

**МИНИСТЕРСТВО НАРОДНОГО ОБРАЗОВАНИЯ РЕСПУБЛИКИ
УЗБЕКИСТАН**

**НУКУССКИЙ ГОСУДАРСТВЕННЫЙ ПЕДАГОГИЧЕСКИЙ ИНСТИТУТ
ИМЕНИ АЖИНИЯЗА**

ФИЗИКО-МАТЕМАТИЧЕСКИЙ ФАКУЛЬТЕТ

КАФЕДРА "МЕТОДИКА ПРЕПОДАВАНИЯ ИНФОРМАТИКИ"

КУРС ЛЕКЦИЙ

ПО ПРЕДМЕТУ

**"ИНФОРМАТИКА И
ИНФОРМАЦИОННЫЕ
ТЕХНОЛОГИИ"**

НУКУС-2015г.

Тема 1. Понятие информации. Общая характеристика процессов сбора, передачи, обработки и накопления информации.

Информацией называют сведения о тех или иных объектах, явлениях или процессах в окружающей среде. Любая форма человеческой деятельности связана с передачей и обработкой информации. Она необходима для правильного управления окружающей действительностью, достижения поставленных целей и в конечном счете - для существования человека. Любая система: социально-экономическая, техническая, или система в живой природе действует в постоянной взаимосвязи с внешней средой – другими системами более высокого и более низкого уровней. Взаимосвязь осуществляется посредством информации, которая передает как команды управления, так и сведения, необходимые для принятия правильных решений. Понятие информации как важнейшего элемента системы, охватывающего все стороны ее жизнедеятельности можно считать универсальным, применимым к любым системам.

Единого научного мнения о количественном смысле понятия "информация" не существует. Разные научные направления дают различные определения исходя из тех объектов и явлений, которые они изучают. Некоторые из них считают, что информация может быть выражена количественно, давая определения количества и объема информации (меры информации), другие ограничиваются качественными толкованиями.

Синтаксическая мера информации используется для количественного выражения обезличенной информации, не выражающей смыслового отношения к объектам.

Семантическое (смысловое) количество информации измеряется тезаурусной мерой. Она выражает способность наблюдателя (пользователя) принимать поступившее сообщение.

Прагматическая мера информации. Создание порции информации происходит по некоторой причине, а получение информации может привести к некоторому результату. Количественной мерой информации в этом случае может быть степень реакции системы на данную информацию.

Эта мера определяет полезность информации (ценность) для достижения пользователем поставленной цели. Она также относительна, обусловлена особенностями использования информации в той или иной системе. Ценность информации целесообразно измерять в тех же самых единицах (или близких к ним), в которых измеряется поставленная цель.

Информационные процессы (сбор, обработка и передача информации) всегда играли важную роль в науке, технике и жизни общества. В ходе эволюции человечества просматривается устойчивая тенденция к автоматизации этих процессов, хотя их внутреннее содержание по существу осталось неизменным.

Сбор информации — это деятельность субъекта, в ходе которой он получает сведения об интересующем его объекте. Сбор информации может производиться или человеком, или с помощью технических средств и систем — аппаратно. Например, пользователь может получить информацию о движении поездов или самолетов сам, изучив расписание, или же от другого человека непосредственно, либо через какие-то документы, составленные этим человеком, или с помощью технических средств (автоматической справки, телефона и т. д.). Задача сбора информации не может быть решена в отрыве от других задач, — в частности, задачи обмена информацией (передачи).

Обмен информацией — это процесс, в ходе которого источник информации ее передает, а получатель — принимает. Если в передаваемых сообщениях обнаружены ошибки, то организуется повторная передача этой информации. В результате обмена информацией между источником и получателем устанавливается своеобразный «информационный баланс», при котором в идеальном случае получатель будет располагать той же информацией, что и источник.

Обмен информации производится с помощью сигналов, являющихся ее материальным носителем. Источниками информации могут быть любые объекты реального мира, обладающие определенными свойствами и способностями. Если объект относится к неживой природе, то он вырабатывает сигналы, непосредственно отражающие его свойства. Если объектом-источником является человек, то вырабатываемые им сигналы могут не только непосредственно отражать его свойства, но и соответствовать тем знакам, которые человек вырабатывает с целью обмена информацией.

Принятую информацию получатель может использовать неоднократно. С этой целью он должен зафиксировать ее на материальном носителе (магнитном, фото, кино и др.). Процесс формирования исходного, несистематизированного массива информации называется накоплением информации. Среди записанных сигналов могут быть такие, которые отражают ценную или часто используемую информацию. Часть информации в данный момент времени особой ценности может не представлять, хотя, возможно, потребуется в дальнейшем.

Хранение информации — это процесс поддержания исходной информации в виде, обеспечивающем выдачу данных по запросам конечных пользователей в установленные сроки.

Обработка информации — это упорядоченный процесс ее преобразования в соответствии с алгоритмом решения задачи.

После решения задачи обработки информации результат должен быть выдан конечным пользователям в требуемом виде. Эта операция реализуется в ходе решения задачи выдачи информации. Выдача информации, как правило, производится с помощью внешних устройств ЭВМ в виде текстов, таблиц, графиков и пр.

Информационная технология — это совокупность методов, производственных процессов и программно-технических средств, объединенных в технологическую цепочку, обеспечивающую сбор, обработку, хранение, распространение и отображение информации с целью снижения трудоемкости процессов использования информационного ресурса, а также повышения их надежности и оперативности.

Информационные технологии характеризуются следующими основными свойствами:

- предметом (объектом) обработки (процесса) являются данные;
- целью процесса является получение информации;
- средствами осуществления процесса являются программные, аппаратные и программно-аппаратные вычислительные комплексы;
- процессы обработки данных разделяются на операции в соответствии с данной предметной областью;
- выбор управляющих воздействий на процессы должен осуществляться лицами, принимающими решение;
- критериями оптимизации процесса являются своевременность доставки информации пользователю, ее надежность, достоверность, полнота.

Литература: [1], с. 5-19; [2], с. 13-19.

Тема 2. Технические средства реализации информационных процессов.

Компьютерная техника представляет собой материальную основу информационной технологии, с помощью которой осуществляется сбор, хранение, передача и обработка информации.

Следует различать понятия архитектуры и структуры ЭВМ. Под архитектурой ЭВМ принято понимать совокупность общих принципов организации аппаратно-программных средств и их основных характеристик, определяющая функциональные возможности ЭВМ при решении соответствующих типов задач.

Архитектуру вычислительного средства необходимо отличать от его структуры. Структура вычислительного средства определяет его текущий состав на определенном уровне детализации и описывает связи внутри средства. Архитектура же определяет основные правила взаимодействия составных элементов вычислительного средства, описание которых выполняется в той мере, в какой необходимо для формирования правил их взаимодействия. Она устанавливает не все связи, а наиболее необходимые, которые должны быть известны для более грамотного использования применяемого средства.

Несмотря на то, что современные ЭВМ внешне несколько не похожи на первые модели, основополагающие идеи, принятые в них и связанные с понятием алгоритма, разработанным Аланом Тьюрингом, а также архитектурной реализацией, предложенной Джоном фон Нейманом, пока не претерпели кардинальных изменений (за исключением, конечно же, систем параллельной обработки информации).

Любая ЭВМ неймановской архитектуры содержит следующие базисные устройства:

- арифметико-логическое устройство (АЛУ);
- устройство управления (УУ)
- запоминающее устройство (ЗУ);
- устройства ввода-вывода (УВВ);
- пульт управления (ПУ).

В современных ЭВМ АЛУ и УУ объединены в одно общее устройство, называемое центральным процессором.

Современный персональный компьютер содержит следующие основные устройства:

- устройства внешней памяти, чаще всего магнитные диски, на которых информация и программы сохраняются и при выключении питания;
- оперативную память, в которой информация и программа находится при выполнении нужных действий;
- процессор, который собственно и выполняет команды в том порядке, который предписан программой;
- устройства ввода и вывода (например, клавиатура и монитор), через которые компьютер получает информацию и программы извне или передает их наружу.

Внешне современные персональные компьютеры состоят обычно из трех частей: системного блока, клавиатуры, монитора. Системный блок является обрабатывающим блоком, в нем расположены, как минимум, следующие узлы компьютера:

- электронные схемы (процессор, оперативная память, контроллеры устройств и т.д.);
- блок электропитания, преобразующий электроэнергию переменного тока в электроэнергию постоянного тока низкого напряжения, подаваемую на электронные схемы;
- накопитель (дискетод) на гибких магнитных дисках, используемый для переноса программ и данных между разными компьютерами;
- накопитель на жестком магнитном диске, предназначенный для длительного хранения больших объемов информации в компьютере.

К периферийным устройствам относятся: принтеры, сканеры, акустические колонки, источники бесперебойного питания, модемы, сменные накопители памяти, т. е. те устройства, без которых компьютер может функционировать самостоятельно.

Принтеры. Принтеры (print — печать) — это автоматические печатающие устройства, предназначенные для печатания на бумагу результатов работы на компьютере (текстов, рисунков, графиков).

По принципу действия различают ударно-матричные, струйные, лазерные и другие принтеры.

Сканер — это устройство, которое позволяет вводить в компьютер изображение текстов, рисунков, фотографий непосредственно с бумажного документа. Настольные сканеры подразделяются на планшетные, роликовые и проекционные.

Для долговременного архивного хранения информации или для перезаписи больших массивов данных на другой компьютер используются сменные накопители памяти.

К сменным носителям информации относятся накопители памяти большой емкости, которые можно отключать от компьютера и использовать по назначению. Накопители обычно имеют небольшую массу и объем, их легко можно переносить даже в дипломате.

Использование сменных накопителей дает возможность полностью исключить несанкционированный доступ к конфиденциальным данным. Информация, оставленная на внутреннем жестком диске компьютера, всегда доступна неограниченному кругу лиц и ее могут случайно испортить.

Для накопления данных используются следующие накопители: на основе дискет, типа винчестера, магнитооптические, стримеры, лазерные, модульные и др.

Литература: [1], с. 20-39; [2], с. 100-156.

Тема 3. Программные средства реализации информационных процессов.

Программное обеспечение является принципиально важной частью информационной системы, построенной на основе компьютеров. Именно в нем заложены алгоритмы, обеспечивающие автоматизацию обработки информации.

Программа (program) - упорядоченная последовательность команд компьютера для решения задачи.

Программное обеспечение (software) - совокупность программ обработки данных и необходимых для их эксплуатации документов.

Все программное обеспечение (ПО) разделяется на системное и прикладное.

Системное ПО (System Software) – это программы и комплексы программ, являющиеся общими для всех, кто совместно использует технические средства компьютера, и применяемые как для автоматизации разработки (создания) новых программ, так и для организации выполнения программ существующих.

Системное ПО делится на пять групп:

1. Операционные системы (ОС).
2. Системы управления файлами.
3. Интерфейсные оболочки для взаимодействия пользователя с ОС и программные среды.
4. Системы программирования.
5. Утилиты.

Под операционной системой (ОС) понимают комплекс управляющих и обрабатывающих программ, который, с одной стороны, выступает как интерфейс (интерфейс – это комплекс спецификаций, определяющих конкретный способ взаимодействия пользователя с компьютером) между аппаратурой компьютера и пользователем с его задачами, а с другой – предназначен для наиболее эффективного использования ресурсов вычислительной системы и организации надежных вычислений.

Назначение системы управления файлами – организация более удобного доступа к данным, организованным как файлы. Именно благодаря системе управления файлами вместо низкоуровневого доступа к данным с указанием конкретных физических адресов нужной нам записи используется логический доступ с указанием имени файла и записи в нем.

Для удобства взаимодействия с ОС могут использоваться дополнительные интерфейсные оболочки. Их основное назначение - либо расширить возможности по управлению ОС, либо изменить встроенные в систему возможности.

Система программирования состоит из следующих компонент: транслятор соответствующего языка, библиотеки подпрограмм, редакторы, компоновщики и отладчики.

Под утилитами понимают специальные системные программы, с помощью которых можно как обслуживать саму операционную систему, так и подготавливать для работы носители данных, выполнять перекодирование данных, осуществлять оптимизацию размещения данных на носителе и производить некоторые другие работы, связанные с обслуживанием вычислительной системы.

Прикладные программы предназначены для того, чтобы обеспечить применение вычислительной техники в различных сферах деятельности человека. Поэтому этот класс программ представляет наибольший интерес для массового пользователя компьютеров.

Из-за огромного разнообразия прикладного ПО существует множество вариантов его классификации. Рассмотрим наиболее общую классификацию прикладных программ. Разделим данное ПО на 2 больших класса:

1. ПС общего назначения. К таким относятся программы, обеспечивающие выполнение наиболее часто используемых, универсальных задач (текстовые редакторы, табличные процессоры, графические редакторы, СУБД и т.д.).

2. ПС профессионального уровня. Программы этого класса ориентируются на достаточно узкую предметную область, но проникают в нее достаточно глубоко (издательские системы, САПР - системы автоматизированного проектирования, программы 3D-графики, программы видеомонтажа, нотные редакторы, АСУ - автоматизированные системы управления и т.д.).

Каждый класс разбивается на многочисленные подклассы.

1. Офисное ПО

Помимо рассмотренных программных средств, возникает много трудно классифицируемого по способам применения ПО. Многие из них роднит среда их применения - в делопроизводстве, в различных аспектах управления мелкими и средними предприятиями. Эти программы можно назвать офисными. Сюда можно отнести следующие виды программ:

2. Средства обработки текстовой информации.

3. Средства табличной обработки информации

4. Системы управления базами данных

Средства разработки презентаций и публикаций

○ Автоматизация ввода информации в компьютер

○ Автоматизация перевода документов

При выполнении разнообразной работы на компьютере обычно не обходятся без использования компьютерной графики. В каждой более или менее серьезной программе имеются средства работы с графическими объектами.

Несмотря на то, что для работы с компьютерной графикой существует множество классов ПО, различают обычно три вида компьютерной графики, которые отличаются принципами формирования изображения при отображении на экране монитора или при печати на бумаге.

Растровая графика. Применяется при разработке электронных и полиграфических изданий. Иллюстрации, выполненные средствами растровой графики чаще всего попадают в компьютер с помощью сканера или фото- и видеокамер и обрабатываются компьютерными программами.

Основным элементов растрового изображения является точка. Если изображение экранное, то эта точка называется пикселем. В зависимости от того, какое графическое разрешение экрана используется операционной системой, на экране могут размещаться изображения, имеющие 640x480, 800x600, 1024x768 и более пикселей.

С размером изображения непосредственно связано его разрешение, которое измеряется в точках на дюйм (dots per inch - dpi) (1 дюйм = 25,4 мм). Полиграфическая печать полноцветного изображения требует разрешения не менее 200-300 dpi.

Основным недостатком растровых изображений является невозможность их увеличения для рассмотрения деталей. При увеличении изображения точки становятся крупнее и искажают рисунок. Этот эффект называется пикселизацией.

Векторная графика. Основным элементом в векторной графике является линия. Линия - это элементарный объект векторной графики. Все объекты в изображении состоят из линий. Например, объект четырехугольник можно рассматривать как четыре связанные линии, а объект куб - как двенадцать связанных линий или как шесть связанных четырехугольников. Из-за такого подхода векторную графику часто называют объектно-ориентированной графикой.

Как и все объекты, линии имеют свойства: форма, толщина, цвет, тип. Замкнутые линии имеют свойство заполнения цветом, текстурой, картой. Простейшая линия, если она не замкнута, имеет две вершины, которые называются узлами. Узлы также имеют свойства, от которых зависит, как выглядит вершина линии и как две линии сопрягаются между собой.

Так как все изображения выводятся на экран в виде точек, то перед выводом векторных изображений программа производит вычисление координат экранных точек в изображении объекта, поэтому векторную графику иногда называют вычисляемой.

Фрактальная графика в общем строится на основе вычислений, что и векторная. Но в отличие от нее никакие объекты в памяти компьютера не хранятся. Изображение строится по уравнению, поэтому в памяти хранится только формула. Изменяя коэффициенты в уравнении, можно получить совершенно другое изображение.

Литература: [1], с. 37-49; [2], с. 150-171.

Тема 4. Модели решения функциональных и вычислительных задач.

Модель - объект или описание объекта, системы для замещения (при определенных условиях предложениях, гипотезах) одной системы (т.е. оригинала) другой системы для изучения оригинала или воспроизведения его каких - либо свойств. Модель - результат отображения одной структуры на другую.

Под моделированием понимается процесс построения, изучения и применения моделей. Оно тесно связано с такими категориями, как абстракция, аналогия, гипотеза и др. Процесс моделирования обязательно включает и построение абстракций, и умозаключения по аналогии, и конструирование научных гипотез. Главная особенность моделирования в том, что это метод опосредованного познания с помощью объектов-заместителей. Модель выступает как своеобразный инструмент познания, который исследователь ставит между собой и объектом и с помощью которого изучает интересующий его объект. Именно эта особенность метода моделирования определяет специфические формы использования абстракций, аналогий, гипотез, других категорий и методов познания.

Применительно к естественным и техническим наукам принято различать следующие виды моделирования:

- концептуальное моделирование, при котором совокупность уже известных фактов или представлений относительно исследуемого объекта или системы истолковывается с помощью некоторых специальных знаков, символов, операций над ними или с помощью естественного или искусственного языков;
- физическое моделирование, при котором модель и моделируемый объект представляют собой реальные объекты или процессы единой или различной физической природы, причем между процессами в объекте-оригинале и в модели выполняются некоторые соотношения подобия, вытекающие из схожести физических явлений;
- структурно-функциональное моделирование, при котором моделями являются схемы (блок-схемы), графики, чертежи, диаграммы, таблицы, рисунки, дополненные специальными правилами их объединения и преобразования;

- математическое (логико-математическое) моделирование, при котором моделирование, включая построение модели, осуществляется средствами математики и логики;

- имитационное (программное) моделирование, при котором логико-математическая модель исследуемого объекта представляет собой алгоритм функционирования объекта, реализованный в виде программного комплекса для компьютера.

Разумеется, перечисленные выше виды моделирования не являются взаимоисключающими и могут применяться при исследовании сложных объектов либо одновременно, либо в некоторой комбинации. Кроме того, в некотором смысле концептуальное и, скажем, структурно-функциональное моделирование неразличимы между собой, так как те же блок-схемы, конечно же, являются специальными знаками с установленными операциями над ними.

Литература: [3], с. 5-19; [4], с. 135-156.

Тема 5. Алгоритмизация и программирование.

Алгоритмизация – это метод описания систем или процессов путем составления алгоритмов их функционирования.

Алгоритмом называется точное предписание, определяющее вычислительный процесс, ведущий от варьируемых начальных данных к искомому результату (ГОСТ 19.781- 74).

Понятие алгоритма является центральным понятием информатики. Термин «алгоритм» своим происхождением связан с именем узбекского математика Аль – Хоремзи, который еще в IX в. сформулировал правила выполнения четырех арифметических действий.

Алгоритм решения задачи имеет ряд обязательных свойств:

- дискретность - разбиение процесса обработки информации на более простые этапы (шаги выполнения), выполнение которых компьютером или человеком не вызывает затруднений;

- определенность алгоритма - однозначность выполнения каждого отдельного шага преобразования информации;

- выполнимость - конечность действий алгоритма решения задач, позволяющая получить желаемый результат при допустимых исходных данных за конечное число шагов;

- массовость - пригодность алгоритма для решения определенного класса задач.

В алгоритме отражаются логика и способ формирования результатов решения с указанием необходимых расчетных формул, логических условий, соотношений для контроля достоверности выходных результатов. В алгоритме обязательно должны быть предусмотрены все ситуации, которые могут возникнуть в процессе решения комплекса задач.

Алгоритм решения комплекса задач и его программная реализация тесно взаимосвязаны. Специфика применяемых средств проектирования алгоритмов и используемых при этом инструментальных средств разработки программ может повлиять на форму представления и содержание алгоритма обработки данных.

Из известных основных способов представления алгоритмов:

- словесный;
- структурно-стилизированный (псевдокод);
- графический;
- программный.

Графическая форма, в виде блок-схем наиболее удобна. Блок-схемой называется графическое изображение структуры алгоритма, в которой каждый этап процесса обработки данных представляется в виде графических символов (блоков)- геометрических

фигур, в контуры которых вписано содержание операции. Фигура, представленная в виде блочного символа (блока) называется символом действия.

Форма и размеры блоков, их перечень, наименование, правила выполнения схем определены ГОСТом, Требованиями Единой системы программной документации (ЕСПД) ГОСТ 19.002-80 и ГОСТ 19-003-80.

Программирование (programming) - теоретическая и практическая деятельность, связанная с созданием программ.

Программирование является собирательным понятием и может рассматриваться и как наука и как искусство, на этом основан научно-практический подход к разработке программ. Программа - результат интеллектуального труда, для которого характерно творчество, а оно, как известно, не имеет четких границ. В любой программе присутствует индивидуальность ее разработчика, программа отражает определенную степень искусства программиста. Вместе с тем программирование предполагает и рутинные работы, которые могут и должны иметь строгий регламент выполнения и соответствовать стандартам.

Язык программирования — формальная знаковая система, предназначенная для описания алгоритмов в форме, которая удобна для исполнителя (например, компьютера). Язык программирования определяет набор лексических, синтаксических и семантических правил, используемых при составлении компьютерной программы. Он позволяет программисту точно определить то, на какие события будет реагировать компьютер, как будут храниться и передаваться данные, а также какие именно действия следует выполнять над этими данными при различных обстоятельствах.

Процесс решения задачи на компьютере - это совместная деятельность человека и ЭВМ. Этот процесс можно представить в виде нескольких последовательных этапов. На долю человека приходятся этапы, связанные с творческой деятельностью - постановкой, алгоритмизацией, программированием задач и анализом результатов, а на долю компьютера - этапы обработки информации в соответствии с разработанным алгоритмом:

1. Постановка задачи; 2. Анализ и исследование задачи, модели; 3. Разработка алгоритма; 4. Программирование; 5. Тестирование и отладка; 6. Анализ результатов решения задачи и уточнение в случае необходимости математической модели с повторным выполнением этапов 2-5; 7. Сопровождение программы.

ЭВМ непосредственно выполняет программы на машинном языке программирования данной ЭВМ. При этом программа представляет собой набор отдельных команд компьютера. Эти команды являются достаточно «простыми», например, сложение, умножение, сравнение или пересылка отдельных данных. Каждая команда содержит в себе сведения о том, какая операция должна быть выполнена (код операции), с какими операндами (адреса данных или непосредственно сами данные) выполняются вычисления и куда (адрес) должен быть помещен результат.

Человеку свойственно формулировать и решать задачи в выражениях более общего характера, чем команды ЭВМ. Поэтому с развитием программирования появились языки, ориентированные на более высокий уровень абстракции при описании решаемой на ЭВМ задачи. Эти языки получили название языков высокого уровня. Их теоретическую основу составляют алгоритмические языки, например, Паскаль, Си, Бейсик, Фортран, PL/1.

Каждый язык программирования может быть представлен в виде набора формальных спецификаций, определяющих его синтаксис и семантику

Эти спецификации обычно включают в себя описание:

- Типов и структур данных
- Операционную семантику (алгоритм вычисления конструкций языка)
- Семантические конструкции языка
- Библиотеки примитивов (например, инструкции ввода-вывода)
- Философии, назначения и возможностей языка

Последовательное выполнение операторов в языке программирования означает, что они выполняются в том порядке, как записаны в программе. В Паскале этот порядок

подразумевается по умолчанию, если не применены структурные операторы, меняющие его.

Условное выполнение означает, что при определенных условиях должны выполняться одни операторы, а при других условиях - другие. Для организации такого порядка служат условные операторы. Условное выполнение также часто называют ветвлением.

При повторяющемся выполнении некоторые операторы выполняются несколько раз, несмотря на то, что в программе они записаны один раз. Такой рациональный способ записи и выполнения обеспечивается операторами цикла.

Литература: [1], с. 35-49; [3], с. 100-156.

Тема 6. Языки программирования высокого уровня.

Основные понятия языков программирования. Развитие языков программирования. Структуры и типы данных языка программирования. Трансляция. Компиляция и интерпретация. Эволюция и классификация языков программирования.

Язык программирования (Я.П.) — формальная знаковая система, предназначенная для описания алгоритмов в форме, которая удобна для исполнителя (например, компьютера). Язык программирования определяет набор лексических, синтаксических и семантических правил, используемых при составлении компьютерной программы. Он позволяет программисту точно определить то, на какие события будет реагировать компьютер, как будут храниться и передаваться данные, а также какие именно действия следует выполнять над этими данными при различных обстоятельствах.

Если Я.П. ориентирован на конкретный тип процессора и учитывает его особенности, то он называется языком программирования низкого уровня.

К таким языкам относят язык Ассемблер, который представляет каждую команду машинного кода, но не в виде чисел, а с помощью символьных условных обозначений, называемых мнемониками (конкретной компьютерной архитектуре соответствует свой язык ассемблера).

Языки программирования высокого уровня значительно ближе и понятнее человеку, чем компьютеру (особенности конкретных компьютерных архитектур в них не учитываются).

Эволюция языков программирования. Я.П. делят на пять поколений:

Поколение 1. Входят языки, созданные в начале 50-х гг. (машиннозависимые языки, язык Ассемблер).

Поколение 2. Конец 50-х-начало 60-х гг. (Символический ассемблер, в котором появилось понятие переменной. Возросли скорость разработки и надежность программ).

Поколение 3. 60-е гг. – 70-е гг. (Появились универсальные языки высокого уровня, с их помощью удается решать задачи из любых областей. Качества таких языков: относительная простота, независимость от конкретного компьютера, возможность использования мощных синтаксических конструкций. Результат – повышение производительности труда программистов.) (Кобол – для экономической области, Pascal – универсальный ЯВУ (Н.Вирт), Basic

Поколение 4. Начало 70-х гг.- по настоящее время. (Языки 4-го поколения предназначены для реализации крупных проектов, повышения их надежности и скорости создания. – Проблемно-ориентированные языки). (С++-объектно-ориентированный С, Java – начало 90-х гг. – компиляция в платформу-независимый байт-код)

Поколение 5. Середина 90-х гг. (Системы автоматического создания прикладных программ с помощью визуальных средств разработки, без знания программирования. Языки программирования для Интернета – скрипт-языки – HTML, Perl, PHP, XML).

Одним из важнейших признаков классификации языков программирования является принадлежность их к одному из стилей, основными из которых являются следующие: процедурный, функциональный, логический и объектно-ориентированный.

Процедурное (императивное) программирование является отражением архитектуры традиционных ЭВМ, которая была предложена фон Нейманом в 40-х годах. Теоретической моделью процедурного программирования служит алгоритмическая система под названием «машина Тьюринга».

Программа на процедурном языке программирования состоит из последовательности операторов (инструкций), задающих процедуру решения задачи. Основным является оператор присваивания, служащий для изменения содержимого областей памяти. Концепция памяти как хранилища значений, содержимое которого может обновляться операторами программы, является фундаментальной в императивном программировании.

Сущность функционального (аппликативного) программирования определена А. П. Ершовым как «... способ составления программ, в которых единственным действием является вызов функции, единственным способом расчленения программы на части является введение имени для функции, а единственным правилом композиции — оператор суперпозиции функции. Никаких ячеек памяти, ни операторов присваивания, ни циклов, ни, тем более, блок-схем, ни передачи управления».

Роль основной конструкции в функциональных языках играет выражение: К выражениям относятся скалярные константы, структурированные объекты, функции, тела функций и вызовы функций. Функция трактуется как однозначное отображение из X в X , где X — множество выражений.

Новую область — логическое, или реляционное программирование, — открыло появление языка PROLOG (Пролог) (PROgramming in LOGic — программирование в терминах логики). Этот язык был создан французским ученым А. Кольмероэ в 1973 году. В настоящее время известны и другие языки, однако наиболее развитым и распространенным языком логического программирования является именно Пролог. Так, имеется свыше 15 различных его реализаций на ПЭВМ. Языки логического программирования, в особенности Пролог, широко используются в системах искусственного интеллекта, рассматриваемых в данном учебном пособии.

Центральным понятием в логическом программировании является отношение. Программа представляет собой совокупность определений отношений между объектами (в терминах условий или ограничений) и цели (запроса). Процесс выполнения программы трактуется как процесс общезначимости логической формулы, построенной из программы по правилам, установленным семантикой используемого языка. Результат вычисления является побочным продуктом этого процесса. В реляционном программировании нужно только специфицировать факты, на которых алгоритм основывается, а не определять последовательность шагов, которые требуется выполнить. Это свидетельствует о декларативности языка логического программирования. Она метко выражена в формуле Р. Ковальского: «алгоритм = логика + управление».

Объектно-ориентированное программирование представляет собой метод программирования, который весьма близко напоминает наше поведение. Оно является естественной эволюцией более ранних нововведений в разработке языков программирования. Объектно-ориентированное программирование является более структурным, чем все предыдущие разработки, касающиеся структурного программирования. Оно также является более модульным и более абстрактным, чем предыдущие попытки абстрагирования данных и переноса деталей программирования на внутренний уровень.

Для перевода программы, написанной на языке высокого уровня, в соответствующую машинную программу используются языковые процессоры. Различают два вида языковых процессоров: интерпретаторы и трансляторы.

Интерпретатор — это программа, которая получает исходную программу и по мере распознавания конструкций входного языка реализует действия, описываемые этими конструкциями.

Транслятор — это программа, которая принимает исходную программу и порождает на своем выходе программу, записываемую на объектном языке программирования (объектную программу). В частном случае объектным может служить машинный язык, и в этом случае полученную на выходе транслятора программу можно сразу же выполнить на ЭВМ. В общем случае объектный язык необязательно должен быть машинным или близким к нему (автокодом). В качестве объектного языка может служить и некоторый промежуточный язык.

Для промежуточного языка может быть использован другой транслятор или интерпретатор — с промежуточного языка на машинный. Транслятор, использующий в качестве входного языка, близкий к машинному (автокод или язык Ассемблера) традиционно называют Ассемблером.

Транслятор с языка высокого уровня называют компилятором.

Литература: [1], с. 115-129; [2], с. 140-178.

Тема 7. Базы данных.

Базы данных. Системы управления базами данных и базами знаний. Объекты баз данных. Основные операции с данными. Назначение и основы использования систем искусственного интеллекта; базы знаний, экспертные системы, искусственный интеллект.

В настоящее время успешное функционирование различных фирм, организаций и предприятий просто не возможно без развитой информационной системы, которая позволяет автоматизировать сбор и обработку данных. Обычно для хранения и доступа к данным, содержащим сведения о некоторой предметной области, создается база данных.

База данных (БД) — именованная совокупность данных, отражающая состояние объектов и их отношений в рассматриваемой предметной области.

Под предметной областью принято понимать некоторую область человеческой деятельности или область реального мира, подлежащих изучению для организации управления и автоматизации, например, предприятие, вуз и т.д.

Система управления базами данных (СУБД) — совокупность языковых и программных средств, предназначенных для создания, наполнения, обновления и удаления баз данных.

Система специальным образом организованных данных — баз данных, программных, технических, языковых, организационно-методических средств, предназначенных для обеспечения централизованного накопления и коллективного многоцелевого использования называется банком данных.

Основополагающими понятиями в концепции баз данных являются обобщенные категории «данные» и «модель данных».

Понятие «данные» в концепции баз данных — это набор конкретных значений, параметров, характеризующих объект, условие, ситуацию или любые другие факторы. Модель данных — это некоторая абстракция, которая, будучи приложима к конкретным данным, позволяет пользователям и разработчикам трактовать их уже как информацию, то есть сведения, содержащие не только данные, но и взаимосвязь между ними.

С помощью модели данных могут быть представлены объекты предметной области и взаимосвязи между ними. В зависимости от вида организации данных различают следующие важнейшие модели БД:

- иерархическую
- сетевую
- реляционную
- объектно-ориентированную

Реляционная БД получила свое название от английского термина relation (отношение). Была предложена в 70-м году сотрудником фирмы IBM Эдгаром Коддом. Реляционная БД представляет собой совокупность таблиц, связанных отношениями. Достоинствами реляционной модели данных являются простота, гибкость структуры. Кроме того ее удобно реализовывать на компьютере. Большинство современных БД для персональных компьютеров являются реляционными.

Коммерческие системы на основе реляционной модели данных начали появляться в конце 70-х – начале 80-х годов. В настоящее время существует несколько сотен типов различных РСУБД как для мейнфреймов, так и для микрокомпьютеров, хотя многие из них не полностью удовлетворяют точному определению реляционной модели данных. Примерами РСУБД для персональных компьютеров являются СУБД Access и FoxPro фирмы Microsoft, Paradox и Visual dBase фирмы Borland, а также R:Base фирмы Microsim.

Благодаря популярности реляционной модели многие нереляционные системы теперь обеспечиваются реляционным пользовательским интерфейсом, независимо от используемой базовой модели.

Реляционная модель основана на математическом понятии отношения, физическим представлением которого является таблица. Атрибут - это поименованный столбец отношения.

Домен – это набор допустимых значений для одного или нескольких атрибутов.

Элементами отношения являются кортежи, или строки, таблицы. Кортеж – это строка отношения. Кортежи могут располагаться в любом порядке, при этом отношение будет оставаться тем же самым, а значит, и иметь тот же смысл.

Описание структуры отношения вместе со спецификацией доменов и любыми другими ограничениями возможных значений атрибутов иногда называют его заголовком (или содержанием (intension)). Обычно оно является фиксированным, до тех пор пока смысл отношения не изменяется за счет добавления в него дополнительных атрибутов. Кортежи называются расширением (extension), состоянием (state) или телом отношения, которое постоянно меняется.

Степень отношения определяется количеством атрибутов, которое оно содержит.

Отношение только с одним атрибутом имеет степень 1 и называется унарным (unary) отношением (или 1-арным кортежем). Отношение с двумя атрибутами называется бинарным (binary), отношение с тремя атрибутами – тернарным (ternary), а для отношений с большим количеством атрибутов используется термин n-арный (n-ary). Определение степени отношения является частью заголовка отношения.

Количество содержащихся в отношении кортежей называется кардинальностью отношения. Эта характеристика меняется при каждом добавлении или удалении кортежей. Кардинальность является свойством тела отношения и определяется текущим состоянием отношения в произвольно взятый момент.

Отношение обладает следующими характеристиками:

- оно имеет имя, которое отличается от имен всех других отношений;
- каждая ячейка отношения содержит только атомарное (неделимое) значение;
- каждый атрибут имеет уникальное имя;
- значения атрибута берутся из одного и того же домена;
- порядок следования атрибутов не имеет никакого значения;
- каждый кортеж является уникальным, т.е. дубликатов кортежей быть не может;
- теоретически порядок следования кортежей в отношении не имеет никакого значения. (Однако практически этот порядок может существенно повлиять на эффективность доступа к ним.)

Записи в таблице могут зависеть от одной или нескольких записей другой таблицы. Такие отношения между таблицами называются связями. Связь определяется следующим образом: поле или несколько полей одной таблицы, называемое внешним ключом, ссылается на первичный ключ другой таблицы.

Типы связей. Существует три типа связей между таблицами.

Один к одному — каждая запись родительской таблицы связана только с одной записью дочерней. Такая связь встречается на практике намного реже, чем отношение один ко многим и реализуется путем определения уникального внешнего ключа. Связь один к одному используют, если не хотят, чтобы таблица «распухала» от большого числа полей. Базы данных, в состав которых входят таблицы с такой связью не могут считаться полностью нормализованными.

Один ко многим — каждая запись родительской таблицы связана с одной или несколькими записями дочерней. Например, один клиент может сделать несколько заказов, однако несколько клиентов не могут сделать один заказ. Связь один ко многим является самой распространенной для реляционных баз данных.

Многие ко многим — несколько записей одной таблицы связаны с несколькими записями другой. Например, один автор может написать несколько книг и несколько авторов — одну книгу. В случае такой связи в общем случае невозможно определить, какая запись одной таблицы соответствует выбранной записи другой таблицы, что делает неосуществимой физическую (на уровне индексов и триггеров) реализацию такой связи между соответствующими таблицами. Поэтому перед переходом к физической модели все связи "многие ко многим" должны быть переопределены (некоторые CASE-средства, если таковые используются при проектировании данных, делают это автоматически). Подобная связь между двумя таблицами реализуется путем создания третьей таблицы и реализации связи типа «один ко многим» каждой из имеющихся таблиц с промежуточной таблицей.

Литература: [1], с. 5-19; [2], с. 135-156.

Тема 8. Программное обеспечение и технологии программирования.

Системное ПО (System Software) – это программы и комплексы программ, являющиеся общими для всех, кто совместно использует технические средства компьютера, и применяемые как для автоматизации разработки (создания) новых программ, так и для организации выполнения программ существующих.

Прикладные программы предназначены для того, чтобы обеспечить применение вычислительной техники в различных сферах деятельности человека.

Инструментальное программное обеспечение входит в состав системного ПО, к нему относят среды и системы программирования.

Система программирования — это система для разработки новых программ на конкретном языке программирования.

Современные системы программирования обычно предоставляют пользователям мощные и удобные средства разработки программ. В них входят:

- компилятор или интерпретатор;
- интегрированная среда разработки;
- средства создания и редактирования текстов программ;
- обширные библиотеки стандартных программ и функций;
- отладочные программы, т.е. программы, помогающие находить и устранять ошибки в программе;
- "дружественная" к пользователю диалоговая среда;
- многооконный режим работы;
- мощные графические библиотеки; утилиты для работы с библиотеками
- встроенный ассемблер;
- встроенная справочная служба;
- другие специфические особенности.

Популярные системы программирования — Turbo Basic, Quick Basic, Turbo Pascal, Turbo C.

Структурное программирование — методология разработки программного обеспечения, предложенная в 70-х годах XX века Дейкстрой и разработанная и дополненная Виртом.

В соответствии с данной методологией любая программа представляет собой структуру, построенную из трёх типов базовых конструкций:

- последовательное исполнение — однократное выполнение операций в том порядке, в котором они записаны в тексте программы;
- ветвление — однократное выполнение одной из двух или более операций, в зависимости от выполнения некоторого заданного условия;
- цикл — многократное исполнение одной и той же операции до тех пор, пока выполняется некоторое заданное условие (условие продолжения цикла).

Структурное программирование - это методология программирования, основанная на суждении, что логичность и понятность программы является залогом ее надежности, облегчает и ускоряет разработку программы, ее освоение, сопровождение и дальнейшую ее модификацию.

Характерной чертой структурного программирования является его отказ от использования оператора безусловного перехода `goto` для передачи управления, ограниченное использование глобальных переменных, модульность программ.

Алгоритм и программа должны составляться поэтапно (то есть по шагам).

Сложная задача должна разбиваться на достаточно простые, легко воспринимаемые части, каждая из которых имеет только один вход и только один выход. Логика алгоритма и программы должны опираться на минимальное число достаточно простых базовых управляющих структур.

Фундаментом структурного программирования является теорема о структурировании. Эта теорема устанавливает, что как бы ни была сложна задача, схема ее алгоритма может быть представлена с использованием ограниченного числа базисных управляющих структур. Эти элементарные структуры могут соединяться между собой, образуя более сложные структуры, по тем же самым элементарным схемам. Базисными элементарными структурами являются следующие:

Базисная структура типа "следование". Имеет следующий смысл: "Выполнить S"

Базисная структура типа "ветвление". Имеет следующий смысл: "Если P то выполнить S1 иначе выполнить S2"

Базисная структура типа "цикл с предусловием". Имеет следующий смысл: "Пока P выполнять S"

Объектно-ориентированное программирование представляет собой метод программирования, который весьма близко напоминает наше поведение. Оно является естественной эволюцией более ранних нововведений в разработке языков программирования. Объектно-ориентированное программирование является более структурным, чем все предыдущие разработки, касающиеся структурного программирования. Оно также является более модульным и более абстрактным, чем предыдущие попытки абстрагирования данных и переноса деталей программирования на внутренний уровень.

Объекты. Объект – это совокупность свойств (структур данных, характерных для этого объекта), методов их обработки (подпрограмм изменения свойств) и событий, на которые данный объект может реагировать и которые приводят к изменению объекта.

Объекты моделируют характеристики и поведение элементов мира, в котором мы живем. Они являются окончательной абстракцией данных.

Не менее важным является и тот факт, что объекты могут наследовать характеристики и поведение того, что мы называем порождающие, родительские объекты (или предки).

Класс. Объекты могут иметь идентичную структуру и отличаться только значениями свойств. В таких случаях в программе создается новый тип, основанный на единой

структуре объекта. Он называется классом, а каждый конкретный объект, имеющий структуру этого класса, называется экземпляром класса.

Объектно-ориентированный язык программирования характеризуется тремя основными свойствами:

1. Инкапсуляция.
2. Наследование.
3. Полиморфизм.

Наследование. Наследованием называется возможность порождать один класс от другого с сохранением всех свойств и методов класса-предка (иногда его называют суперклассом) и добавляя, при необходимости, новые свойства и методы. Наследование призвано отобразить такое свойство реального мира, как иерархичность.

Полиморфизмом называют явление, при котором классы-потомки могут изменять реализацию метода класса-предка, сохраняя его сигнатуру (таким образом, сохраняя неизменным интерфейс класса-предка). Это позволяет обрабатывать объекты классов-потомков как однотипные объекты, не смотря на то, что реализация методов у них может различаться.

Инкапсуляция — это принцип, согласно которому любой класс должен рассматриваться как чёрный ящик — пользователь класса должен видеть и использовать только интерфейс (т.е. список декларируемых свойств и методов) класса и не вникать в его внутреннюю реализацию. Этот принцип (теоретически) позволяет минимизировать число связей между классами и, соответственно, упростить независимую реализацию и модификацию классов.

Визуальное программирование.

Технологии объектного, событийного и структурного программирования объединяются в RAD-системах, которые содержат много готовых классов, представленных в виде визуальных компонентов, которые добавляются в программу одним щелчком. Программисту необходимо спроектировать внешний вид окон приложений и определить обработку событий, весь вспомогательный код программа сгенерирует сама.

Программирование (programming) - теоретическая и практическая деятельность, связанная с созданием программ. Решение задач на компьютере включает в себя следующие основные этапы, часть из которых осуществляется без участия компьютера.

1. Постановка задачи:
 - сбор информации о задаче;
 - формулировка условия задачи;
 - определение конечных целей решения задачи;
 - определение формы выдачи результатов;
 - описание данных (их типов, диапазонов величин, структуры и т. п.).
2. Анализ и исследование задачи, модели:
 - анализ существующих аналогов;
 - анализ технических и программных средств;
 - разработка математической модели;
 - разработка структур данных.
3. Разработка алгоритма:
 - выбор метода проектирования алгоритма;
 - выбор формы записи алгоритма (блок-схемы, псевдокод и др.);
 - выбор тестов и метода тестирования;
 - проектирование алгоритма.
4. Программирование:
 - выбор языка программирования;
 - уточнение способов организации данных;
 - запись алгоритма на выбранном языке

программирования.

5. Тестирование и отладка:

- синтаксическая отладка;
- отладка семантики и логической структуры;
- тестовые расчеты и анализ результатов тестирования;
- совершенствование программы.

6. Анализ результатов решения задачи и уточнение в случае необходимости математической модели с повторным выполнением этапов 2-5.

7. Сопровождение программы:

- доработка программы для решения конкретных задач;
- составление документации к решенной задаче, к математической модели, к алгоритму, к программе, к набору тестов, к использованию.

Литература: [1], с. 150-209.

Тема 9. Локальные и глобальные сети ЭВМ.

Если два или более компьютера соединены каким-либо способом, обеспечивающим быструю передачу информации между ними, то они образуют систему, для которой принято название компьютерная сеть (сеть ЭВМ).

На сегодняшний день в мире существуют сотни миллионов компьютеров и большинство из них объединены в сети от малых локальных сетей в офисах до глобальных сетей, например, Интернет. Всемирная тенденция к объединению компьютеров в сети обусловлена следующими важными причинами.

1. Ускорение передачи информации.
2. Разделение данных.
3. Разделение периферийных устройств.
4. Разделение времени процессора.
5. Разделение программных средств.
6. Объединение запоминающих устройств.
7. Объединение вычислительной мощности процессоров.
8. Дистанционный доступ к уникальному оборудованию.

Очевидно, что компьютерная сеть кроме компьютеров, должна включать в себя какие-то технические средства для передачи информации. Принципиально важными среди них являются:

- система связи;
- сетевое программное обеспечение.

Локальная сеть (локальная вычислительная сеть, ЛВС, Local Area Network, LAN) - ограниченная группа компьютеров, которые связываются с помощью системы связи, специально созданной для сети. Обычно локальная сеть интегрирует компьютеры в географически ограниченной (локальной) области, например в офисе или здании. Прокладывается специализированная кабельная система, и положение возможных точек подключения пользователей ограничено этой кабельной системой.

Для локальных сетей характерны

- небольшие количества компьютеров в сети (от 2 до 10000);
- относительно небольшие расстояния (менее 100 км);
- высокие скорости передачи (от 1 до 1000 Мбит/с), предоставляемые отдельному пользователю.

Глобальная сеть (Wide Area Network, WAN) состоит из нескольких локальных, которые связаны вместе. Глобальные сети - очень широкое определение, включая и Интернет и ведомственные сети, использующие свои собственные системы связи.

Характерные признаки глобальной сети:

- преимущественное использование универсальных, а не специализированных систем связи (например, телефонной сети и спутниковых систем связи);
- практически неограниченное количество компьютеров в сети;
- неограниченные расстояния;
- низкие скорости передачи на одного пользователя (менее 1 Мбит/с).

Способ объединения узлов в сеть с помощью соединений называют топологией. Топология зависит не от геометрической формы соединения (прямое или изогнутое), а от взаимного соответствия соединений и узлов. Например, простейшая топология сети из двух компьютеров (двухточечное соединение, соединение точка-точка) считается одной и той же независимо от формы и длины кабеля. Если же к этой сети будет присоединен третий компьютер, топология изменится.

Существует несколько десятков глобальных информационных сетей. Среди них наиболее важное место занимает Интернет. Под этим словом понимают единую сеть, состоящую из множества сетей различного типа, как локальных, так и глобальных, работающую по определенным единым принципам.

Услуги, предоставляемые Интернет, доступны любому оплатившему их пользователю, вне зависимости от его государственной или ведомственной принадлежности. Поэтому по сравнению с другими, более узко специализированными глобальными сетями Интернет приобрел значительно большую популярность. Основные направления развития услуг, предоставляемых Интернет, базируются на быстром доступе к самой разнообразной информации.

Информация в Интернет хранится в компьютерах в виде обычных файлов разнообразных форматов.

Пользователь может получить эту информацию в различных видах:

- в виде исполняемой программы, которую можно установить на свой компьютер и использовать в своей работе;
- в виде файла данных (текст, графика, аудио и видео);
- в виде ответов программы-сервера, запущенной на удаленном компьютере.

Технология доступа к информации, построенная на ссылках обычно называется гипертекстовой. В настоящее время она стала основной в Интернет. Наиболее распространенным ее вариантом является World Wide Web ("всемирная паутина") - распределенная гипертекстовая информационная система. World Wide Web представляет удобный доступ к большинству информационных ресурсов.

Информационные ресурсы Интернет — это вся совокупность информационных технологий и баз данных, доступных при помощи этих технологий и существующих в режиме постоянного обновления. Кроме World Wide Web (WWW) к числу наиболее востребованных относятся:

- поисковые машины;
- система файловых архивов FTP;
- электронная почта;
- непосредственное живое общение пользователей между собой посредством клавиатуры (чат).

Браузеры (browsers) - обозреватели Всемирной Паутины (WWW). Это программы, позволяющие находить и просматривать гипертекстовые документы, опубликованные в Сети, на Вашем компьютере. Примерами таких программ являются: Netscape Navigator, Microsoft Internet Explorer, Ariadna, Opera и т. п.

Литература: [1], с. 210-239; [4], с. 45-76.

Тема 10. Защита информации.

Информационная безопасность и ее составляющие. Методы защиты информации. Организационные меры защиты информации. Антивирусные средства. Классификация и характеристики компьютерных вирусов. Методы защиты от компьютерных вирусов.

Под информационной безопасностью понимается защищенность информации и поддерживающей инфраструктуры от случайных или преднамеренных воздействий естественного или искусственного характера, которые могут нанести неприемлемый ущерб субъектам информационных отношений, в том числе владельцам и пользователям информации и поддерживающей инфраструктуры.

Защита информации - это комплекс мероприятий, направленных на обеспечение информационной безопасности.

Информационная безопасность - многогранная, можно даже сказать, многомерная область деятельности, в которой успех может принести только систематический, комплексный подход.

Спектр интересов субъектов, связанных с использованием информационных систем, можно разделить на следующие категории: обеспечение доступности, целостности и конфиденциальности информационных ресурсов и поддерживающей инфраструктуры.

Обеспечение информационной безопасности КС является непрерывным процессом, целенаправленно проводимым на всех этапах ее жизненного цикла с комплексным применением всех имеющихся методов и средств.

Существующие методы и средства защиты информации можно подразделить на четыре основные группы:

- методы и средства организационной защиты информации;
- методы и средства инженерно-технической защиты информации;
- криптографические методы и средства защиты информации;
- программно-аппаратные методы и средства защиты информации.

Организационные методы защиты информации включают меры, мероприятия и действия, которые должны осуществлять должностные лица в процессе создания и эксплуатации компьютерной системы (КС) для обеспечения заданного уровня безопасности информации. Организационные методы защиты информации тесно связаны с правовым регулированием в области безопасности информации, В соответствии с законами и нормативными актами в министерствах, ведомствах, на предприятиях (независимо от форм собственности) для защиты информации создаются специальные службы безопасности (на практике они могут называться и иначе). Эти службы подчиняются, как правило, руководству учреждения. Руководители служб организуют создание и функционирование систем защиты информации. На организационном уровне решаются следующие задачи обеспечения безопасности информации в КС:

- * организация работ по разработке системы защиты информации;
- * ограничение доступа на объект и к ресурсам КС;
- * разграничение доступа к ресурсам КС;
- * планирование мероприятий;
- * разработка документации;
- * воспитание и обучение обслуживающего персонала и пользователей;
- * сертификация средств защиты информации;
- * лицензирование деятельности по защите информации;
- * аттестация объектов защиты;
- * совершенствование системы защиты информации;
- * оценка эффективности функционирования системы защиты информации;
- * контроль выполнения установленных правил работы в КС.

Организационные методы являются стержнем комплексной системы защиты информации в КС. Только с помощью этих методов возможно объединение на правовой основе технических, программных и криптографических средств защиты информации в единую комплексную систему. Конкретные организационные методы защиты

информации будут приводиться при рассмотрении парирования угроз безопасности информации.

Одной из основных угроз на сегодняшний день компьютерным системам являются компьютерные вирусы — это разновидность самовоспроизводящихся компьютерных программ, которые распространяются, внедряя себя в исполняемый код других программ или в документы специального формата, содержащие макрокоманды, такие, как MS Word и Excel. Многие вирусы вредят данным на заражённых компьютерах, хотя иногда их единственной целью является лишь заражение как можно большего количества компьютеров. К компьютерным вирусам причисляют все вредоносные программы, такие как сетевые и файловые черви, тройские кони, программы-шпионы.

В настоящее время не существует единой системы классификации и именования вирусов (хотя попытка создать стандарт была предпринята на встрече CARO в 1991 году). Принято разделять вирусы по поражаемым объектам (файловые вирусы, загрузочные вирусы, скриптовые вирусы, сетевые черви), по поражаемым операционным системам и платформам (DOS, Windows, Unix, Linux, Java и другие), по технологиям, используемым вирусом (полиморфные вирусы, стелс-вирусы), по языку на котором написан вирус (ассемблер, высокоуровневый язык программирования, скриптовый язык и др.).

Сейчас основной канал распространения вирусов — это электронная почта. Так же распространена рассылка ссылок на фото, музыку либо программы, в действительности являющиеся вирусами, по ICQ и другим IM, и по электронной почте. Возможно так же заражение через странички Интернет. В этом случае используются уязвимости ПО установленного на компьютере пользователя, либо уязвимости в ПО владельца сайта (это опаснее всего, так как заражению подвергаются добропорядочные сайты с большим потоком посетителей). Хакеры и спамеры используют зараженные компьютеры пользователей для рассылки спама или DDoS-атак.

Массовое распространение вирусов, серьезность последствий их воздействия на ресурсы КС вызвали необходимость разработки и использования специальных антивирусных средств и методов их применения. Антивирусные средства применяются для решения следующих задач:

- * обнаружение вирусов в КС;
- * блокирование работы программ-вирусов;
- * устранение последствий воздействия вирусов.

Обнаружение вирусов желательно осуществлять на стадии их внедрения или, по крайней мере, до начала осуществления деструктивных функций вирусов. Необходимо отметить, что не существует антивирусных средств, гарантирующих обнаружение всех возможных вирусов.

При обнаружении вируса необходимо сразу же прекратить работу программы-вируса, чтобы минимизировать ущерб от его воздействия на систему.

Устранение последствий воздействия вирусов ведется в двух направлениях:

- * удаление вирусов;
- * восстановление (при необходимости) файлов, областей памяти.

Восстановление системы зависит от типа вируса, а также от момента времени обнаружения вируса по отношению к началу Деструктивных действий. Восстановление информации без использования дублирующей информации может быть невыполнимым, если вирусы при внедрении не сохраняют информацию, на место которой они помещаются в память, а также, если деструктивные действия уже начались, и они предусматривают изменения информации.

Для борьбы с вирусами используются программные и аппаратно-программные средства, которые применяются в определенной последовательности и комбинации, образуя методы борьбы с вирусами. Можно выделить методы обнаружения вирусов и методы удаления вирусов.

Литература: [3], с. 5-19; [5], с. 135-156.

Тема 11. Компьютерный практикум.

Офисное программное обеспечение предназначено для того, чтобы обеспечить применение вычислительной техники в различных сферах деятельности человека. Поэтому этот класс программ представляет наибольший интерес для массового пользователя компьютеров.

Из-за огромного разнообразия прикладного ПО существует множество вариантов его классификации. Рассмотрим наиболее общую классификацию прикладных программ. Разделим данное ПО на 2 больших класса: ПС общего назначения, ПС профессионального уровня.

Каждый класс разбивается на многочисленные подклассы, которые мы сейчас и попытаемся достаточно широко охватить:

Средства обработки текстовой информации. Несмотря на широкие возможности использования компьютеров для обработки самой разной информации, самыми популярными по-прежнему остаются программы, предназначенные для работы с текстом.

Программы, предназначенные для обработки текстовой информации называют текстовыми редакторами.

Все многообразие современных текстовых редакторов условно можно разбить на три основные группы:

К первой относятся простейшие текстовые редакторы, обладающие минимумом возможностей и способные работать с документами в обычном текстовом формате TXT, который, как известно, при всей своей простоте и всеобщей поддержке совершенно не позволяет более или менее прилично форматировать текст.

Промежуточный класс текстовых редакторов включает в себя достаточно широкие возможности по части оформления документов. Они работают со всеми стандартными текстовыми файлами (TXT, RTF, DOC). К таким программам можно отнести Microsoft Works, Лексикон.

К третьей группе относятся мощные текстовые процессоры, такие, как Microsoft Word или StarOffice Writer. Они выполняют практически все операции с текстом. Большинство пользователей использует именно эти редакторы в повседневной работе, правда, подавляющее большинство их возможностей практически не используется.

Средства табличной обработки информации. Табличное представление данных имеет свои особенности. Многие виды данных намного удобнее хранить и обрабатывать в табличной форме, особенно числовые. Автоматизация табличных расчетов во много раз повышает эффективность и качество работы. Компьютерные программы, предназначенные для хранения и обработки данных, представленных в табличном виде, называют электронными таблицами или табличными процессорами.

Области применения электронных таблиц. Появление исторически совпадает с началом распространения персональных компьютеров. Первая программа для работы с ЭТ - табличный процессор, была создана в 1979 г., предназначалась для компьютеров типа Apple II и называлась VisiCalc. В 1982 г. появляется знаменитый табличный процессор Lotus 1-2-3, предназначенный для IBM PC. Lotus объединял в себе вычислительные возможности деловую графику и функции реляционной СУБД. Популярность табличных процессоров росла очень быстро. Появлялись новые программные продукты этого класса: Multiplan, Quattro Pro, SuperCalc и др. Одним из самых популярных табличных процессоров сегодня является MS Excel, входящий в состав пакета Microsoft Office. Электронные таблицы позволяют решать целый комплекс задач.

Литература: [1], с. 70-95; [2], с. 157-182

1. Основная литература:

1. Симонович, С. В. Информатика. Базовый курс. 2-е издание: учебное пособие для вузов. – СПб.: Питер, 2007. – 640 с.
2. Макарова, Н. В. Информатика: учебник для вузов. – М.: Финансы и статистика, 2006. – 679 с.
3. Акулов, О. А. Информатика: базовый курс. Учебник для вузов. – М.: Омега-Л, 2005. – 590 с.
4. Острейковский, В. А. Информатика: учебник для технических специальностей вуза. – М.: Высшая школа, 2005. – 653 с.

2. Дополнительная литература:

5. Фаронов В.В. Delphi .Программирование на языке высокого уровня: учеб. для вузов по напр. «Информатика и вычисл. техника». - СПб.: Питер, 2007.- 639с.
6. Попов В.Б.Паскаль и Дельфи: учеб. курс. – СПб.: Питер, 2005. – 575 с.

Перечень вопросов для подготовки к экзамену по дисциплине «Информатика»

1 семестр

1. Информатика. Предмет информатики. Основные задачи информатики.
2. Понятие информации, свойства информации.
3. Измерение информации, количество и качество информации.
4. Меры измерения количества информации.
6. Формы и способы представления информации.
7. Информация и информационные технологии.
8. Сигналы, кодирование сигналов.
9. Кодирование информации.
10. Системы счисления. Виды систем счисления.
11. Правила перевода систем счисления.
12. Состав и назначение основных элементов персонального компьютера.
13. Периферийные устройства.
14. Архитектура ЭВМ.
15. Центральный процессор, системные шины, их основные характеристики.
16. Запоминающие устройства: классификация, принцип работы, основные характеристики.
17. Системная память: ОЗУ, ПЗУ, кэш.
18. Внешняя память: винчестер
19. Внешняя память: накопитель на гибких магнитных дисках; накопители на компакт-дисках.
20. Устройства ввода/вывода данных, данных, их разновидности и основные характеристики. Клавиатура. Координатные устройства ввода.
21. Видео- и звуковые адаптеры. Назначение, разновидности и основные характеристики.
22. Сканеры. Основные виды и их характеристики.
23. Принтеры, плоттеры. Основные виды и их характеристики.
24. Мониторы. Основные виды и их характеристики.
25. Программное обеспечение. Классификация.
26. Понятие прикладного программного обеспечения. Классификация.
27. Понятие системного программного обеспечения: назначение, возможности, классификация.
28. Операционные системы, назначение, классификация. Примеры.
29. Организация и средства человеко-машинного интерфейса.
30. Файловая структура. Системы управления файлами.
31. Служебное, вспомогательное программное обеспечение.
32. Алгоритмизация. Понятие алгоритма, свойства алгоритма.
33. Проектирование алгоритмов. Блок-схема алгоритма.
34. Основные типы алгоритмов, их сложность и их использование для решения задач.
35. Языки программирования. История развития. Примеры.
36. Понятия формализации, алгоритмизации, программирования.
37. Программа на языке высокого уровня, типы данных, переменные, выражения.
38. Операторы циклов и ветвления. Примеры.
39. Моделирование как метод познания.
40. Классификация и формы представления моделей.
41. Методы и технологии моделирования.
42. Информационная модель объекта.
43. Основные понятия языков программирования.

44. История развития и характеристика языка программирования Turbo Pascal.
45. Структуры и типы данных языка программирования Turbo Pascal.
46. Трансляция. Компиляция и интерпретация.
47. Эволюция и классификация языков программирования.
48. Базы данных. Основные понятия.
49. Основные модели баз данных.
50. Системы управления базами данных.
51. Классификация баз данных.
52. Основные понятия реляционных баз данных.

2 семестр

1. Программное обеспечение ЭВМ. Понятие, классификация, назначение
2. Системное программное обеспечение. Понятие, классификация.
3. Операционные системы. Понятие, функции, виды.
4. Понятие о структурном программировании.
5. Объектно-ориентированное программирование.
6. Системы программирования.
7. Этапы разработки программного обеспечения.
8. Текстовые редакторы. Понятие, назначение, пример.
9. Электронные таблицы. Понятие, назначение, пример.
10. Графические редакторы. Понятие, виды, назначение.
11. Антивирусные средства защиты информации. Понятие, виды.
12. Компьютерные вирусы. Понятие, классификация. Антивирусная защита.
13. Средства защиты информации от потерь и сбоев оборудования.
14. Топология сетей.
15. Глобальные компьютерные сети. Понятие, назначение. Интернет.
16. Программы-браузеры. Понятие, назначение, пример.
17. Методы и средства поиска информации в Интернете.
18. Протоколы Интернет. Характеристика, назначение.
19. Описать интерфейс текстового процессора M. Word.
20. Опишите интерфейс MS Excel. Что может быть содержимым ячеек MS Excel. Какие задачи позволяют решать электронные таблицы?
21. В чем разница абсолютной и относительной адресации, что такое смешанная адресация. Приведите примеры использования абсолютной и относительной адресации в MS Excel.
22. Опишите процесс создания автоматического оглавления и документа слияния в M. Word.
23. Базы данных. Системы управления базами данных и базами знаний.
24. Объекты баз данных. Основные операции с данными.
25. Назначение и основы использования систем искусственного интеллекта; базы знаний, экспертные системы, искусственный интеллект.
26. Локальные и глобальные сети ЭВМ.
27. Основы компьютерной коммуникации. Принципы построения сетей.
28. Опишите модель OSI.
29. Компьютерные коммуникации и коммуникационное оборудование.
30. Программы для работы в сети Интернет
31. Основы защиты информации и сведений, составляющих государственную тайну.
32. Информационная безопасность и ее составляющие
33. Методы защиты информации. Организационные меры защиты информации.
34. Классификация и характеристики компьютерных вирусов. Методы защиты от компьютерных вирусов.

35. Понятие системного программного обеспечения: назначение, возможности, структура; операционные системы.
36. Основы машинной графики. Системы компьютерной графики и анимации
37. Анализ и обработка данных электронной таблицы.
38. Принцип работы с формулами и функциями и анализ данных в электронной таблице.
39. Для чего используется СУБД MS Access? Какие способы создания таблиц в СУБД MS Access вы знаете?
40. Дайте определение понятию «ключевое поле таблицы». Объясните последовательность добавления и удаления полей в таблице.
41. Что описывает схема данных в MS Access? Перечислите и охарактеризуйте типы связей в реляционных базах данных.
42. Какие виды запросов существуют в СУБД? Какая последовательность создания запросов в MS Access?
43. Объясните последовательность работы с Конструктором в MS Access. Как сформировать условие отбора?
44. Объясните последовательность работы запроса на удаление? Объясните последовательность работы запроса добавления?
45. Дайте определению понятию «поисковая система»? Перечислите известные Вам поисковые системы?
46. Дайте определение понятию «электронная почта»? Расскажите об адресации в системе электронной почты. Объясните последовательность настройки электронной почты в Интернет?
47. Перечислите основные теги языка HTML? Как оформить гиперссылки на Web-странице?