.: «ЮНВЕС», 2004. – 416 с.
3. Захарова И. Г. Информационные технологии в образовании: Учебн. пособие для высш. учебн. заведений. – М.: Издательский центр «Академия», 2005. – 192 с.
4. Информатика. 10-11 класс / Под ред. Н. В. Макаровой. – СПб.: Питер, 2004. – 300 с.: ил.
5. Информатика: Учеб. пособие для студ. сред. проф. образования / Е. А. Колмыкова, И. А. Кумскова. – М.: Издательский центр «Академия», 2005. – 416 с.
6. Угринович Н. Д. Информатика и информационные технологии. Учебник для 10 – 11 классов. / Н. Д. Угринович. – 4-е изд. – М.: БИНОМ. Лаборатория знаний, 2007. – 511 с.: ил.
7. Оре О. Теория графов. — М.: Наука, 1968. — 336 с.
8. Уилсон Р. Введение в теорию графов. — М.: Мир, 1977. — 208 с.
9. Харари Ф. Теория графов. — М.: Мир, 1973.
10. Кормен Т. М. и др. Часть VI. Алгоритмы для работы с графами // Алгоритмы: построение и анализ = Introduction to Algorithms. — 2-е изд. — М.: Вильямс, 2006. — С. 1296.