

ГОСУДАРСТВЕННЫЙ КОМИТЕТ СВЯЗИ, ИНФОРМАТИЗАЦИИ И
ТЕЛЕКОММУНИКАЦИОННЫХ ТЕХНОЛОГИЙ РЕСПУБЛИКИ УЗБЕКИСТАН

ТАШКЕНТСКИЙ УНИВЕРСИТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

Факультет информационных технологий

МЕТОДИЧЕСКИЕ УКАЗАНИЯ

ПО ВЫПОЛНЕНИЮ ЛАБОРАТОРНЫХ РАБОТ

ПО ДИСЦИПЛИНЕ

«КОМПЬЮТЕРНЫЕ СИСТЕМЫ И СЕТИ»

Ташкент 2013

Методические указания к _____ лабораторных работ по
дисциплине «Компьютерные системы и сети»

Мирюсупов З.З., Назаров А.И. Ташкент. 2013.- 49с.

Данные методические указания содержат лабораторные работы по основным разделам курса «Компьютерные системы и сети» с кратким изложением аппаратных и программных средств для построения локальной сети.

Включены задания, охватывающие установку компонентов компьютера и операционной системы, методы передачи данных и доступа к среде, стандарты построения локальной сети Ethernet, протоколы

модели OSI и межсетевого взаимодействия, а также определение оптимального быстродействия процессора и оптимальной топологии компьютерной сети.

Методическое указание предназначено для студентов направления: 5521900 - «Информатика и информационные технологии»,
5523500 – «Информационная безопасность»,
5523600 – «Электронная коммерция»,
5811100 – сервис предприятия «Электронная и компьютерная техника»,
5522200 – «Телекоммуникация».

Методические указания рассмотрены и утверждены на заседании кафедры «Компьютерные системы» протокол №_____ от_____2013г.

Напечатано по решению научно-методического совета факультета информационных технологий №_____ от_____2013г.

Рецензенты: д.т.н., проф. Мусаев М.М.

Ташкентский Университет Информационных Технологий, 2013

В

Методические указания представляют собой указания для получения базовых знаний по организации и функционированию компьютерных локальных сетей. Даны общие понятия компьютерных локальных сетей, их структуры, сетевых компонентов. Здесь приведены

виды топологии, используемые для физического соединения компьютеров в сети, методы доступа к каналу связи, физические среды передачи данных. Передача данных в сети рассматривается на базе эталонной базовой модели, разработанной Международной организацией по стандартам взаимодействия открытых сетей. Описываются правила и процедуры передачи данных между компьютерными системами. Приводятся типы сетевого оборудования, их назначение и принципы работы. Описываются протоколы, используемые для организации сетей. Изучаются наиболее популярные сетевые операционные системы, их достоинства и недостатки. Рассматриваются принципы межсетевого взаимодействия.

Лабораторная работа №1.

«Изучение конфигурации ПК и его структуры»

Цель работы: Изучение основных компонентов системного блока и установка материнской платы.

Данная тема рассказывает о материнских платах и их установке. Задание применяемое для проведения практической работы направлено на закрепления навыков по установке материнской платы и проведения анализа материнской платы:

Описание:

- 1.Откройте корпус системного блока.
- 2.Проверьте наличие необходимых крепежных болтов.



3. Для удобства работы положим наш ПК на бок.
4. Найдите специальные болты для крепления материнской платы к корпусу (болты не дающие материнской плате соприкаться с корпусом.) и поместите их в специальные отверстия.



5. Теперь все в блоке СБ подготовлено к инсталляции.



6. Поместите материнскую плату 7. Соединения передней панели в корпус.

На всех материнских платах разъемы для обеспечения функций, выводимых на переднюю панель, группируются в ее нижней правой части. Описание находится в руководстве, поставляемом вместе с материнской платой. Аббревиатуры, напечатанные на плате, для начинающих пользователей будут не очень полезны. Их надо сначала узнать.



Соединения передней панели: HDD LED (светодиодный дисплей жесткого диска),

кнопка включения питания (on/off) и кнопка RESET в верхнем ряду. Внизу: мини-микрофон

- SP, SPK, или SPEAK: выход для динамика. У него 4 штырька.
- RS, RE, RST или RESET: кнопка Reset. Кабель 2-штыревой.
- PWR, PW, PW SW, PS или Power SW: кнопка включения питания. Разъем 2-штыревой.
- PW LED, PWR LED или Power LED: светодиод на передней панели корпуса. Загорается, когда компьютер включен. Имеет 2-штыревой кабель.
- HD, HDD LED: эти два штыря соединяются с кабелем для подключения светодиода, отмечающего работу жесткого диска.

Не беспокойтесь о полярности при подключениях. Кнопки Reset и On/Off будут работать независимо от того, как вы их подключите, а светодиоды

просто не зажгутся, если их подключить в обратной полярности. Если вы слышите, что жесткий диск заработал, а светодиод не загорается, просто переверните разъем.

Окончательная проверка

Перед загрузкой проверьте некоторые насущно необходимые вещи.

- Конфигурацию джамперов на материнской плате: все ли установки для процессора правильны?
- Установки джамперов для дисководов: ведущий/ведомый выполнены правильно?
- Крепко ли сидят в своих разъемах процессор, модули RAM и вставляемые карты?
- Все ли кабели вставлены в разъемы? Крепко ли они там держатся?
- Затянуты ли все винты на вставляемых картах и вставлены ли все зажимы?
- Все ли дисководы закреплены?
- Все ли кабели подключены к дисководам?

8. Подключите питание к вашей материнской плате.

Задание: Проведите внешний анализ материнской платы и заполните таблицу. И укажите все эти элементы на материнской плате.

№	Наименование	Наличие на МП	Описание	Кол-во
1	Cheapsett			
2	Socket			
3	DDR			
4	PCI			
5	AGP			
6	ISA			
7	IDE			
8	Serial ATA			

9	IEEE 1394			
10	USB			
11	PCI Express x16			
12	LAN			
13	Audio			

Пример: МП Asus P5GDC-V Deluxe



№	Наименование	Наличие на МП	Описание	Кол-во
1	Cheapsett	+	915G	1
2	Socket	+	LGA 775	1
3	DDR	+	До 2 GB	4
4	PCI	+	Станд.	3
5	AGP	-	-	1
6	ISA	-	-	0

7	IDE	+	Подключение до 4-х устройств.	2
8	Serial ATA	+	Станд.	4
9	IEEE 1394	+	Станд.	1
10	USB	+	2.0	4
11	PCI Express x16	+	Станд.	1
12	LAN	+	Gigabyte Ethernet	1
13	Audio	+	7.1	1

Контрольные вопросы для опроса :

1. Что такое материнская плата ?
2. Понятие чипсет и сокет ?
3. Интерфейс IEEE 1394 ?
4. Интерфейс PCI Express x16 ?
5. Как установить МП ?

Литература

1. Фигурнов В.Э. IBM PC для пользователя. - М.: Финансы и статистика. 1997.
2. Мячев А.А., Степанов В.И. Справочник. Персональные ЭВМ и микро-ЭВМ. Основы организации. - М.: Радио и связь, 1991.
3. Windows XP Русская версия. Самоучитель по Windows XP. CD. 2005.
4. Microsoft Corporation. Компьютерные сети. Учебный курс / Пер. с англ. - М.: Издательский отдел «Русская Редакция» ТОО «Channel Trading Ltd.». - 1997.

5. Олифер В.Г., Олпфер 11.А. Компьютерные сети. 3-е издание. М.- Л.: Питер, 2006.

Лабораторная работа №2

« УСТАНОВКА СЕТЕВОЙ ОПЕРАЦИОННОЙ СИСТЕМЫ»

Цель занятия: Освоить установку сетевой операционной системы, разобраться с возможностями сетевой печати, выяснить как устанавливаются сетевые приложения, выяснить особенности технологии «клиент – сервер».

Основные теоретические сведения

Операционная система

Операционная система (ОС) – это понятие опирающиеся на определенные традиции, параметры оборудования, размеры программ, реализующих некоторые функции, а так же распределение функций между различными прикладными программами. Считается, что операционная система должна:

- 1) поддерживать связь с оператором в реальном масштабе времени, для подготовки устройств к работе, переопределения конфигурации и изменения состояния системы. Этот интерфейс, как правило, дает возможность оператору запускать программы, прерывать выполнение программ, запрашивать информацию об использовании аппаратуры и о ходе выполнения конкретной программы, сообщать системе о соответствии наборов данных конкретным устройствам и т.д.;
- 2) поддерживать функции, относящиеся к вводу – выводу. В частности, в состав операционной системы входят программы обработки очередей запросов на выполнение ввода – вывода, распределения запросов между

устройствами, обработки прерываний от устройств ввода-вывода, распределения устройств и каналов конкретным программам, а так же наборам данных;

3) поддерживать механизмы мультипрограммирования, способствующие использованию процессоров выполняющимися программами. Иногда частью операционной системы считаются и механизмы управления, которые предоставляют выполняющимся программам системное обслуживание соответствующих уровней, но эти же механизмы могут считаться расширением системы, поскольку реализуют управляющие функции высшего уровня;

4) поддерживать механизмы, обеспечивающие методы доступа к ресурсам вычислительной системы. Например, компоненты считывания и управления входным потоком в пакетном режиме и интерпритации поступающих с терминалов команд в диалоговом режиме позволяют установить связь между программистами и системой для приема запросов от первых на соответствующее обслуживание.

На сегодняшний день точного определения операционной системы не существует, однако чаще всего это комплекс программ, который выполняет следующие задачи:

- управление процессом путем чередования выполнения программ;
- обработка прерываний и синхронизация доступа к ресурсам вычислительной системы;
- управления памятью путем выделения программам во время из выполнения требуемой памяти;
- управление устройствами путем инициализации запросов на ввод вывод, управления очередями и фиксации завершения обменов;
- управление инициализаций программ и обеспечение межпрограммных связей;
- управление данными путем поддержки создания, открытия, закрытия, чтения и обновления файлов.

Поддержка сетевой операционной системой сетевой деятельности довольно сложная

задача. Фактически ею должны быть реализованы все задачи, описанные в семиуровневой модели OSI. В связи с этим, выбирая операционную систему для сетевой среды, необходимо принимать во внимание такую ее особенность, как многозадачность (multitasking).

Многозадачная ОС позволяет выполнять на компьютере более одной задачи одновременно. «Настоящая» многозадачность возможна в многопроцессорной системе. Если задач больше чем процессоров, то необходимо распоряжаться их временем так, чтобы процессоры уделяли каждой задаче некоторое время, пока они не будут все выполнены. Компьютер под управлением такой ОС «выглядит» так, будто он одновременно обрабатывает несколько задач.

Существует два основных типа многозадачности.

- Вытесняющая (preemptive) многозадачность. Основная ее особенность – возможность управления процессором без «согласования» с решаемой в данный момент задачей.
- Невытесняющая (non-preemptive) многозадачность. В этом случае управление процессором никогда не отнимается у задачи. Она сама решает, когда ей освободить процессор. Поэтому программы должны периодически уступать процессор и другие ресурсы системы друг другу.

Для сетевой операционной системы вытесняющая многозадачность представляет особую значимость, так как не позволяет захватить приложению на долгое время процессор и задержать прием или передачу данных в сеть. Примером вытесняющей ОС является UNIX. Настоящая сетевая ОС имеет встроенные средства работы с сетью, такая система кроме вышеперечисленных функций позволяет:

- связывать все компьютеры и периферийные устройства в сеть;
- координировать функции компьютеров и периферийных устройств в сети;
- обеспечивать защищенный режим доступа к сетевым ресурсам.

Компоненты сетевой операционной системы. Сетевая ОС состоит из двух основных компонент:

- СПО, устанавливаемого на компьютерах-клиентах;
- СПО, устанавливаемого на компьютерах - серверах

Клиентское программное обеспечение. В несетевом компьютере ОС, имея запросы от пользователя, всегда обращается только к ресурсам своего компьютера. Если речь идет о сетевом компьютере, запрос может быть адресован к локальным ресурсам компьютера или к ресурсам сервера, т.е. сетевым ресурсам. Для выбора направления адресации запросов в программном обеспечении сетевого компьютера

предусмотрена специальная компонента – редиректор (оболочка – shell) или запросчиком (requester). Редиректор принимает запрос от пользователя и определяет направление запроса. Компьютер пользователя называется клиентом, потому что он формирует запрос к серверу. Запрос распознается редиректором и отправляется на сервер. Редиректор выполняет несколько важных задач. Он должен следить за тем, чтобы обозначения устройств соответствовали определенным сетевым ресурсам. Посылать запросы не только к компьютерам, но и к отдельным периферийным устройствам. Благодаря редиректору пользователь может больше не заботиться о реальном местоположении данных или периферийных устройств. Например, для получения доступа к данным на сетевом компьютере, достаточно набрать букву – обозначение диска, присвоенное местонахождению ресурса, и редиректор выполнит все необходимые действия. Если целью запроса является специализированный компьютер – сервер печати, то вам не нужно указывает его местоположение, а достаточно только указать на один из принтеров, подключенных к этому серверу, и редиректор выполнит все остальные действия сам. Соответствие ресурсов сети и условные обозначения для них определяются для каждого пользователя индивидуально. Принципиально возможно, используя оболочку shell создать пользователя «свою» среду, настроенную для решения конкретных задач.

Серверное программное обеспечение. Собственно, именно оно позволяет всем сетевым компьютерам совместно использовать данные сервера и его периферийные устройства, включая принтеры, плоттеры и диски.

Существуют сетевые ОС, у которых все компьютеры содержат программное обеспечение и клиента, и сервера. Примером может служить Windows NT и, наоборот, Novell - сетевая ОС, у которой клиентская часть отделена от серверной.

Сетевая ОС не только предоставляет возможность доступа к совместно используемому ресурсу, но и определяет порядок их совместного использования. При этом имеется в виду следующее **возможности по управлению совместно используемым ресурсам:**

- предоставление различным пользователям разного уровня доступа к ресурсам;
координацию доступа к ресурсам, - чтобы исключить ситуацию, когда два компьютера пытаются одновременно получить доступ к одному ресурсу.

Другой аспект деятельности сетевой ОС связан с **управлением правами доступа**, т.е. определять, кто может работать с ресурсами сети. Для этого администратор сети имеет возможность:

- 1) добавлять в список пользователей сети новых пользователей;
- 2) предоставлять привилегии отдельным пользователям сети или снимать эти привилегии;
- 3) удалять определенных пользователей из списка пользователей, поддерживаемой сетевой ОС.

Современные сетевые ОС имеют инструментальные средства управления сетью, которые помогают администратору анализировать состояние сети.

Установка Windows NT Server

Установка современных операционных систем производится с помощью специальной программы. На первом этапе с помощью вопросов, которые задает эта программа, уточняются условия установки, затем выбирается нужный набор отдельных компонент (дистрибутив) и генерируется сетевая ОС. Точная и правильная формулировка условий работы ОС – залог ее правильной и устойчивой работы:

- размером сети;
- типом задач, которые сервер будет выполнять в сети;
- типом файловой системы, которую будет использовать сервер;
- идентификацией сервера;
- операционными системами, установленными на сервере;
- распределение дискового пространства сервера.

Идентификация сервера. Программа установки запрашивает следующие данные о сервере и сети: имя сегмента сети (имя домена или рабочей группы); имя сервера.

Сеть, работающая под управлением Windows NT Server, разделяется на отдельные области – домены. Домен – это логическое объединение

нескольких компьютеров с целью упрощения управления ресурсами и администрированием. В таких сетях сервер домена может выполнять задачи связанные с ведением файловой системы, управлять сетевой печатью, отвечать за безопасность и хранение важной информации, осуществлять резервное копирование. Другими словами первый установленный в такой сети сервер – это главный контроллер домена (Primary Domain Controller, PDC). Такой контроллер будет содержать важнейшую информацию о домене и его пользователях, проверять их полномочия и он будет единственным в домене. Другие серверы, устанавливаемые в домене позже, могут выполнять функции резервных контроллеров домена (Backup Domain Controller, BDC). В домене в целях безопасности должен быть хотя бы один BDC. Другие серверы, устанавливаемые в сети, могут выполнять функции серверов файлов, печати и приложений.

Устанавливая сервер, вы должны указать, является ли он главным сервером (PDC) или это резервный контроллер или простой сервер.

Разделы на жестких дисках сервера. Перед установкой сетевой ОС ,необходимо решить как будет использоваться пространство жестких дисков сервера. В любом случае диск разбивается на разделы (Partition). Каждый из разделов может резервироваться для «своих» целей, например, для базы данных или для графических приложений. Очевидно, должен быть раздел для самой сетевой ОС. Размеры этих разделов определяются на основании исходных данных ,по соответствующей ОС или приложению.

Сетевые адаптеры. Сервер, как обязательный элемент имеет в своем составе, по крайней мере, хотя бы один сетевой адаптер, который обязательно должен быть сконфигурирован и, для него, необходимо установить драйвер. Как обязательный шаг здесь потребуется определить протокол. Для Windows NT допускается использование протоколов: TCP/IP; NetBEUI; IPX/SPX. По умолчанию принимается TCP/IP.

Установка TCP/IP. Для установки протокола необходимо задать: адрес, маску подсети и шлюз по умолчанию.

IP - адрес, представляет собой 32 разрядное число. Этот адрес состоит из двух частей: адреса подсети и адреса компьютера (хоста) в подсети.

Маска подсети служит для упрощения поиска маршрута движения информации от компьютера к компьютеру.

Шлюз по умолчанию необходим для определения маршрута движения информации при ее передаче компьютеру другой сети. Если этот параметр не задан, то маршруты передачи ограничены локальной сетью.

Конфигурирование TCP/IP вручную. В этом случае все параметры вы должны определить сами и ввести их в тот момент, когда будет производиться установка драйвера и протокола. Если вы устанавливаете сервер в существующей сети, то при выборе всех этих параметров вы должны проконсультироваться с администратором сети.

Автоматическое конфигурирование. В этом случае Microsoft Windows NT Server поддерживает динамический протокол конфигурирования (Dynamic Host Configuration Protocol, DHCP). Для поддержки этой возможности в сети должен быть сконфигурирован сервер DHCP. При его наличии компьютеры, поддерживающие DHCP, могут запрашивать у него необходимые параметры конфигурации. Если такой сервер вам доступен, то достаточно активировать Enable Automatic DHCP Configuration. Автоматическая конфигурация с использованием DHCP может быть произведено позже с помощью Control Panel в Network.

Системные требования к серверу. Для нормальной работы сервера необходимо точно знать минимальные требования, которым он должен удовлетворять. Обычно эти требования указаны в технической документации к пакету программ сетевой ОС. Все в общем сводиться к определению типа процессора, требуемому дисковому пространству и емкости оперативной памяти.

Установка сетевых служб Windows NT Server

Сетевые службы – это стандартные приложения сетевой ОС, которые обеспечивают минимальный набор сервиса (service). Как всегда в таких случаях, минимальный набор этих служб устанавливается при установке системы по умолчанию. В процессе работы могут потребоваться изменения. Установка и удаление служб напоминает операции с драйверами. Все установки осуществляются в окне Control Panel и Network Settings. Сетевая служба реализована программно, поэтому при добавлении новой службы нужно установить дополнительное программное обеспечение (Add Software) в окне Add Network Software. Выберите службу, которую вы хотите установить и щелкните по кнопке Continue.

При включении компьютера некоторые службы запускаются автоматически (по умолчанию). Другие службы могут быть запущены вручную. Для этого необходимо нужно открыть окно Services в Control Panel. Если в столбце Status открывшегося окна требуемой службы стоит Manual, то для запуска нужно щелкнуть по кнопке Start. Для перевода запуска службы в автоматический режим требуется только воспользоваться кнопкой Startup.

Задание 1. Копирование файлов сетевой ОС .

1. Запустите программу установки сетевой ОС .
2. Прочитав предложенный на экране текст щелкните по кнопке Start Setup.
3. Получив на экране приглашение Welcome, нажмите Enter.
4. Появится приглашение с предложением о режиме установки, выберите быстрый (Express) режим, нажав Enter.
5. Внимательно читайте все сообщения, появляющиеся на экране и сделайте так как указывает система установки. Появится сообщение о том, что для этой имитации диск не нужен (в реальных условиях это может быть не так). Для продолжения установки, нажмите Enter.
6. В ответ на приглашение вставить диск вновь нажмите Enter.
7. Появится новое сообщение о том, что найден контроллер диска фирмы Adaptec. Для продолжения процесса нажмите Enter.
8. На экране появится новое сообщение о возможности установки ОС с компакт-диска или с 3,5” гибких дисков. Установка в лабораторной работе должна производиться с компакт-диска. Для продолжения вновь нажмите Enter.

9. Для размещения ОС система предложит вам создать раздел на жестком диске. Поступайте в соответствии с рекомендациями в Help, нажимая Enter.

10. На следующем шаге вам необходимо выбрать тип файловой системы, сделав выбор нажмите Enter.

11. Операционная система на диске занимает определенный объем, поэтому на следующем шаге проверяется наличие этого места. Если пространства хватает, процесс установки продолжается. В ответ на ваши действия система сообщит о завершении операции переноса файлов на диск. Первый этап работы завершен.

Контрольные вопросы

1. Что используется в компьютере для обмена данными с периферийными устройствами?

2. Что представляет собой внешний интерфейс компьютера и какое название он еще имеет?

3. Какие наиболее распространенные интерфейсы компьютера Вы знаете?

4. Какие внешние порты используются в компьютере для реализации интерфейса с периферийными устройствами?

5. Какие существуют правила подключения периферийных устройств к компьютеру?

6. В чем заключается особенность отключения Flash Memory от USB порта компьютера?

7. Что называется драйвером периферийного устройства?

8. Какими возможностями установки драйверов периферийных устройств располагают операционные системы семейства Windows?

9. Как осуществляется установка драйверов нестандартных периферийных устройств?

Литература

1. Фигурнов В.Э. IBM PC для пользователя. - М.: Финансы и статистика. 1997.
2. Мячев А.А., Степанов В.И. Справочник. Персональные ЭВМ и микро-ЭВМ. Основы организации. - М.: Радио и связь, 1991.
3. Windows XP Русская версия. Самоучитель по Windows XP. CD. 2005.
4. Microsoft Corporation. Компьютерные сети. Учебный курс / Пер. с англ. - М.: Издательский отдел «Русская Редакция» ТОО «Channel Trading Ltd.». - 1997.
5. Олифер В.Г., Олпфер И.А. Компьютерные сети. 3-е издание. М.- Л.: Питер, 2006.

ЛАБОРАТОРНАЯ РАБОТА №3

«Организация соединения одноранговой сети»

Цель:

Краткие теоретические сведения о одноранговых локальных сетях.
Изучить основные требования к аппаратному обеспечению
Изучить установку параметров локальной сети

Задачи:

1. Настройка соединения по локальной сети.
2. Предоставление другим пользователям доступа к файлам и папкам на данном компьютере.
3. Установка и использование сетевого принтера.
4. Настройка совместного использования соединения Интернет.

Краткие теоретические сведения

С точки зрения организации взаимодействия компьютеров, сети делят на одноранговые (Peer-to-Peer Network) и с выделенным сервером (Dedicated Server Network).

Одноранговые сети

Все компьютеры одноранговой сети равноправны. Любой пользователь сети может получить доступ к данным, хранящимся на любом компьютере.

Одноранговые сети могут быть организованы с помощью таких операционных систем, как LANtastic, windows'3.11, Novell Netware Lite. Указанные программы работают как с DOS, так и с windows. Одноранговые сети могут быть организованы также на базе всех современных 32-разрядных операционных систем - windows 9x\ME\2k, windows NT workstation версии, OS/2) и некоторых других.

Достоинства одноранговых сетей:

1. Наиболее просты в установке и эксплуатации.
2. Операционные системы DOS и windows обладают всеми необходимыми функциями, позволяющими строить одноранговую сеть.

Недостатки:

В условиях одноранговых сетей затруднено решение вопросов защиты информации. Поэтому такой способ организации сети используется для сетей с небольшим количеством компьютеров и там, где вопрос защиты данных не является принципиальным.

Иерархические сети

В иерархической сети при установке сети заранее выделяются один или несколько компьютеров, управляющих обменом данных по сети и распределением ресурсов. Такой компьютер называют **сервером**. Любой компьютер, имеющий доступ к услугам сервера называют клиентом сети или рабочей станцией. Сервер в иерархических сетях - это постоянное хранилище разделяемых ресурсов. Сам сервер может быть клиентом только сервера более высокого уровня иерархии. Поэтому иерархические сети иногда называются сетями с выделенным сервером. Серверы обычно представляют собой высокопроизводительные компьютеры, возможно, с несколькими параллельно работающими процессорами, с винчестерами

большой емкости, с высокоскоростной сетевой картой (100 Мбит/с и более).

Иерархическая модель сети является наиболее предпочтительной, т.к. позволяет создать наиболее устойчивую структуру сети и более рационально распределить ресурсы, а также более высокий уровень защиты данных.

К недостаткам иерархической сети, по сравнению с одноранговыми сетями, относятся:

1. Необходимость дополнительной ОС для сервера.
2. Более высокая сложность установки и модернизации сети.
3. Необходимость выделения отдельного компьютера в качестве сервера

Две технологии использования сервера

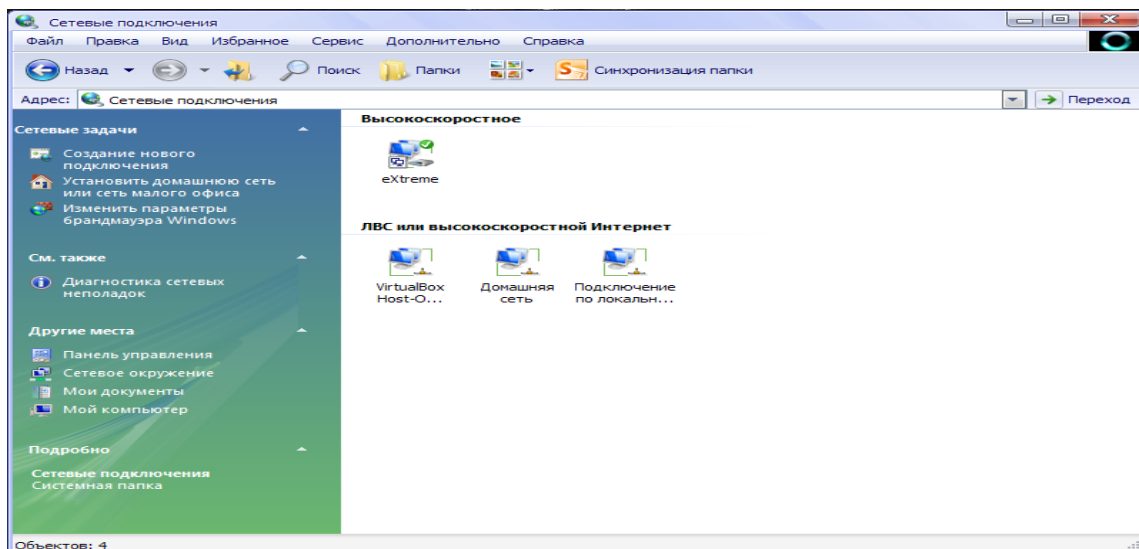
Различают две технологии использования сервера: технологию **файл-сервера** и архитектуру **клиент-сервер**.

В первой модели используется *файловый сервер*, на котором хранится большинство программ и данных. По требованию пользователя ему пересылаются необходимая программа и данные. Обработка информации выполняется на рабочей станции.

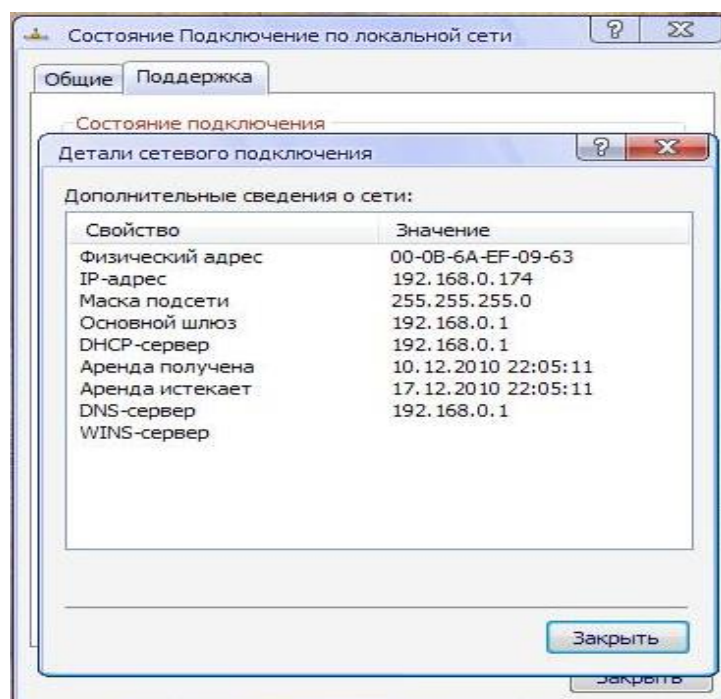
В системах с архитектурой клиент-сервер обмен данными осуществляется между *приложением-клиентом* (front-end) и *приложением-сервером* (back-end). Хранение данных и их обработка производится на мощном сервере, который выполняет также контроль за доступом к ресурсам и данным. Рабочая станция получает только результаты запроса. Разработчики приложений по обработке информации обычно используют эту технологию. Использование больших по объему и сложных приложений привело к развитию многоуровневой, в первую очередь трехуровневой архитектуры с размещением данных на отдельном сервере базы данных (БД). Все обращения к базе данных идут через сервер приложений, где они объединяются.

Соединение через сетевые карты

Достаточно экономичным и не требующим особых технических затрат способом соединения компьютеров является использование сетевых карт. При этом варианте получается небольшая одноранговая сеть (из двух компьютеров). Список необходимого оборудования: две сетевые карты и отрезок кабеля. В зависимости от выбранного типа кабеля



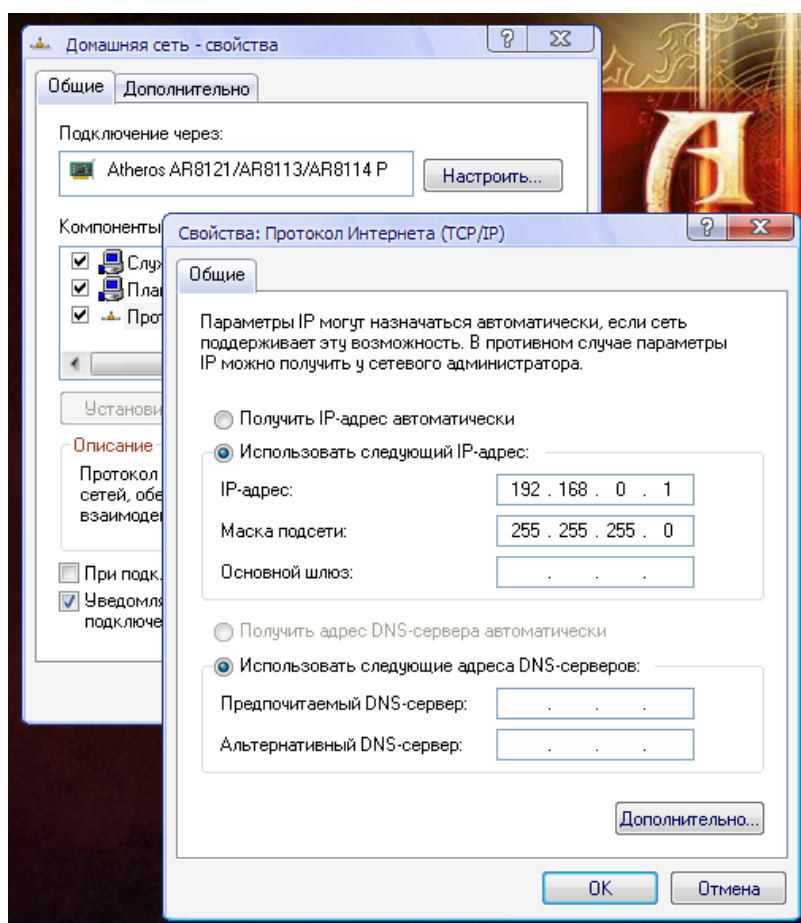
различаются детали реализации подключения



Выполненные действия.

1. После установки сетевых драйверов, автоматически было создано соединение **Подключение по локальной сети 2**. Это соединение было переименовано в **Домашняя сеть**. Соединение **VirtualBox Host-Only Network** было создано при установке виртуальной машины VirtualBox. Также было создано соединение с Интернет через ADSL-модем (Подключение по локальной сети относится к соединению Интернет).

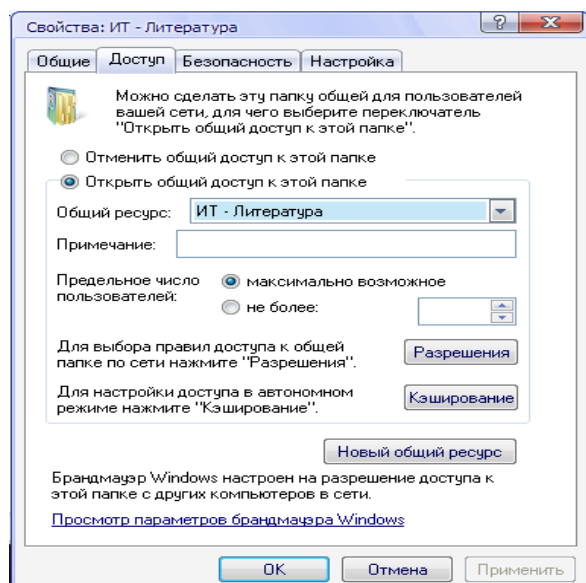
2. Сначала было настроено соединение Домашняя сеть. Ей присвоен статический адрес 192.168.0.1 (данный адрес необходим в дальнейшем для



3. совместного использования Интернет-соединения) и маска подсети 255.255.255.0 .
4. В настройках удаленного соединения с Интернет на вкладке Дополнительно настраивается общий доступ к подключению к Интернету. Для этого устанавливается флажок Разрешить другим пользователям сети использовать подключение к Интернету данного компьютера. В поле Подключение домашней сети указывается название локального соединения, которое будет использоваться. При установке флажка Устанавливать вызов по требованию осуществляется соединение с

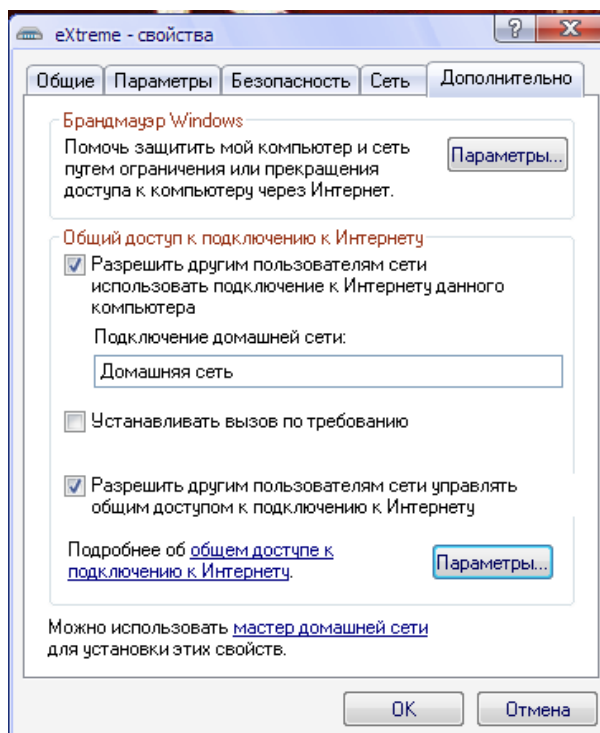
Интернетом при любой попытке пользователя домашней сети использовать Интернет. На компьютере, использующем общий доступ к Интернет, IP-адрес автоматически присваивается DHCP-сервером.

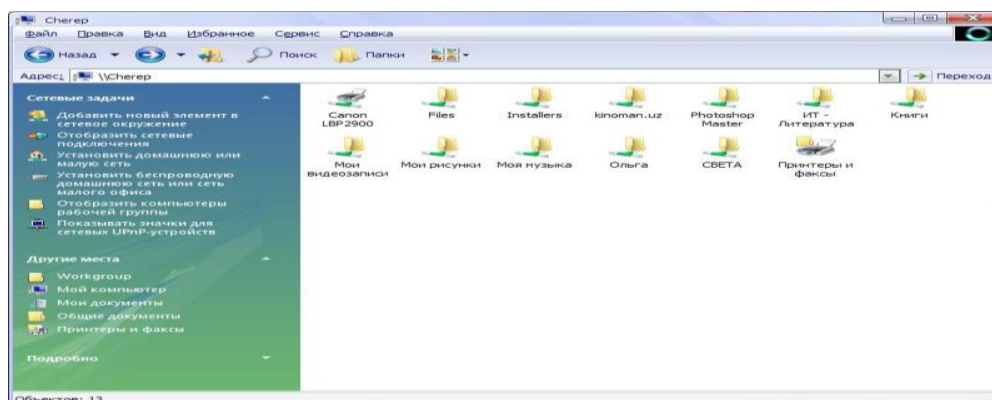
5. На данном компьютере был открыт общий доступ к некоторым файлам и папкам. Доступ открыт только на чтение. Также можно устанавливать



особые разрешения для различных групп и отдельных пользователей сети.

6. Как видно имеется общий доступ к файлам данного компьютера для других пользователей сети. Был предоставлен общий доступ к некоторым папкам данного компьютера.
7. На данном компьютере был открыт общий доступ к принтеру **Canon**

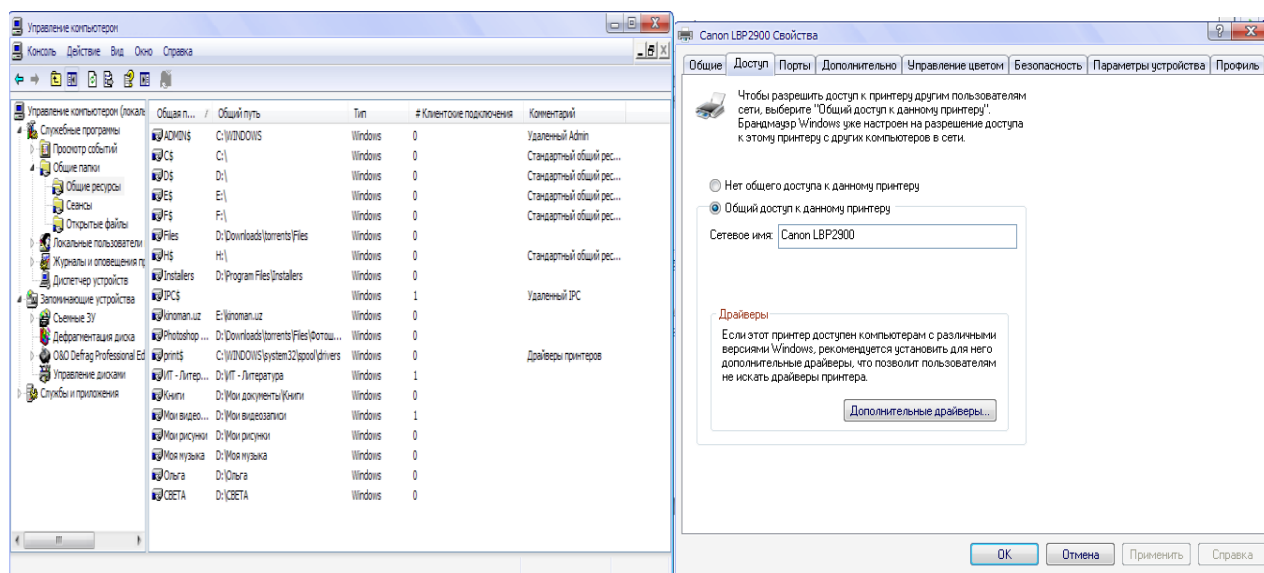




8. LBP2900.

9. На другом компьютере сети, при первой попытке использования данного принтера, сначала установится драйвер принтера из общего ресурса **print\$**. После чего сетевой принтер можно использовать как и локально подключенный.
10. Через оснастку Управление компьютером – Общие папки можно легко управлять общим доступом к папкам и просматривать какие ресурсы используются удаленными пользователями.

Общие ресурсы данного компьютера:



Контрольные вопросы

1. Что такое Сервер и рабочая станция (клиент) ?

3. Что такое ресурсы ?
4. Что такое среда передачи данных ?
5. Какие виды серверов вы знаете ?
6. Что такое Локальная сеть, глобальная сеть, городская сеть ?
9. Основные отличия этих сетей ?
10. Какие ресурсы присутствуют в сетях ?
11. Сеть Peer-TO-Peer и Client-Server ?

Литература

1. Фигурнов В.Э. IBM PC для пользователя. - М.: Финансы и статистика. 1997.
2. Мячев А.А., Степанов В.И. Справочник. Персональные ЭВМ и микро-ЭВМ. Основы организации. - М.: Радио и связь, 1991.
3. Windows XP Русская версия. Самоучитель по Windows XP. CD. 2005.
4. Microsoft Corporation. Компьютерные сети. Учебный курс / Пер. с англ. - М.: Издательский отдел «Русская Редакция» ТОО «Channel Trading Ltd.». - 1997.
5. Олифер В.Г., Олпфер 11.А. Компьютерные сети. 3-е издание. М.- Л.: Питер, 2006.

ЛАБОРАТОРНАЯ РАБОТА №4

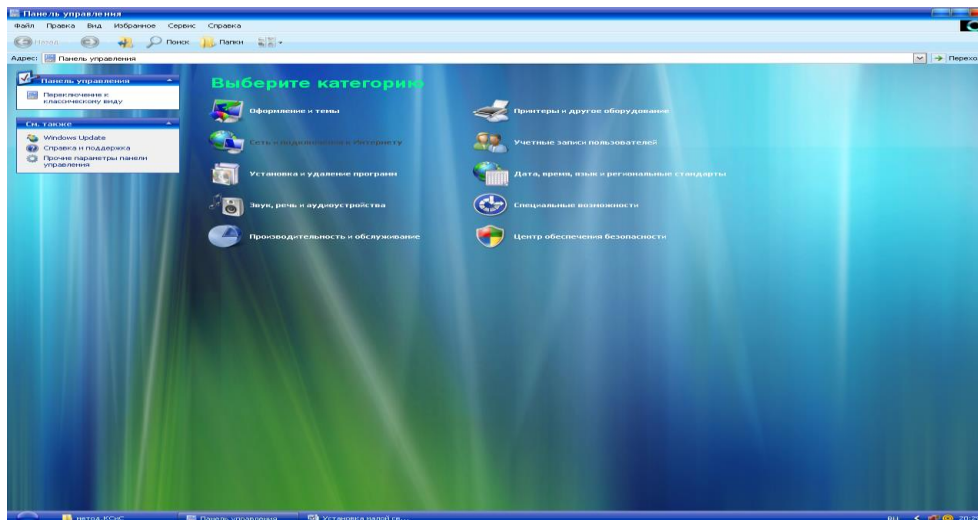
«Аппаратные средства и сетевая конфигурация малых КС»

Цель работы:

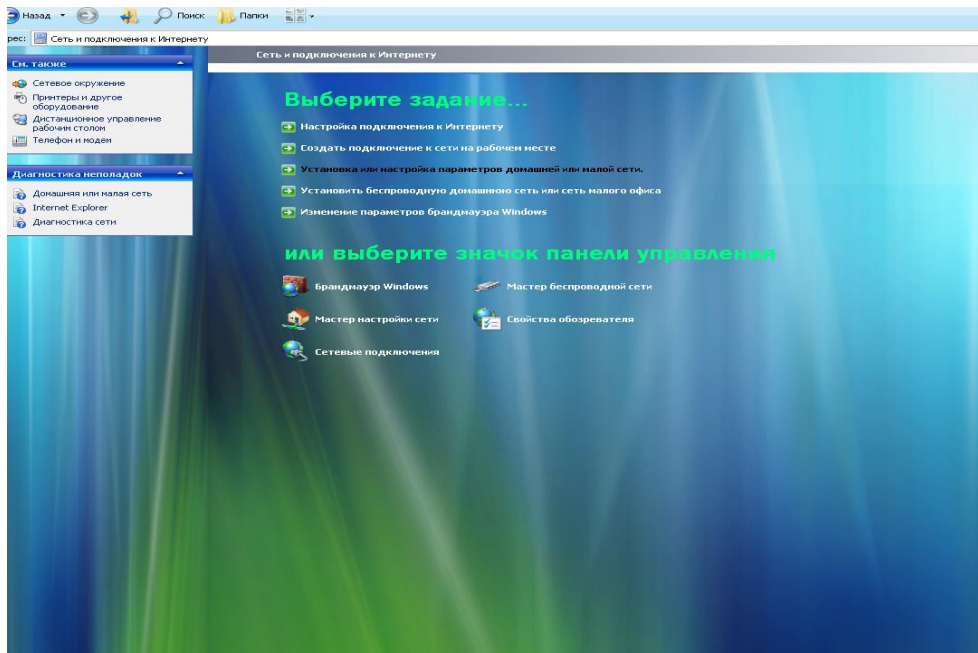
1. Краткие теоретические сведения о домашних или малых сетях.
2. Изучить основные требования к аппаратному обеспечению
3. Изучить установку параметров домашней сети
4. Реализация прямого соединения компьютеров

Порядок выполнения работы:

1. Зайти в «Пуск – настройка – панель управления»

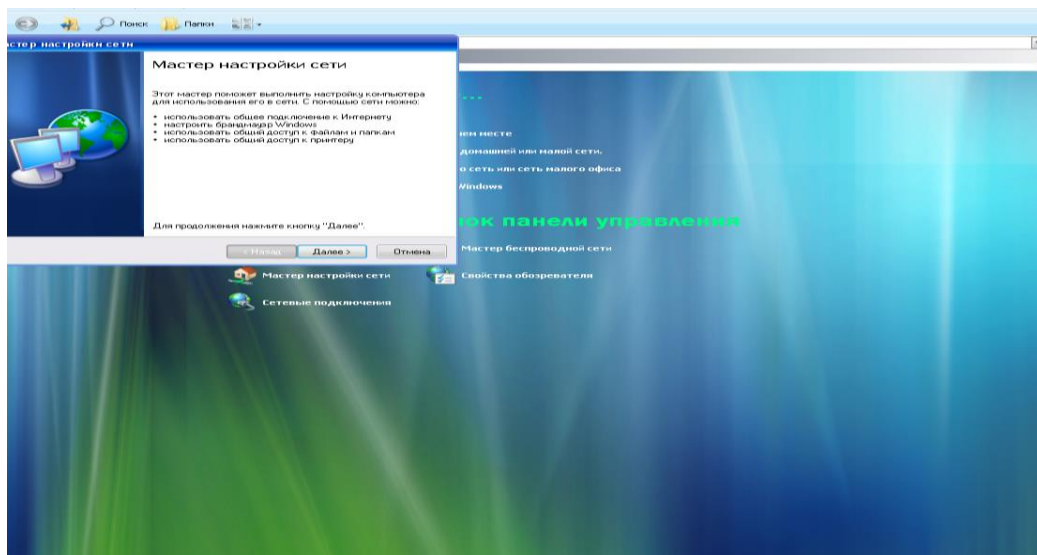


2. Выбрать «Сеть и подключение к Интернету»

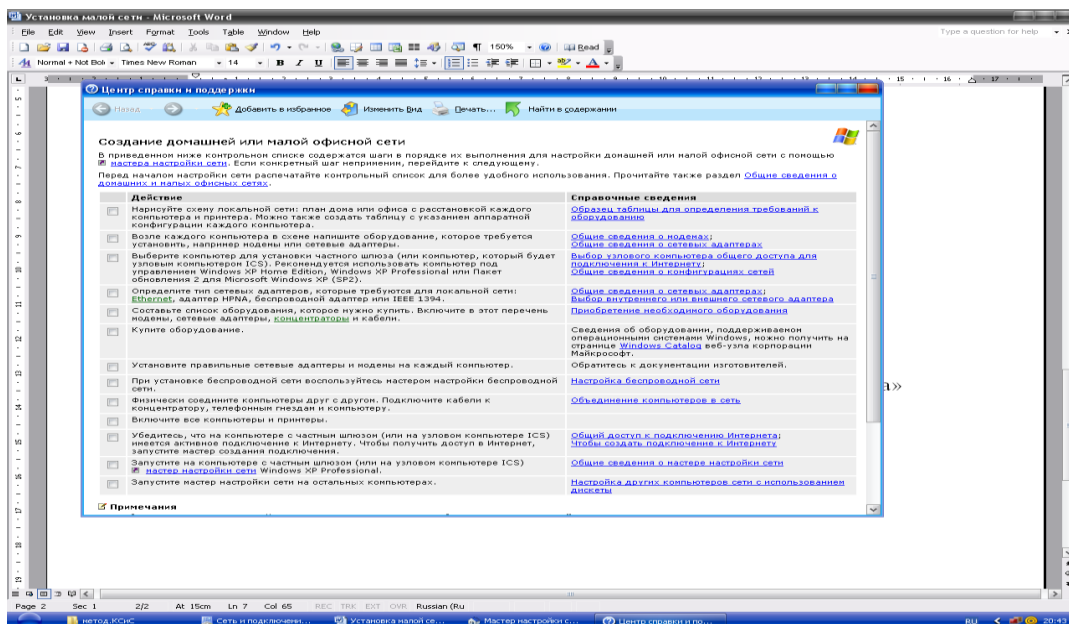


3. Выбрать «Установка или настройка параметров домашней или малой сети».

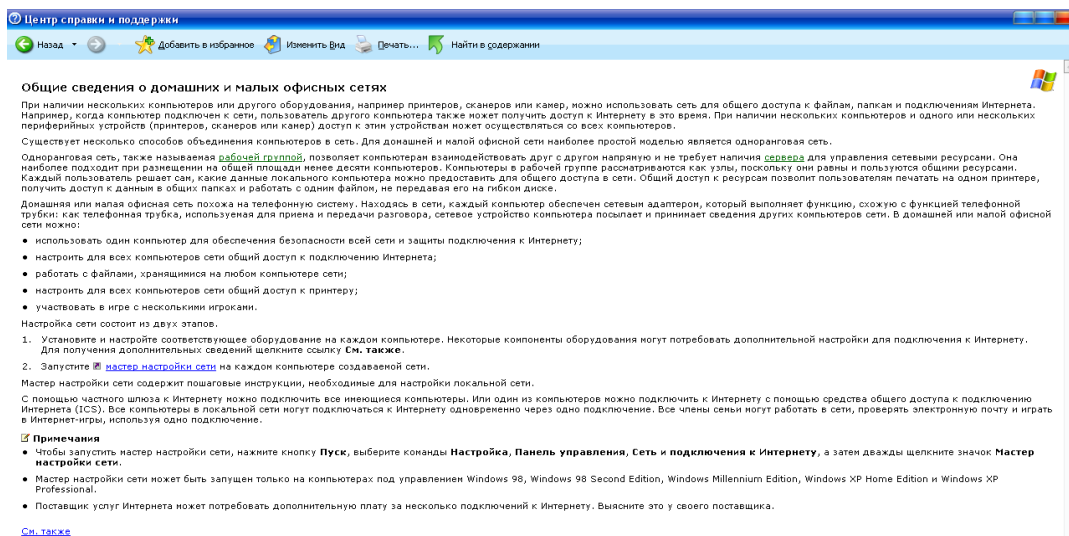
Появится окно «Мастер настройки сети»



4. Ознакомьтесь с разделом справки «контрольный список, установка»



5. Ознакомьтесь с разделом Общие сведения о домашней и малой сети



6. Ознакомьтесь со справочными сведениями (модемы, сетевые адаптеры, общие сведения о конфигурациях сетей, узловой компьютер, шлюзы).

Порядок выполнения работы

Выполните первые пять действий

1. Нарисуйте схему локальной сети: план дома или офиса с расстановкой каждого компьютера и принтера. Можно также создать таблицу с указанием аппаратной конфигурации каждого компьютера.
2. Возле каждого компьютера в схеме напишите оборудование, которое требуется установить, например модемы или сетевые адаптеры.
3. Выберите компьютер для установки частного шлюза (или компьютер, который будет узловым компьютером ICS). Рекомендуется использовать компьютер под управлением Windows XP Home Edition, Windows XP Professional или Пакет обновления 2 для Microsoft Windows XP (SP2).
3. Определите тип сетевых адаптеров, которые требуются для локальной сети: [Ethernet](#), адаптер HPNA, беспроводной адаптер или IEEE 1394.
4. Составьте список необходимого сетевого оборудования. Включите в этот перечень модемы, сетевые адаптеры, [концентраторы](#) и кабели.
5. Составьте отчет: цель работы, краткие теоретические сведения, задания пункта 7.

Самая простая с точки зрения технической реализации возможность установления связи между двумя компьютерами - прямое соединение (peer-to-peer) по последовательным или параллельным портам. Встроенная функция операционной системы windows (начиная с версии 95) позволяет легко устанавливать прямое соединение компьютеров.

Порядок выполнения прямого соединения:

1. Проверьте возможность доступа извне к ресурсам вашего компьютера. Для этого выполните следующие действия: откройте Панель управления -> Сеть -> Конфигурация -> Доступ к файлам и принтерам и активизируйте нужную опцию.
2. Определите доступ к каким файлам и папкам должен иметь подключенный компьютер.
3. Выключите оба компьютера и присоедините кабель к соответствующим портам на задней стенке системного блока, после этого включите компьютеры

4.Далее запустите программу прямого соединения нажав на кнопку Пуск и выбрав последовательно пункты меню: Программы -> Стандартные -> Прямое соединение.

Если пункт меню Прямое соединение отсутствует, то выполните его установку, с помощью Панели управления

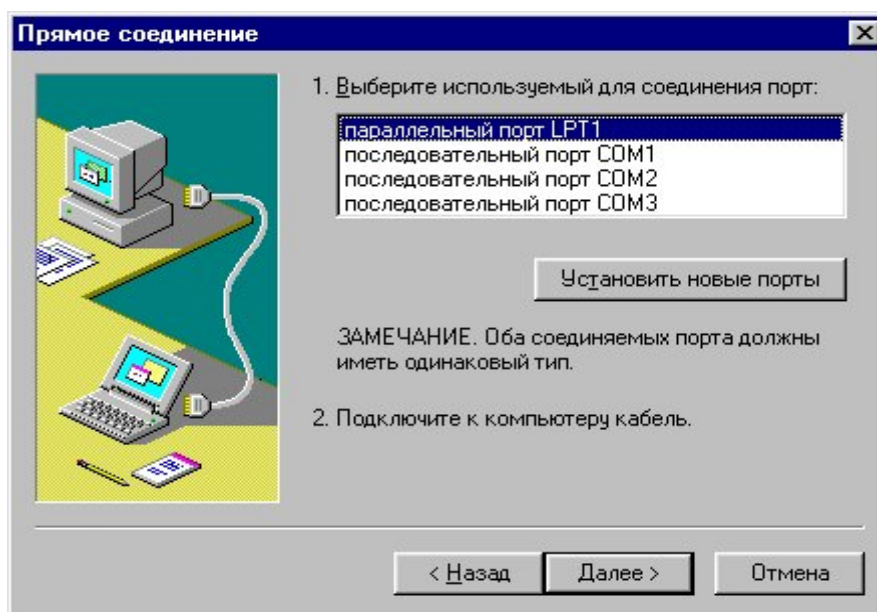
5.В появившемся окне определите статус вашего компьютера:



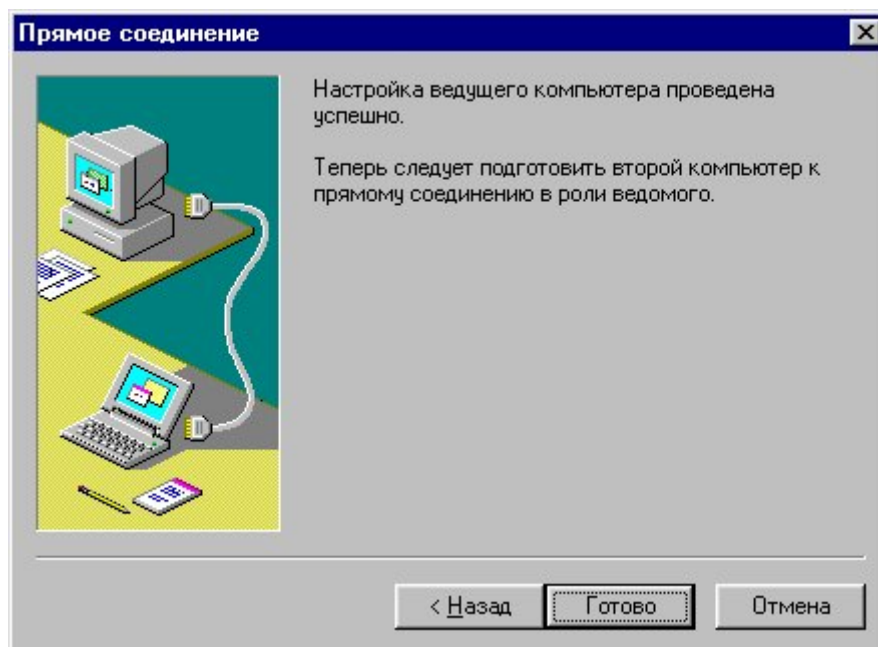
Ведомый. Компьютер, на котором находятся необходимые данные

Ведущий. Компьютер, осуществляющий доступ к ресурсам ведомого компьютера. Выберите Ведущий

6.Определив статус компьютера, нажмите кнопку Далее для открытия следующего диалогового окна - окна выбора порта соединения

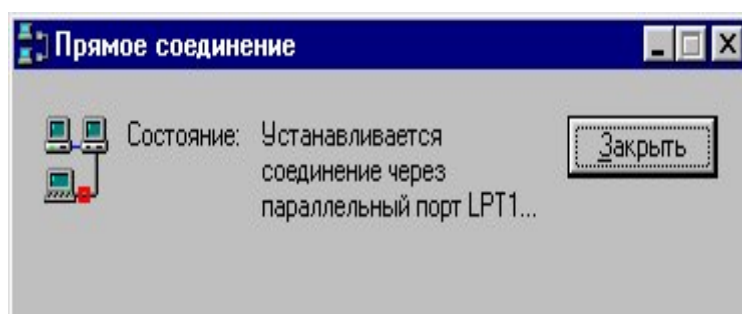


7. Выбрав нужный порт нажмите кнопку Далее, откроется следующее диалоговое окно



Прежде чем нажать кнопку Готово, проведите установку прямого соединения на ведомом компьютере. Она происходит аналогично вышеописанной процедуре, только теперь в первом диалоговом окне выберите пункт Ведомый

8. Запустите прямое соединение, нажав кнопку Готово на ведущем компьютере. Сначала на ведомом, а затем и на ведущем появится следующее окно:



9. При соединении со стороны ведущего компьютера укажите имя ведомого. Теперь вы получили доступ к ведомому компьютеру и хранящимся на нем данным.

Хотя прямое соединение компьютеров является самым быстрым и дешевым способом, оно имеет ряд недостатков.

Наиболее существенный из них состоит в том, что напрямую можно соединить только две машины. Следующим недостатком является низкая скорость передачи данных. При обмене небольшими файлами это не столь заметно, зато пересылка больших по объему данных существенно снижает общую производительность обеих компьютеров и увеличивает время, необходимое для выполнения этих операций.

Таблица, приведенная ниже показывает насколько различаются скорости передачи данных при прямом соединении и в 100-мегабитной сети.

Время, необходимое для передачи файла объемом 2 Мб при различных типах соединения		
Последовательный порт	Параллельный порт	100-мегабитная сеть
14 мин.	80 с.	0,5с.

Прямое соединение подходит в качестве решения только при передаче сравнительно небольших по объему файлов. Типичным случаем, когда прямая связь себя оправдывает, является файловый обмен между настольным компьютером и ноутбуком.

Контрольные вопросы

1. Какие основные аппаратные компоненты системного блока компьютера вы знаете?
2. Расскажите о принципе работы внутреннего интерфейса компьютера на примере программируемого параллельного адаптера?
3. Какие системные шины компьютера вы знаете, их организация?
4. Принципы считывания и записи в основную память компьютера?
5. Расскажите о принципе работы внешнего интерфейса компьютера?
6. Расскажите о драйверах компьютера различного назначения?
7. Клиент-серверная архитектура сети ?

8. Что такое среда передачи данных ?
9. Сеть Peer-TO-Peer и Client-Server ?

Литература

1. Фигурнов В.Э. IBM PC для пользователя. - М.: Финансы и статистика. 1997.
2. Мячев А.А., Степанов В.И. Справочник. Персональные ЭВМ и микро-ЭВМ. Основы организации. - М.: Радио и связь, 1991.
3. Windows XP Русская версия. Самоучитель по Windows XP. CD. 2005.
4. Microsoft Corporation. Компьютерные сети. Учебный курс / Пер. с англ. - М.: Издательский отдел «Русская Редакция» ТОО «Channel Trading Ltd.». - 1997.
5. Олифер В.Г., Олпфер 11.А. Компьютерные сети. 3-е издание. М.- Л.: Питер, 2006.

Лабораторная работа 5. Изучение построения IP адресации

Цель работы.

Изучение масок IP-адресации

Использование масок в IP-адресации

Традиционная схема деления IP-адреса на номер сети и номер узла основана на понятии класса, который определяется значениями нескольких первых бит адреса. Именно потому, что первый байт адреса 185.23.44.206 попадает в диапазон 128-191, мы можем сказать, что этот

адрес относится к классу В, а значит, номером сети являются первые два байта, дополненные двумя нулевыми байтами - 185.23.0.0, а номером узла - 0.0.44.206. А что если использовать какой-либо другой признак, с помощью которого можно было, бы более гибко устанавливать границу между номером сети и номером узла? В качестве такого признака сейчас получили широкое распространение маски. Маска - это число, которое используется в паре с IP-адресом; двоичная запись маски содержит единицы в тех разрядах, которые должны в IP-адресе интерпретироваться как номер сети. Поскольку номер сети является цельной частью адреса, единицы в маске также должны представлять непрерывную последовательность.

Для стандартных классов сетей маски имеют следующие значения:

класс А - 11111111.00000000.00000000.00000000 (255.0.0.0);

класс В - 11111111.11111111.00000000.00000000 (255.255.0.0);

класс С - 11111111.11111111.11111111.00000000 (255.255.255.0).

Снабжая каждый IP-адрес маской, можно отказаться от понятий классов

адресов и сделать более гибкой систему адресации. Например, если рассмотренный выше адрес 185.23.44.206 ассоциировать с маской 255.255.255.0, то номером сети будет 185.23.44.0, а не 185.23.0.0, как это

определено системой классов.

В масках количество единиц в последовательности, определяющей границу номера сети, не обязательно должно быть кратным 8, чтобы повторять деление адреса на байты. Пусть, например, для IP-адреса 129.64.134.5 указана маска 255.255.128.0, то есть в двоичном виде:

IP-адрес 129.64.134.5 - 10000001. 01000000.10000110. 00000101

Маска 255.255.128.0 - 11111111.11111111.10000000. 00000000

Если игнорировать маску, то в соответствии с системой классов адрес 129.64.134.5 относится к классу В, а значит, номером сети являются первые 2

байта - 129.64.0.0, а номером узла - 0.0.134.5. Если же использовать для определения границы номера сети маску, то 17 последовательных единиц в маске, «наложенные» на IP-адрес, определяют в качестве номера сети в двоичном выражении число:

10000001. 01000000. 10000000. 00000000 или в десятичной форме записи -

номер сети 129.64.128.0, а номер узла 0.0.6.5.

Механизм масок широко распространен в IP-маршрутизации, причем маски могут использоваться для самых разных целей. С их помощью

администратор может структурировать свою сеть, не требуя от поставщика

услуг дополнительных номеров сетей. На основе этого же механизма поставщики услуг могут объединять адресные пространства нескольких сетей путем введения так называемых «префиксов» с целью уменьшения

объема таблиц маршрутизации и повышения за счет этого производительности маршрутизаторов.

Компьютер в сети может иметь адреса трех уровней: физический (MAC-адрес), сетевой (IP-адрес) и доменный адрес (DNS-имя).

- Локальный адрес узла, определяемый технологией, с помощью которой построена отдельная сеть, в которую входит данный узел. Для узлов,

входящих в локальные сети - это MAC-адрес сетевого адаптера или порта маршрутизатора, например, 11-A0-17-3D-BC-01. Эти адреса

назначаются производителями оборудования и являются уникальными адресами, так как управляются централизованно. Для всех существующих технологий локальных сетей MAC-адрес имеет формат 6 байтов: старшие 3 байта - идентификатор фирмы производителя, а младшие 3 байта назначаются уникальным образом самим производителем. Для узлов, входящих в глобальные сети, такие как X.25 или frame relay, локальный адрес назначается администратором глобальной сети.

- IP-адрес, используется на сетевом уровне. Он назначается администратором во время конфигурирования компьютеров и маршрутизаторов. IP-адрес состоит из двух частей: номера сети и номера узла, например, 109.26.17.100.

Номер узла в протоколе IP назначается независимо от локального адреса узла. Деление IP-адреса на поле номера сети и номера узла - гибкое, и граница между этими полями может устанавливаться весьма произвольно. Узел может входить в несколько IP-сетей. В этом случае узел должен иметь несколько IP-адресов, по числу сетевых связей. Таким образом, IP-адрес характеризует не отдельный компьютер или маршрутизатор, а одно сетевое соединение.

– Символьный идентификатор-имя, например, SERV1.IBM.COM. Этот адрес назначается администратором и состоит из нескольких частей, например, имени машины, имени организации, имени домена. Такой адрес, называемый также DNS-именем, используется на прикладном уровне, например, в протоколах FTP или telnet.

Все компьютеры объединены в локальную сеть, и имеют локальную IP-адресацию. Пакеты с такой адресацией "путешествовать" в глобальной сети не смогут, т.к. маршрутизаторы их не пропустят.

Поэтому существует шлюз, который преобразовывает пакеты с локальными IP-адресами, давая им свой внешний адрес. И дальше ваши пакеты путешествуют с адресом шлюза (рис.1). Таким образом, очень

важно остановится на изучении IP-адреса. IP-адрес является уникальным 32-битным идентификатором IP-интерфейса в Интернет.

IP-адреса принято записывать разбивкой всего адреса по октетам, каждый октет записывается в виде десятичного числа, числа разделяются точками.

Например, адрес 101000000101000100000110000011 записывается как 10100000.01010001.00000101.10000011=160.81.5.131.

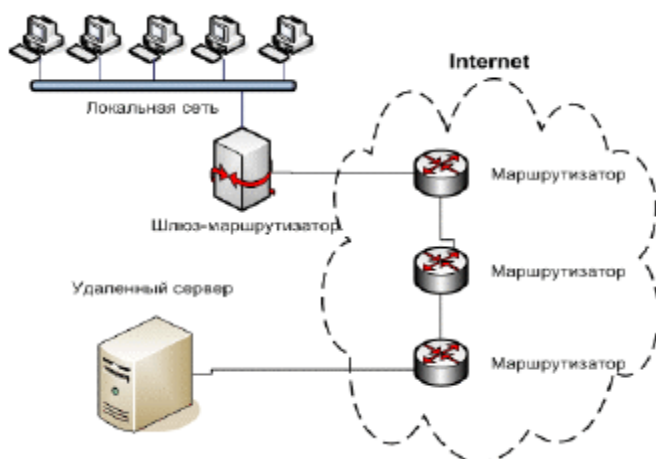


Рис.1. Схема прохождения пакетов из локальной сети к серверу

Положение границы сетевой и хостовой частей (обычно оно характеризуется количеством бит, отведенных на номер сети) может быть различным, определяя различные типы IP-адресов, которые рассматриваются ниже. Старшие биты 4-байтного IP-адреса определяют номер IP-сети, а оставшиеся биты - номер узла.

Классовая модель. IP-адреса разделяются на 5 классов, отличающихся

количеством бит в цифровом адресе сети и цифровом адресе узла (таблица 1).

Класс А	0	номер сети	номер узла
Класс В	10	номер сети	номер узла
Класс С	110	номер сети	номер узла
Класс D	1110	групповой адрес	
Класс E	11110	групповой адрес	

Адреса класса А предназначены для использования в больших сетях общего пользования. Адреса класса В предназначены для использования в сетях среднего размера (сети больших компаний, научно-исследовательских институтов, университетов). Адреса класса С предназначены для использования в сетях с небольшим числом компьютеров (сети небольших компаний и фирм).

Адреса класса D используют для обращения к группам компьютеров, а адреса класса E - зарезервированы.

- Если адрес начинается с 0, то сеть относят к классу А, и номер сети занимает один байт, остальные 3 байта интерпретируются как номер узла в сети. Сети класса А имеют номера в диапазоне от 1 до 126. (Номер 0 не используется, а номер 127 зарезервирован для специальных целей, о чем будет сказано ниже.) В сетях класса А количество узлов должно быть больше 216 , но не превышать 224.
- Если первые два бита адреса равны 10, то сеть относится к классу В и является сетью средних размеров с числом узлов 28 - 216. В сетях класса В под адрес сети и под адрес узла отводится по 16 битов, то есть по 2 байта.
- Если адрес начинается с последовательности 110, то это сеть класса С с числом узлов не больше 28. Под адрес сети отводится 24 бита, а под адрес узла - 8 битов.

– Если адрес начинается с последовательности 1110, то он является адресом класса D и обозначает особый, групповой адрес - multicast. Если в

пакете в качестве адреса назначения указан адрес класса D, то такой пакет должны получить все узлы, которым присвоен данный адрес.

– Если адрес начинается с последовательности 11110, то это адрес класса E.

В таблице приведены диапазоны номеров сетей, соответствующих каждому классу сетей.

Класс	Наименьший адрес	Наибольший адрес
A	0.1.0.0	126.0.0.0
B	128.0.0.0	191.255.0.0
C	192.0.1.0	223.255.255.0
D	224.0.0.0	239.255.255.255
E	240.0.0.0	247.255.255.255

Контрольные вопросы

- 1) Классификация протоколов.
- 2) Сетевые протоколы
- 3) Адресация в сети Internet. IP-адреса.
- 4) Транспортные протоколы.
- 5) Особенности TCP/IP.

Тесты

Из какого количества байт состоит IP-адрес?

- а) 1;
- б) 2;
- в) 3;
- г) 4.

4) MAC-адрес – это:

- а) адрес, назначаемый динамически при входе в сеть;
- б) адреса, назначаемые производителями оборудования и являющиеся уникальными;
- в) адрес, выбираемый пользователем при входе в сеть.

5) Старшие биты 4-байтного IP-адреса определяют:

- а) номер сети;
- б) номер подсети;
- в) номер хоста;
- г) MAC-адрес.

6) Сколько классов IP-адресов вы знаете?

- а) 2;
- б) 3;
- в) 5;
- г) 9.

7) Выберите IP-адрес, соответствующий интерфейсу обратной связи:

- а) 192.168.1.123;
- б) 127.0.0.1;
- в) 13.45.12.1;
- г) 1.1.1.1;

8) Выберите IP-адрес, соответствующий всей сети целиком:

- а) 192.168.1.1;
- б) 127.0.0.0;

в) 255.255.255.255;

г) 0.0.0.0.

9) Выберите IP-адрес, соответствующий широковещательной передаче:

а) 192.168.1.1;

б) 127.0.0.0;

в) 255.255.255.255;

г) 1.1.1.1.

Литература

1. Ларионов А.М., Майоров С.А., Новиков Г.И. Вычислительные комплексы, системы и сети. - Л.: Энергоатомиздат, 1987.
2. Максименков А.В., Селезнев М.Л. Основы проектирования информационно-вычислительных систем и сетей ЭВМ. - М.: Радио и связь, 1991.
3. Головкин Б.А. СуперЭВМ. Аппаратная и программная реализация. - М.: Наука, 1991.
4. Олифер В.Г., Олифер И.А. Компьютерные сети. 3 - е издание. М.- Л.: Питер, 2006.

Лабораторная работа № 6

Определение оптимальной топологии компьютерной сети

Цель работы - изучение и закрепление на практике методики определения топологии компьютерной сети, обеспечивающей минимальную суммарную длину линий связи между ее узлами.

Теоретическая часть

Основу проектирования компьютерных сетей составляют математические модели оптимизации структуры сетей. Компьютерная сеть может быть задана графом или численной моделью. Численные модели структур компьютерных сетей строятся на их конкретных конфигурациях. Рассмотрим частные случаи - описание структуры компьютерной сети.

Связностью сети называется минимальное число K независимых путей, имеющих между собой каждую пару узлов. Структура такой сети оптимизируется по критерию длины L . Считается заданным число конечных пунктов или число узлов коммутации и их местоположение.

Требуется соединить эти пункты или узлы таким образом, чтобы суммарная длина линий была минимальной.

Точное решение этой задачи производится по алгоритму Прима.

Алгоритм «Прима»

1. Среди множества заданных узлов сети отыскивается пара, расстояние между которой минимально, найденные узлы соединяются отрезком прямой и образуют фрагмент.
2. Среди оставшегося множества узлов ищется тот, который ближе других расположен к полученному фрагменту; если таких узлов несколько, то берется любой из них и подсоединяется к фрагменту, т.е. образуется новый фрагмент.
3. Процедура подсоединения узлов к фрагментам продолжается до тех пор, пока к сети не будет подключен последний узел.

На практике при программировании алгоритма Прима может быть использован подход, сущность которого видна из примера.

Пример. Пусть задано некоторое множество (h) конечных пунктов (рис. 6.1) или узлов коммутации и расстояния между ними, т.е. задана матрица (таблица 6.1).

Таблица 6.1

	1	2	3	4	5	6	7	8
1	0	8	6,1	3,4	8,9	2,1	3,7	7,4
2	8	0	4,6	9,9	10	6,9	4,4	6,9
3	6,1	4,6	0	5,9	5,9	4,9	6,1	11,9
4	3,4	9,9	5,9	0	6,4	3,9	6,4	6,1
5	8,9	10	5,9	6,4	0	5,8	11,9	12,9
6	2,1	6,9	4,9	3,9	5,8	0	7,1	8,9
7	3,7	4,4	6,1	6,4	11,9	7,1	0	6,1
8	7,4	6,9	11,9	6,1	12,9	8,9	6,1	0

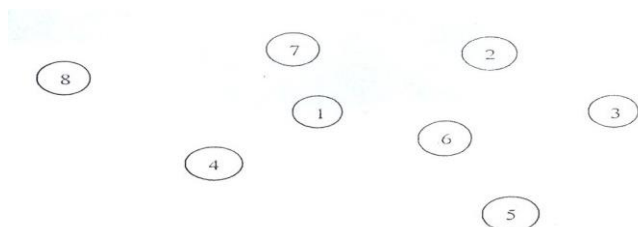


Рис. 6.1. Множество конечных пунктов

В соответствии с алгоритмом Прима выписываются первая строка матрицы без первого столбца (таблица 6.2), что соответствует организации связи от первой вершины к *i*-ой.

Таблица 6.2

2	3	4	5	6	7	8
8	6,1	3,4	8,9	2,1	3,7	7,4

$L_6, i = 2, 1$

Выбирается минимальный элемент строки и выписывается

минимальное расстояние L_j , равное минимальному элементу строки, где i - номер столбца, содержащего минимальный элемент, j - номер строки, который по умолчанию равен единице.

Формируется новая строка (таблица 6.3). При формировании следующей строки из предыдущей исключается i -й столбец с минимальным элементом. Данные новой строки сравниваются с данными i -ой строки исходной матрицы. В новую строку заносятся минимальные из этих значений с указанием номера строки. Алгоритм заканчивает работу, когда в строке останется один элемент.

Таблица 6.3

2	3	4	5	7	8
6,9	4,9	3,4	5,8	3,7	7,4
(6)	(6)		()		

$L_4, i = 3, 4$

Таблица 6.4

2	3	5	7	8
6,9	4,9	5,8		6,1
(6)	(6)	(6)		(4)

$L_{7.1} = 3, 7$

Таблица 6.5

2	3	5	8
4,4	4,9	5,8	6,1
(7)	(6)	(6)	(4)

(7)

Таблица 6.6

3	5	8
4,6	5,8	6,1

(2) (6) (4)

(7)

L з.2 = 4,6

Таблица 6.7

5	8
5,8	6,1

L 5.6= (6)

(4) (7)

5,8

Таблица 6.8

8
6,1

L 8.4= 6,1 L 8.7= 6,1

(4) (7)

Суммарная длина линии определяется как сумма всех полученных соединений L_{ij} . На рис. 6.2 приведена оптимальная топология сети, полученная в результате реализации алгоритма Прима.

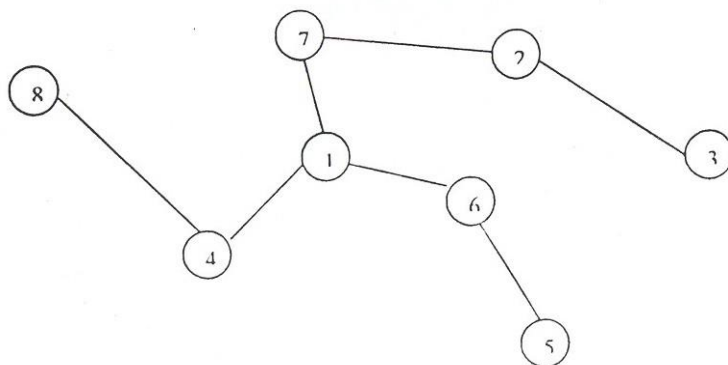


Рис. 6.2. Оптимальная топология сети
 $L = 2,1 + 3,4 + 3,7 + 4,4 + 4,6 + 5,8 + 6,1 = 30,1$

Задание для самоподготовки

1. Изучить алгоритм Прима на приведенном в описании лабораторной работы примере.
2. Произвести разработку блок-схемы алгоритма и программы синтеза компьютерной сети.
3. Задать координаты (x, y) $n = 7$ 4- 10 узлов компьютерной сети, принадлежащие множеству точек дискретного пространства: $X_j, y_i \in [200, 400]$; $\Delta x \gg \Delta y$; $i = 1, \dots, n$.
4. Подготовить текст программы реализации алгоритма Прима для составленной компьютерной сети.

Порядок выполнения работы

1. Включить компьютер и запустить операционную систему.
2. Загрузить транслятор Turbo Pascal и разработанную программу, при необходимости отредактировать ее.
3. Ввести исходные данные и запустить программу.
4. Проанализировать полученный результат работы программы, сравнить его с правильным решением задачи.
5. При необходимости скорректировать программу.
6. Правильный результат работы программы вывести на печать.

Контрольные вопросы

1. Что называется топологией сети?
2. Какие виды структур сетей Вы знаете?

3. Что такое связность сети?
4. Какие параметры функционирования сетей зависят от структуры сети?
5. В чем заключается сущность алгоритма Прима?

Литература

1. Дженнингс Ф. Практическая передача данных. Модели, сети и протоколы. М.: Мир. 1980.
2. Максименков А.В., Селезнев М. Основа проектирования информационно-вычислительных систем и сетей ЭВМ. - М.: Радио и связь, 1991.

Лабораторная работа № 7

Определение оптимального направления мультипрограммирования

Цель работы - изучение и закрепление на практике методики выбора оптимального направления мультипрограммирования для компьютерных систем.

Теоретическая часть

Основная задача мультипрограммирования заключается в том, чтобы на основе анализа типовых для данной компьютерной системы мультипрограммных ситуаций разработать и провести такие мероприятия, которые обеспечили бы наиболее эффективное функционирование компьютерной системы в данных условиях. И при этом под мультипрограммной ситуацией (МП — ситуацией) будем понимать определенную совокупность условий многопрограммной работы компьютерной системы, которая состоит из трех составляющих (элементов МП - ситуаций):

- 1) некоторого набора алгоритмов (в предельном случае одного алгоритма), который необходимо реализовать в компьютерной системе;
- 2) компьютерной системы;
- 3) задания (или требования) по реализации данного набора задач в компьютерной системе.

мероприятиям указанного типа следует отнести следующие:

1. Выявление и анализ типовых МП-ситуаций;
2. Преобразование исходных алгоритмов заданных наборов задач для обеспечения возможности реализации их в компьютерной системе;
3. Разработка оптимальных, в смысле выбранного действия, планов реализации заданных наборов задач в компьютерной системе; \у
4. Организация управления многопрограммной работой компьютерной системы в динамике, т.е. организация мультипрограммного диспетчирования (МП - диспетчирование);
5. Исследование эффективности компьютерной системы и мультипрограммирования.

В настоящей работе рассматриваются вопросы анализа МП - ситуаций. Основной целью 'такого анализа является определение оптимального, в смысле выбранного критерия, направления и уровня мультипрограммирования.

Сущность рассматриваемого метода анализа МП - ситуаций состоит в нахождении зависимости длины критического пути в графе (максимального или минимального, в смысле определения числовых характеристик) заданного набора задач от числа компьютеров в системе, а также в определении временных и вероятностных характеристик ситуаций, которые в совокупности позволяют судить об эффективности компьютерной системы в данной МП - ситуации.

Для определения оптимального направления мультипрограммирования необходимо построить граф заданного набора задач, полагая, что имеются такие характеристики алгоритмов этих задач, как

приоритеты задач, их функциональная связность и некоторые другие, например, среднее время t_0 их реализации на одном компьютере, время «ввода», «счета», «вывода» и т.д. Тогда, если множеству задач соотнести множество вершин, нагруженных числовыми значениями характеристик задач, а множеству функциональных связей и отношению порядка - множество дуг, соединяющих эти вершины, то получим нагруженный функциональный граф заданного набора задач.

Пример 1. Пусть имеет место таблица задач, которая указывает порядок их решения, функциональные связи и время t_0 реализации на единичном компьютере.

На рис. 4.1 изображен граф заданного набора задач. Однако более удобной для анализа формой представления заданного набора задач является ярусный граф.

Алгоритм получения ярусного графа следующий: все вершины исходного графа размещаются по множеству ярусов; к вершинам первого яруса относят те, у которых нет ни одной входной дуги; к вершинам второго яруса относят те вершины, для которых выходные дуги вершин первого яруса являются входными; к вершинам k -го яруса относят те, для которых входными дугами являются выходные дуги вершин ярусов с меньшими номерами и среди них выходная дуга по крайней мере одной вершины $(k-1)$ -го яруса. Для рассмотренного пр .-

мера приведенный граф преобразуется в ярусный граф, изображенный на рис. 4.2.

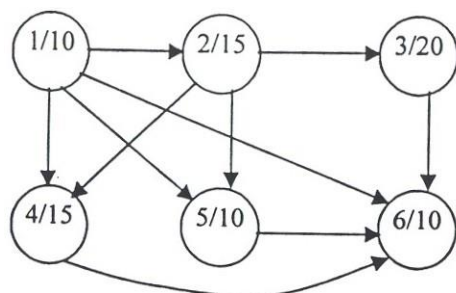


Рис. 4.1. Граф заданного набора задач

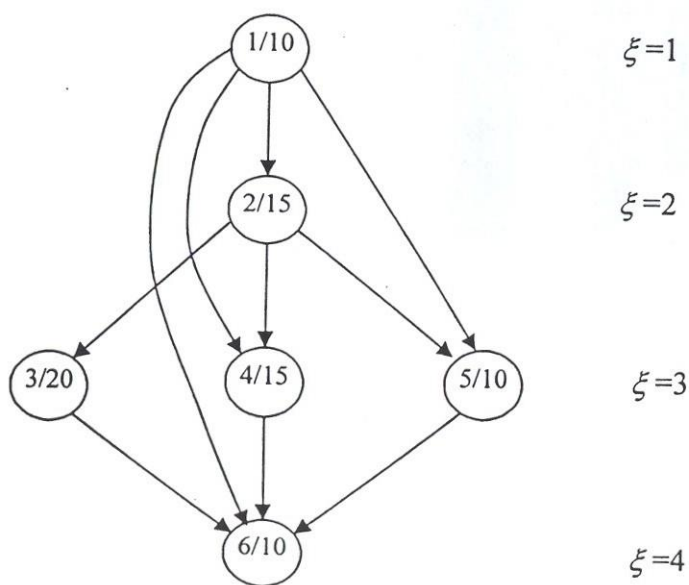


Рис. 4.2. Ярусный граф заданного набора задач

Имеют место в основном два направления мультипрограммного преобразования алгоритмов для реализации в компьютерной системе:

- а) распараллеливание каждого из алгоритмов заданного набора задач с последующей параллельной их реализацией в компьютерной системе;
- б) совмещение реализации алгоритмов заданного набора задач без распараллеливания.

Задача выбора оптимального направления состоит в преобразовании заданного набора задач.

Таблица 4.1

№ за- дачи	Идентифика- тор	Время (усл.ед.)	Данная задача	Зависи- мость
---------------	--------------------	--------------------	------------------	------------------

			решается после за- дачи №	данной задачи от задачи
1		10	-	-
2		15	1	1
3		20	2	
4		15	2	1
5		10	2	1
6		10 •	3, -1, 5	1

Обычно первое направление называют г - направлением, а второе направление называют S - направлением. Целесообразность применения того или иного направления при решении задач зависит от соответствующего подбора задач и характеристик системы.

Пример 2. Пусть имеется 4 задачи, каждая из которых состоит из трех частей, и два идентичных компьютера. Допустим также, что все задачи и их части идентичны с точки зрения времени их решения, а каждый компьютер допускает совмещение во времени выполнения двух частей различных задач с разными номерами, как, например, показано на рис. 4.3.

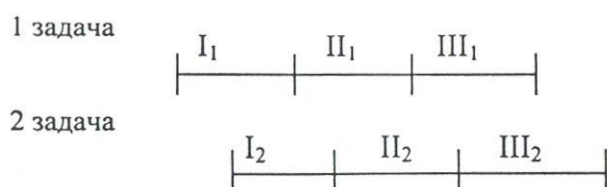


Рис. 4.3. Совмещение во времени решения двух частей различных задач на каждом компьютере

Тогда можно рассмотреть три случая:

1. Все задачи полностью распараллеливаются (т.е. I, II и III части каждой задачи можно решать параллельно), и все задачи не зависят одна от другой. При этом время решения 4-х задач при S - направлении преобразования алгоритмов будет равно $m_s = 4 \tau_0 / 2 = 2m_0$, что видно из рис. 4.4.

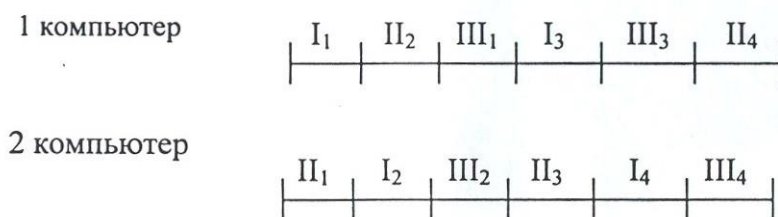


Рис. 4. 4. Решение 4-х задач при S - направлении

Время решения всех задач при г- направлении преобразования алгоритмов будет равно $\tau_g = \tau_0 + \tau_0 / 3 = 4\tau_0 / 3$, что видно из рис. 4.5.

2. Второй случай отличается от первого только тем, что рассматриваемые алгоритмы связаны между собой. При этом $m_s = 2m_0$, т.е. не изменяется. Действительно, на рис. 4.4 видно, что задачи решаются последовательно, т.е. указанное ограничение не изменяет общего времени решения задач.

Что же касается г - направления преобразования алгоритмов, то время увеличивается вдвое и равно $m_g = 4\tau_0$, что видно из рис. 4.6.

2. Алгоритмы решения задач связаны между собой и каждый из них не поддается распараллеливанию. При этом $\tau_g = \tau_s = 4\tau_0$. Таким образом, видно, что в тех или иных условиях желательно определить целесообразность того или иного преобразования алгоритмов.

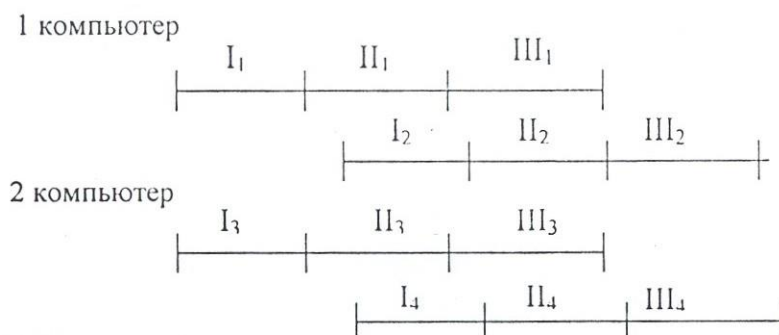


Рис. 4.5. Решение 4-х несвязанных между собой задач при g - направлении

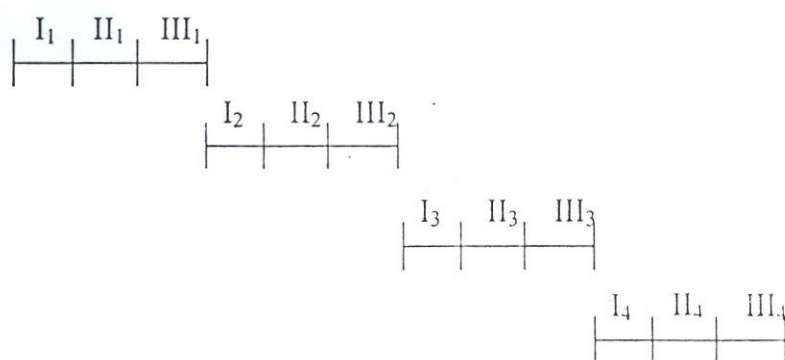


Рис. 4.6. Решение 4-х связанных между собой задач при g - направлении

Верхняя граница для обоих преобразований одинакова. Что же касается нижней границы, то она определяется при исследовании конкретного набора алгоритмов. Однако, для большинства задач инженерного и исследовательского характера граница определяется выражением (см. 1-ый случай) $M_{2,i} m$

где j - номер задачи ($j = 1, \dots, M$); M - число компьютеров в системе.

Если ступенчатую функцию $T_m = m \wedge M$, изображенную на рис. 4.7, аппроксимировать гиперболой (рис. 4.8), то можно видеть, что, начиная с некоторого (критического) числа компьютеров $M_{кр}$, дальнейшее уменьшение времени реализации заданного набора задач за счет увеличения числа компьютеров нецелесообразно.

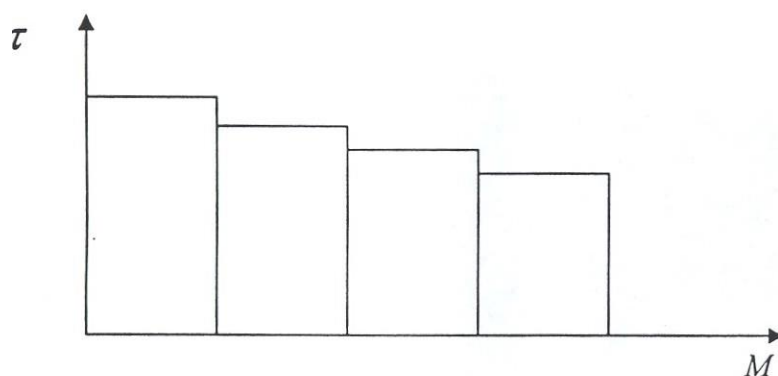


Рис. 4.7. График ступенчатой функции $\tau_r = \tau_r(M)$

Формулу (4.1) можно использовать для определения числа компьютеров в компьютерной системе, с помощью которых гарантируется возможность реализации заданного набора задач за заданное время T . Действительно для S-направления нижняя граница γ_s определяется связностью алгоритмов заданного набора задач и соотношением функциональных частей последовательных алгоритмов как внутри каждого алгоритма, так и между алгоритмами. Зависимости $\Gamma_s = \gamma_s(\sqrt{V})$ и величина M_{cr} могут быть определены на основе анализа графов заданного набора задач.

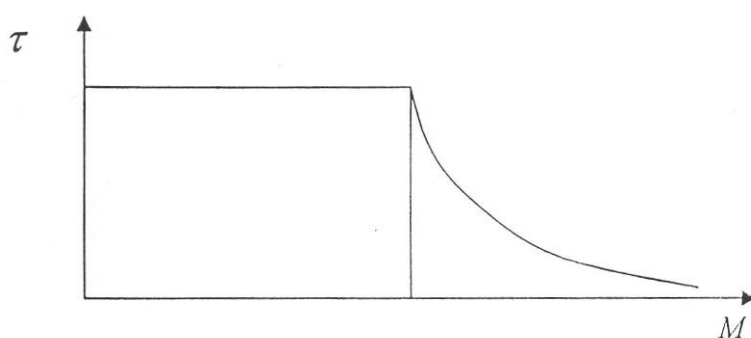


Рис. 4.8. Аппроксимация ступенчатой функции τ_r гиперболой

Пусть порядок реализации заданного набора задач в системе задан графом $G = G(X, V)$, в котором вершины X ($X = \{X_i, X_r, X_m\}$) соотнесены

алгоритмам заданного-набора задач, а дуги V ($V = \{V_1, V_2, \dots, V_N\}$) соотнесены связям отношений порядка выполнения этих алгоритмов (рис. 4.1).

Представим этот граф в ярусном виде (рис. 4.2). Максимальное (критическое) число загруженных вычислительных машин в системе при реализации заданного набора задач равно максимальной ширине графа G (в заданном примере $M_{кр} = 3$). Увеличение числа компьютеров в системе сверх $M_{кр}$ не уменьшит времени реализации заданного набора задач.

При $M = M_{кр}$ минимальное (критическое) время, за которое заданный набор задач может быть реализован в системе, определяется как длина (по времени) критического пути в графе G . Действительно, нагрузим вершины графа значениями времен X_j , $j = 1, M$, необходимых для реализации j -ой задачи на одном компьютере и будем искать длину критического пути (или просто критический путь) $m_{кр}$ в графе G как

$$m_{кр} = \max_{i \in V} \{t_i\}$$

$$t_i = \sum_{j \in P(i)} t_j + x_i$$

где j - номер вершины графа, $P(i)$ - матрица отношений вершин i -го и $(i-1)$ -го ярусов графа G . Матрица эквивалентна подграфу G_{i-1} - $(i-1)$ -го ярусов графа G . Например, для 3-го яруса графа G , (рис. 4.2) матрица отношений будет иметь вид

$$A_{32} =$$

$X_{кр} = 10 + 15 + 20 + 10 = 55$ условных единиц. Задача отыскания $m_{кр}$ в данном случае решается методом возвратного программирования. При этом рассмотрение графа G_i начинается с последнего N -го яруса, для которого можно записать

$$t_i = \max_{j \in P(i)} \{t_j\} + x_i \quad (4-4)$$

$$t_i = \max_{j \in P(i)} \{t_j\} + x_i$$

$$t_i = \max_{j \in P(i)} \{t_j\} + x_i$$

где j - номер вершины, лежащей на критическом пути в N -ом ярусе графа G . При решении выражения (4.4) будет найдена функция

$$j(N) = j(CN \wedge j(CN - I)) \quad (45)$$

Индуктивно для $(N-1)$ -го яруса можно записать

$$Ca^{''''} - \max N_{\theta} * \Gamma^+ \max x_j < > \}$$

или

$$z^{TM} \max b^* < \Gamma^+$$

$$j(\wedge - /) = j(iv. /)_{(j)(\wedge)} \quad (46)$$

Соответственно, для любого $(N - \varepsilon)$ -го яруса рекуррентно получим

$$.(N-4)$$

$$\psi'$$

$$(.1$$

$$(\mathcal{L}^r - c - 0j -$$

$$\max iV',$$

$$\bullet i \Gamma \quad (1.7)$$

$$4f$$

fdr.,	
1	
	—
ам	

Получив g_j, y , можно найти функцию $g''(A/)$. Для этого введем производную от g_ϕ по M как

О при $M > M_{кр}$ $XG^{\wedge}tyM$ при $M < M_{кр}$

где $\{\partial t / \partial M\}^i$ - частная производная от t^i по числу компьютеров в i -ом ярусе, $X(j^{(i)})$ - значение времени t^i -ой вершины, лежащей на критическом пути в i -ом ярусе, $M_{кр}$ - критическое число компьютеров i -го яруса.

Среднюю частную производную от $t_{кр}$ по M получим в виде $\sum_{i=1}^L \frac{\partial t^i}{\partial M}$.

$$1. \frac{\partial t_{кр}}{\partial M} = \sum_{i=1}^L \frac{\partial t^i}{\partial M} \quad (4.9)$$

СР = Л: 5. Тогда время реализации заданного набора задач (с учетом заголовка и мостов от числа компьютеров в системе) будет равно

$$T_{ср} = t_{кр} + \frac{\partial t_{кр}}{\partial M} (M - M_{кр}), \quad M = 2, L/ \quad (4.10)$$

Это и есть формула для функции $r_s = r_s(A)$. Далее, определив $r_r(M)$ и $r_s(M)$ и сравнив их значения при $M \sim M_{кр}$, можно сделать выбор между r_r и r_s направлениями и выбрать оптимальное направление мультипрограммирования в данной ситуации.

Пример 3. Пусть имеется следующая мультипрограммная ситуация. Заданный набор задач (16) представлен нагруженным (временем выполнения) графом равноприоритетных задач (рис. 4.9).

Нагрузка вершин графа приводится в минутах. Исходная система имеет следующие характеристики: $M = 3$, память общая, система прерывания аппаратная, внешние устройства неавтономны.

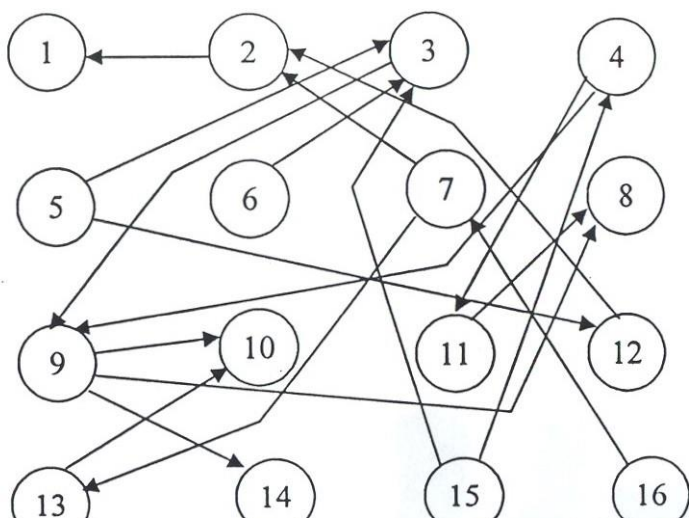


Рис. 4.9. Граф заданного набора задач

Задание на реализацию следующее: $t_3 < 2$ ч. $P_3 > 0,98$.

^Необходимо определить оптимальное направление М -преобразований алгоритмов заданного набора задач.

Определим возможность реализации заданного набора задач на компьютерной системы время t_3 по формуле (4.2). В данном примере это задание выполнено. VI. Построим ярусный граф (рис. 4.10).

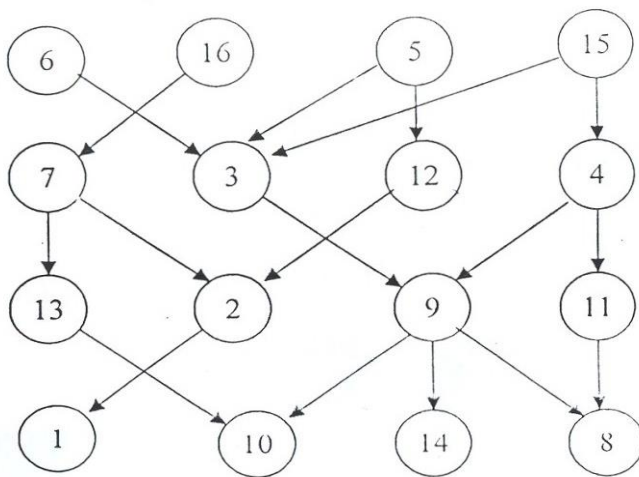


Рис. 4.10. Ярусный граф заданного набора задач

3. Определим критическое число компьютеров. Так как максимальная ширина графа H во всех ярусах одинакова и равна 4, то $M^{\wedge} = H = 4$ для $\varepsilon = 1, 2, 3, 4$.

4. Определим критический путь и критическое время, необходимое для реализации заданного набора задач/Для простоты записи вводится следующее обозначение: $X^{(w) \wedge} = X_j$

ay^{***} , т.е. и разрешенная матрицей CP^1 вершина для включения в критический путь, и соответствующее ей время обозначаются

Тогда уравнение (4.7) нахождения критического пути будет иметь вид

$$C \langle J^{< \wedge - \rangle} = \gg \{ X \Gamma^* \cdot \varepsilon - \bullet \bullet / \langle * - \rangle \rangle$$

Поярусное решение начинаем с $N-(5=4$.

Для $N-f=4$ имеем $\Gamma^{(j^{(3)})}$ - шах $X^{(f)}$

Получаем таблицу результатов (таблица 4.2).

Таблица 4.2

	13	2	9	1 1
	10	1	14	8
Γ	10	1	14	8

При составлении 'таблицы каждой вершине 3-го яруса в соответствии с матрицей

α или графом (рис. 4.10) находится одна вершина из 4-го яруса, обеспечивающая $\max X^{(4)}$. Для $N - f = 3$ имеем следующее уравнение для нахождения критического пути

$r \geq G^{(2)} - \max \{X^{(f)} + r_j\} (j^{(3)})$ и таблицу результатов (таблица 4.3)

Таблица 4.3

	7	3	12	4
$r(3)$	23	23	3	23
$j^{(j)}$	13	9	2	9

В данном случае каждой вершине 2-го яруса в соответствии с графом находится одна вершина из 3-го яруса,

обеспечивающая $\max \{X_y^{(3)} + r_p^* (j^{(3)})\}$ -

Для $N - f = 2$ имеем $\Gamma(j^{(1)}) = \max \{X^{(2)} + r(j^{(2)})\}$ и таблицу 4.4.

Таблица 4.4

	6	16	5	15
--	---	----	---	----

$r^{(2)}$	26	30	26	27
r	3	7	3	4

Для $N - \epsilon = 1$ имеем $r_{JJ} (j^{(0)}) = \max \{Xy\} + r^{TM}$ и таблицу 4.5.

Таблица 4.5

$j(0)$	0	0	0	0
	32	46	31	42
j_0	6	16	5	15

Таким образом, из последней таблицы видно, что $m_{кр} = 46$ минут. Это время получается на критическом пути 16, 7, 13, 10.

5. Выбор оптимального направления мультипрограммного преобразования алгоритмов.

Найдем значения производной от критического времени реализации заданного набора задач в системе по формуле (4.9). Рассчитанные значения приведены в таблице 4.6.

Таблица 4.6

Количество компьютеров					ΔM	ср
	к	)итический путь				
	16	7	13	10		
2	8	3,5	6,5	5	5,75	
3	5,33	2,33	4,33	3,33	3,83	
4	4	1,75	3,25	2,5	2,88	

Рассчитаем теперь значения функции $r_s = r_s(M)$ по формуле (4.10) для M ($M=2,3,4$). Результаты приведены в табл. 4.7. Для сравнения в этой же табл. приведены расчеты функции $r_r = r_r(M)$ по формуле (4.1).

Таблица 4.7

м	1	2	3	4
	136	68	45,33	34
T_s	136	57,5	49,83	46

.Из таблицы следует, что для двух компьютерного варианта компьютерной системы целесообразно выбрать S - направление, а для трех- и четырех компьютерного вариантов более эффективно г - направление. Гак как в условии задачи задай трех компьютерный вариант, то выбирается г - направление, т.е. необходимо каждый из алгоритмов заданного набора задач распараллелить и обеспечить последующую параллельную их реализацию на компьютерной системе.

Задание для самоподготовки

1. Ознакомиться с методикой построения ярусного графа, определения длины критического пути и выбора оптимального направления мультипрограммирования.
2. Составить текст программы решения задачи выбора оптимального направления мультипрограммирования на одном из алгоритмических языков высокого уровня.
3. Подготовить текстовый файл с таблицей смежности по заданному варианту.

Порядок выполнения работы

1. Включить компьютер и запустить операционную систему.
2. Загрузить транслятор Turbo Pascal и разработанную программу, при необходимости отредактировать ее.
3. Запустить отредактированную программу и вести в качестве исходных данных время выполнения задач.
4. Проанализировать полученные результаты работы программы.

5. При необходимости скорректировать программу.
6. Полученные правильные результаты работы программы вывести на печать.
7. Сделать выводы по полученным результатам.

Контрольные вопросы

1. Что понимается под мультипрограммированием?
2. В чем заключается основная задача мультипрограммирования?
3. На чем основана сущность метода анализа мультипрограммной ситуации?
4. Что необходимо осуществить для определения оптимального направления мультипрограммирования?
5. Как строится граф заданного набора задач?
6. Какие направления мультипрограммного преобразования заданного набора задач Вы знаете?
7. Как осуществляется поиск критического пути и критического времени, необходимого для реализации заданного набора задач?

Литература

1. Ларионов А.М., Майоров С.А., Новиков Г.И. Вычислительные комплексы, системы и сети. - Л.: Энергоатомиздат, 1987.
2. Максименков А.В., Селезнев М.Л. Основы проектирования информационно-вычислительных систем и сетей ЭВМ. - М.: Радио и связь, 1991.
3. Олифер В.Г., Олифер И.А. Компьютерные сети. 3 - е издание. М.- Л.: Питер, 2006.

ЗАКЛЮЧЕНИЕ

В методическом указании рассмотрены архитектура локальных сетей и среда передачи данных, методы передачи данных, протоколы модели OSI, протоколы межсетевого взаимодействия, сетевая операционная система. Это позволяет рассматривать взаимодействие локальных сетей как единую работающую коммуникационную систему.

Приводятся сегментация локальных сетей на основе сопряжения по скорости передачи данных сетевых адаптеров, концентраторов и коммутаторов, а также поддержки протоколов передачи данных аппаратными сетевыми средствами.

СОДЕРЖАНИЕ

Введение.....	3
Лабораторная работа №1. Изучение конфигурации ПК и его структуры...	3
Лабораторная работа №2 Установка сетевой ОС WINDOWS и его этапы, настройка приложений.....	7
Лабораторная работа №3. Организация локальной компьютерной сети...	15
Лабораторная работа №4. Аппаратные средства и сетевая конфигурация малых компьютерных сетей.....	21

Лабораторная работа №5. Изучение построения IP адресации.....	27
Лабораторная работа №6. Определение оптимальной топологии компьютерной сети	32
Лабораторная работа №7. Определение оптимального быстродействия процессора в компьютерной управляемой сети.....	36
Заключение.....	48

Методические указания к выполнению практических работ по дисциплине «Компьютерные системы и сети» для студентов по направлению бакалавриата высших учебных заведений:

5521900 - «Информатика и информационные технологии»,

5523500 – «Информационная безопасность»,

5523600 – «Электронная коммерция»,

5811100 – сервис предприятия «Электронная и компьютерная техника»,
5522200 – «Телекоммуникация».

Составители: Мирюсупов З.З., Назаров А.И.