

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ
Н.Г. ЧЕРНЫШЕВСКОГО»

Факультет компьютерных наук и информационных технологий

УТВЕРЖДАЮ

Декан факультета компьютерных наук
и информационных технологий

С.В. Миронов

"24" сентября 2021 г.

Рабочая программа дисциплины

КОМПЬЮТЕРНЫЕ СЕТИ И ИНТЕРНЕТ ТЕХНОЛОГИИ

Направление подготовки бакалавриата
44.03.01 – Педагогическое образование

Профиль подготовки бакалавриата
Информатика

Квалификация (степень) выпускника
Бакалавр

Форма обучения
Очная

Саратов,
2021

| Статус | ФИО | Подпись | Дата |
|--------------------------------|---------------------------------|---|----------|
| Преподаватель-разработчик | Векслер Виталий Абрамович |  | 24.09.21 |
| Председатель НМК | Кондратова Юлия Николаевна |  | 24.09.21 |
| Заведующий кафедрой | Александрова Наталья Алексеевна |  | 24.09.21 |
| Специалист Учебного управления | | | |

1. Цели освоения дисциплины

Целями освоения дисциплины «Компьютерные сети и интернет технологии» является формирование систематизированных знаний в области компьютерных сетей, изучение принципов организации компьютерных сетей, практическое освоение логики работы сетевых протоколов и системы адресации, принципов разработки и применения интернет-приложений для решения профессиональных задач.

2. Место дисциплины в структуре ООП

Данная учебная дисциплина относится к вариативной части Блока 1 «Дисциплины (Модули)» ООП (часть, формируемая участниками образовательных отношений), является дисциплиной по выбору и направлена на формирование у обучающихся профессионально-прикладных и специальных компетенций (Б1.В.ДВ.02.01).

Изучение данной дисциплины запланировано на втором году обучения.

Для изучения дисциплины необходимы компетенции, сформированные в результате изучения следующих дисциплин: «Информационные технологии в педагогическом образовании. Часть 1.», «Архитектура компьютера», «Компьютерная графика».

Компетенции, сформированные при изучении данной дисциплины, будут востребованы при изучении дисциплин «Компьютерное моделирование и пакеты прикладных программ», «Цифровая образовательная среда».

Курс «Компьютерные сети и интернет-технологии» содержательно и методологически взаимосвязан с курсом «Современные информационно-коммуникационные технологии в образовании».

3. Результаты обучения по дисциплине

| Код и наименование компетенции | Код и наименование индикатора (индикаторов) достижения компетенции | Результаты обучения |
|---|---|---|
| УК-4 Способен осуществлять деловую коммуникацию в устной и письменной формах на государственном языке Российской Федерации и иностранном (-ых) языке (ах) | 2.1_Б.УК-4. Использует информационно-коммуникационные технологии при поиске необходимой информации в процессе решения стандартных коммуникативных задач на государственном и иностранном (-ых) языках. 3.1_Б.УК-4. Ведет деловую переписку, учитывая особенности стилистики официальных и неофициальных писем, социокультурные различия в формате корреспонденции на государственном и иностранном (-ых) языках. | Знать: - продуктивный лексический запас на иностранном языке в рамках тематики курса; Уметь: - спрашивать и отвечать на вопросы и обмениваться идеями и информацией по знакомой тематике в рамках предсказуемых повседневных и деловых ситуаций; умеет делать короткие, заранее отрепетированные доклады, приводить краткие доводы и объяснения точек зрения в сфере профессиональной деятельности. Владеть: - стратегиями анализа и создания |

| | | |
|---|--|--|
| | <p>5.1_Б.УК-4. Демонстрирует умение выполнять перевод академических текстов с иностранного (-ых) языка (-ов) на государственный язык.</p> | <p>устных и письменных текстов, используя элементарные синтаксические структуры с заученными конструкциями, словосочетания и стандартные обороты для того, чтобы передать ограниченную информацию по темам курса.</p> |
| <p>ПК - 7. Способен использовать математический аппарат, методы программирования и современные информационно-коммуникационные технологии для решения практических задач получения, хранения, обработки и передачи информации</p> | <p>ПК - 7.1. Решает практические задачи получения, хранения, обработки и передачи информации. ПК - 7.2. Использует математический аппарат, методы программирования и современные информационно-коммуникационные технологии для решения учебных задач.</p> | <p>Знать:</p> <ul style="list-style-type: none"> - Интернет-сервисы для обмена информацией; - принципы построения компьютерных сетей; - протоколы и технологии передачи данных в сетях; - состав и принципы функционирования Интернет-технологий; <p>Уметь:</p> <ul style="list-style-type: none"> - взаимодействовать с обучающимися через системно-деятельностный, исследовательский подходы в образовании; - организовывать непосредственное общение с участниками образовательного процесса; - разрабатывать простейшие сетевые приложения, основанные на архитектуре клиент-сервер; <p>Владеть:</p> <ul style="list-style-type: none"> - навыками обмена информацией с использованием различных Интернет-сервисов; - навыками управления общением в ходе педагогического взаимодействия; - навыками совместного социально-педагогического проектирования. <p>способами создания информационных и интерактивных Интернет-ресурсов.</p> <ul style="list-style-type: none"> - - современной классификацией и методологией построения компьютерных сетей и интернет/Интернет-технологий. |

4. Структура и содержание дисциплины

Общая трудоемкость дисциплины составляет 4 зачетные единицы 144 часа.

| № п/п | Раздел дисциплины | Семестр | Неделя семестра | Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах) | | | | | СР | Формы текущего контроля успеваемости (по неделям семестра) Формы промежуточной аттестации (по семестрам) |
|-------|--|---------|-----------------|--|-----------|--------------------|----------------------------------|-----------|----------------------|---|
| | | | | Всего часов | Лекции | Практические | | | | |
| | | | | | | Общая трудоемкость | Из них – практическая подготовка | | | |
| 1 | Понятие о локальных и глобальных компьютерных сетях. | 3 | 1-3 | 16 | 6 | 4 | 1 | 6 | Реферат | |
| 2 | Стек протоколов TCP/IP | 3 | 4-6 | 24 | 6 | 8 | 2 | 10 | | |
| 3 | Глобальная сеть Internet. | 3 | 7 | 8 | 2 | 2 | 1 | 4 | Тест | |
| 4 | Электронная почта. | 3 | 8 | 8 | 2 | 2 | | 4 | Контрольная работа 1 | |
| 5 | World Wide Web, сервисы сети Интернет | 3 | 9 | 6 | 2 | 2 | 1 | 2 | | |
| 6 | Создание HTML-документов. | 3 | 10-12 | 20 | 6 | 6 | 2 | 8 | | |
| 7 | Язык JavaScript и создание интерактивных ресурсов. | 3 | 13-15 | 20 | 6 | 6 | 2 | 8 | Реферат | |
| 8 | Мультимедиа и Интернет. | 3 | 16 | 6 | 2 | 2 | 1 | 2 | Контрольная работа 2 | |
| | Промежуточная аттестация | | | | | | | | Экзамен 36 | |
| | ИТОГО | | | 144 | 32 | 32 | 10 | 44 | | |

4.1 Содержание дисциплины

Понятие о локальных и глобальных компьютерных сетях. Понятие о локальных и глобальных компьютерных сетях. Назначение и классификация компьютерных сетей (КС). Архитектура компьютерных сетей. Характеристика процесса передачи данных. Аппаратная реализация передачи данных. Протоколы КС. Беспроводные сети.

Стек протоколов TCP/IP. Преимущество сети со стеком протоколов TCP/IP. Соответствие уровней модели и протоколов. Адресация в IP-сетях. Типы адресов: физический (MAC-адрес), сетевой (IP-адрес), символьный (DNS). Проблемы адресации в IP-сетях. IPv4. IPv6. Особенности адресации IPv6. DHCP. DNS. Маршрутизация. Служба архивов FTP. Удаленный доступ к ресурсам Интернет (протокол Telnet).

Глобальная сеть Internet. История создания. Представление о структуре и системы адресации. Способы организации передачи информации. Поиск информации в Internet. Проблема поиска информации в Интернет. Работа с поисковыми системами. Основные принципы написания

запросов к поисковым машинам. Основные поисковые системы. Поиск в русском Интернет.

Электронная почта. Протоколы электронной почты. Принципы организации. Интерфейсные программы. Форматы почтовых сообщений. Системы почтовой рассылки.

World Wide Web, сервисы сети Интернет. Работа с World Wide Web (WWW). Понятие гипертекста. Основные компоненты технологии WWW. Архитектура построения системы. Протокол обмена гипертекстовой информации. Структура и характеристики программного обеспечения WWW.

Создание HTML-документов. Разработка собственных Internet-ресурсов. Язык гипертекстовой разметки HTML. Принципы построения и интерпретации HTML. Основные команды (теги) языка HTML. Гипертекстовые ссылки.

Язык JavaScript и создание интерактивных ресурсов. Основные синтаксические конструкции языка JavaScript. Использование JavaScript для создания Web-ресурсов.

Мультимедиа и Интернет. Понятие мультимедиа. Создание мультимедийных приложений. Мультимедиа как средство и технология.

План практических занятий

Наряду с прослушиванием лекций по курсу «Компьютерные сети и интернет технологии» важное место в учебном процессе занимают практические занятия, призванные закреплять полученные студентами теоретические знания.

Перед практическим занятием студенту необходимо восстановить в памяти теоретический материал по теме практического занятия. Для этого следует обратиться к соответствующим главам учебника, конспекту лекций. Каждое занятие начинается с повторения теоретического материала по соответствующей теме. Студенты должны чётко уметь ответить на вопросы, поставленные преподавателем. По характеру ответов преподаватель делает вывод о том, насколько тот или иной студент готов к выполнению упражнений. После такой проверки студентам предлагается выполнить соответствующие задания.

| № занятия | Тема | Задания для лабораторного практикума |
|------------------|--|---|
| 1 | 2 | 3 |
| 1 | Знакомство с cisco packet tracer | 1 |
| 2 | Знакомство со средой cisco packet tracer | 2 |
| 3-4 | Протоколы ARP и ICMP | 3 |
| 5 | DNS и Web сервер | 4 |
| 6-7 | Протоколы прикладного уровня | 5 |
| 8 | Создаем веб-браузер | 6 |
| 9 | Создание простейшего HTML-документа. | 7 |

| | | |
|----|---|----|
| | Форматирование шрифта и абзаца | |
| 10 | Вставка в HTML-документ рисунков. Создание закладок и гиперссылок | 8 |
| 11 | Создание и форматирование таблиц | 9 |
| 12 | Фреймы | 10 |
| 13 | Каскадные таблицы стилей | 11 |
| 14 | Введение В JavaScript | 12 |
| 15 | Вычисления в веб-документах | 13 |
| 16 | Создание сайтов при помощи конструкторов | 14 |

Практикум 1.

ЗНАКОМСТВО С CISCO PACKET TRACER

Цель работы: познакомиться с симулятором Cisco Packet Tracer Student, изучить элементы рабочей области и интерфейс симулятора.

Данный программный продукт разработан компанией Cisco и рекомендован использоваться при изучении телекоммуникационных сетей и сетевого оборудования. На основе программного продукта Packet Tracer есть возможность создавать сетевые топологии из широкого множества маршрутизаторов и коммутаторов компании Cisco, рабочих станций и сетевых соединений типа Ethernet, Serial, ISDN, Frame Relay. Функции симулятора могут быть пригодны как для обучения, так и для работы, настройки сети еще на этапе планирования.

Packet Tracer включает следующие особенности:

- Рабочее пространство для создания сети любого размера и сложности
- Моделирование в режиме реального времени
- Моделирование в режиме симуляции
- Графический интерфейс для взаимодействия с пользователем при настройке сетевых устройств
- Изображение сетевого оборудования с поддержкой добавления, удаления, перемещения различных компонентов

Данный симулятор позволяет студентам проектировать свои собственные сети, создавая и отправляя различные пакеты данных, сохранять и комментировать свою работу. Предоставляется возможность изучать и использовать такие сетевые устройства, как коммутаторы, маршрутизаторы, рабочие станции, определять типы связей между ними и соединять их.

Отличительной особенностью данного симулятора является наличие в нем режима симуляции (рис. 3.1). В данном режиме все пакеты, пересылаемые внутри сети, отображаются графически. Эта возможность позволяет студентам наглядно продемонстрировать, по какому интерфейсу в данный момент перемещается пакет, какой протокол используется и т. д. Работая в симуляторе в другом режиме, режиме реального времени, нельзя проследить за перемещением пакетов, сразу отображается конечный результат выполненных действий.

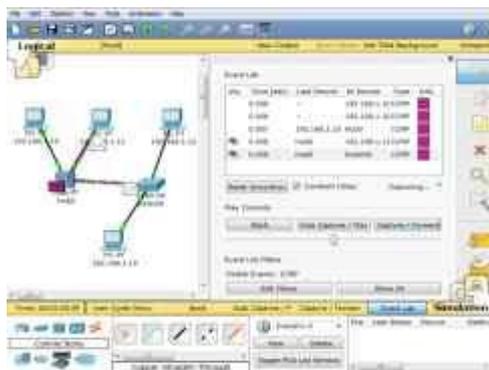


Рис. 3.1 Режим симуляции в Packet Tracer

Моделируемые устройства:

- Коммутаторы второго и третьего уровня
- Маршрутизаторы
- Сетевые концентраторы
- Конечные устройства (рабочие станции, ноутбуки, серверы, принтеры)
- Беспроводные устройства (точки доступа, беспроводные маршрутизаторы)
- Глобальная сеть WAN

Поддерживаемые типы связей между устройствами:

- Консоль
- Медный кабель с прямым подключением
- Медный кабель с перекрещиванием
- Волоконно-оптический кабель
- Телефонная линия
- Serial DCE/DTE

Каждое устройство в программном продукте Cisco Packet Tracer может быть сконфигурировано через окно свойств, которое вызывается по двойному клику на устройстве. Первая вкладка Physical отвечает за физические параметры устройства (рис.3.2). При настройке маршрутизаторов и коммутаторов в них можно добавлять новые модули, в рабочие станции и серверы — вставлять сетевые адаптеры.



Рис. 3.2 Физический вид устройства (маршрутизатора)

На вкладке Config можно задавать основные параметры сетевых интерфейсов (IP-адреса, маску подсети, параметры беспроводной сети и пр.) В сетевых устройствах также можно конфигурировать маршрутизацию – статическую или динамическую, у серверов — конфигурировать службы (рис. 3.3).



Рис. 3.3 Конфигурация сервера

Третья вкладка CLI сетевых устройств обеспечивает доступ к командной строке операционной системы IOS. Третья вкладка Desktop рабочих станций и серверов содержит интерфейсы доступа к различным сетевым параметрам, а также несколько клиентских приложений (рис. 3.4).

1. Menu Bar – предоставляет интерфейс управления для оконных приложений со стандартными разделами:

- File – управление файлом в программе;
- Edit – правка, позволяет выполнять с открытым документом различные операции;
- Options – опции программы;
- View – вид программы;



Рис. 3.4 Вкладка Desktop рабочей станции

При запуске программы открывается главное окно симулятора (рис. 3.5):

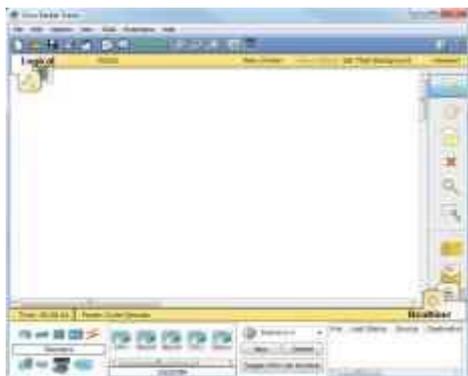


Рис. 3.5 Общий вид симулятора Packet Tracer

Опишем элементы рабочей области главного окна программы (рис. 3.6):

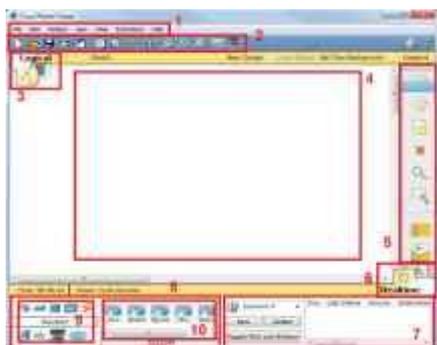


Рис. 3.6 Элементы рабочей области программы

- Tools – настройки;
 - Extensions – возможные расширения для программы;
 - Help – справка;
2. Main Tool Bar – панель управления содержит графические изображения ярлыков для доступа к командам главного меню File, Edit, View и Tools, а так же кнопку Network Information.
 3. Logical/Physical Workspace – вкладки переключения между логической рабочей областью и физической. Физическая топология подразумевает расположение устройств в городе, районе, офисе.
 4. Workspace – рабочая область программы, в которой происходит создание сети, проводятся наблюдения за симуляцией и просматривается другая информация о сети и статистика.
 5. Common Tools Bar – панель, которая обеспечивает доступ к часто используемым инструментам:
 - Select – выбрать элемент/отдельную область сети;
 - Move layout – перемещение по карте сети;
 - Place note – разместить комментарий на карте сети;
 - Delete – удалить элемент/отдельную область сети;
 - Inspect – просмотр подробной информации о выбранном устройстве;
 - Кнопки визуального моделирования потоков данных:
 - Add simple PDU – сформировать простой пакет ping-запроса между двумя узлами;
 - Add complex PDU – сформировать сложный пакет данных;
 6. Realtime/Simulation Bar – вкладки переключения между режимом realtime (реального времени) и режимом simulation (симуляции). Содержит кнопки Power cycle devices, Play control, Event list в режиме simulation.
 7. User Created Packet Window – окно для управления пакетами, которые были созданы в сети во время симуляции.
 8. Network Component Box – область, которая содержит все представленное оборудование, с помощью которого можно проектировать сеть.
 9. Device-Type Selection Box – содержит все доступные типы устройств и связей в симуляторе.
 10. Device-Specific Selection Box – содержит конкретные модели выбранного типа устройств и соединений.

Симулятор Packet Tracer поддерживает широкий диапазон сетевых соединений (таблица 3.1). Каждый тип кабеля может быть соединен лишь с определенным типом интерфейса.

Таблица 3.1

| Тип кабеля | Описание |
|------------|----------|
|------------|----------|

| | |
|--|---|
|  Console | <p>Консольное соединение может быть выполнено между ПК и маршрутизаторами или коммутаторами. Скорость соединения обеих сторон должна быть одинаковая, передаваться может любой поток данных.</p> |
|  Copper straight-through | <p>Этот тип кабеля является стандартной средой передачи Ethernet для соединения устройств, которые функционируют на разных уровнях OSI. Сигнал передается напрямую из одного конца в другой, а именно с 1-го контакта на 1-й, с 2-го на 2-й и т. д. Используется между ПК и хабом, ПК и DSL-модемом, хабом и коммутатором.</p> |
|  Copper cross-over | <p>Этот тип кабеля является средой передачи Ethernet для соединения устройств, которые функционируют на одинаковых уровнях OSI. Используется для соединения двух ПК напрямую, т. е. без хаба или коммутатора. Таким образом можно подключить только 2 компьютера одновременно.</p> |
|  Fiber | <p>Оптоволоконный кабель используется для соединения между оптическими портами.</p> |
|  Phone | <p>Соединение через телефонную линию может быть осуществлено только между устройствами, имеющими модемные порты.</p> |
|  Coaxial | <p>Коаксиальная среда используется для соединения между коаксиальными портами.</p> |
|  Serial Data Circuit Equipment/Data Terminal Equipment (DCE/DTE) | <p>Соединения через последовательные порты, часто используются для связей WAN. Для настройки таких соединений необходимо установить синхронизацию на стороне DCE-устройства. Синхронизация DTE выполняется по выбору. Сторону DCE можно определить по маленькой иконке “часов” рядом с портом. При выборе типа соединения Serial DCE, первое устройство, к которому применяется соединение, становится DCE-устройством, а второе - автоматически станет стороной DTE. Возможно и обратное расположение сторон, если выбран тип соединения Serial DTE.</p> |

Выводы _____

Практикум 2.

ЗНАКОМСТВО СО СРЕДОЙ CISCO PACKET TRACER

Цель работы: познакомиться с интерфейсом симулятора, изучить режим реального времени, основные операции с устройствами.

Программа работы:

1. Создание топологии сети;
2. Добавление конечных узлов;
3. Подключение к конечным узлам сетевых устройств;
4. Настройка IP-адресов и масок сети на узлах;
5. Проверка работы сети в режиме реального времени

Выполнение работы:

Запускаем среду Cisco Packet Tracer. При запуске программы открывается главное окно симулятора (см. рис. 3.5).

1. Построение топологии сети

Создаем новую топологию сети, выбираем необходимые устройства и соединения.

Топология сети может быть сконфигурирована из различных устройств и связей. В данной лабораторной работе мы используем простые сетевые устройства: концентратор, коммутатор, конечные устройства (компьютеры).

Network Component Box содержит все представленное оборудование, с помощью которого можно построить сеть (см. рис.3.6). С помощью одного клика по каждой группе устройств и соединений можно отобразить различные их варианты, отличающиеся между собой (рис. 4.1).





Рис. 4.1 Виды устройств и соединений

2. Построение топологии, добавление узлов
Один клик по конечным устройствам (рис. 4.2).



Рис. 4.2 Виды конечных устройств

Один клик по выбранному устройству, для нашей работы это PC (рис. 4.3).



Рис. 4.3 Выбор конечного устройства

Переместите курсор на рабочую область симулятора. Курсор должен превратиться в знак "+". Щелкните мышью в любом месте на области и выбранное вами устройство скопируется. Проведите эту процедуру еще три раза, на рабочей области у вас будет 4 PC (рис. 4.4).



Рис. 4.4. Вид рабочей области

3. Подключение к узлам концентратора и коммутатора

Выберите группу устройств концентраторы (Hubs), из этой группы выберите первую модель (Hub-PT). Разместите концентратор между PC0 и PC1 (рис. 4.5).

Задача концентратора довольно проста: он повторяет пакет, принятый на одном порту на всех остальных портах.



Рис. 4.5 Вид рабочей области

Подключим PC0 к Hub0, выбрав сначала тип подключения. Для этого случая подойдет медный кабель с прямым подключением (рис. 4.6).



Рис. 4.6 Выбор соединения с прямым подключением

Для подключения PC0 к Hub0 выполните следующие действия (рис. 4.7):

- 1) Один раз щелкните мышью на PC0
- 2) Выберите тип интерфейса FastEthernet
- 3) Переместите курсор на Hub0
- 4) Нажмите на Hub0 один раз и выберите порт 0
- 5) Обратите внимание на зеленые индикаторы двух устройств на соединении, что значит, оба устройства готовы к работе.

| | | | | |
|---|---|---|---|---|
| 1 | 2 | 3 | 4 | 5 |
|---|---|---|---|---|

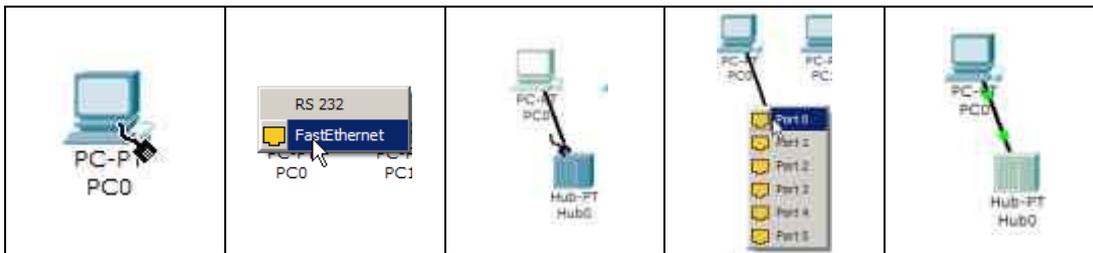


Рис. 4.7 Подключение PC0 к Hub0

Повторите описанные выше действия для подключения PC1 к Hub0, выбрав на концентраторе порт 1 (рис.4.8). Фактически номер порта значения не имеет, однако удобнее занимать порты последовательно.

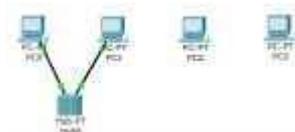


Рис. 4.8 Вид рабочей области

Далее размещаем на рабочей области симулятора коммутатор, например, модель 2950-24 (рис. 4.9). Описание семейства коммутаторов серии 2950 можно найти на сайте компании Cisco Systems. [Электронный ресурс]. URL:

<http://www.cisco.com/web/RU/products/hw/switches/ps628/ps627/index.html>.

Коммутаторы - это устройства, работающие на канальном уровне модели OSI и предназначенные для объединения нескольких узлов в пределах одного или нескольких сегментах сети. Коммутатор передает пакеты на основании внутренней таблицы - таблицы коммутации, следовательно, трафик идёт только на тот порт, которому он предназначен, а не повторяется на всех портах, в отличие от концентратора.



Рис. 4.9 Вид рабочей области

Подключим PC2 к Switch0, выбрав тип соединения медный кабель с прямым подключением.

Для подключения выполните следующие действия (рис. 4.10):

- 1) Щелкните мышью один раз на PC2
- 2) Выберите тип интерфейса FastEthernet
- 3) Переместите курсор на Switch0
- 4) Нажмите один раз на Switch0 и выберите FastEthernet0/1

5) Обратите внимание, что для правильной работы сети оба подключенных устройства должны быть готовы, о чем свидетельствуют зеленые индикаторы. В отличие от подключения к концентратору, это может занять некоторое время.



Рис. 4.10 Подключение PC2 к Switch0

Повторите описанные выше действия для подключения PC3 к Switch0, выбрав один из его интерфейсов FastEthernet0/2 (рис. 4.11).

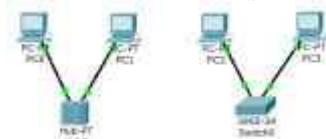


Рис. 4.11 Вид рабочей области

Если навести курсор на один из индикаторов, можно посмотреть, какой интерфейс задействован при данном подключении (рис. 4.12).

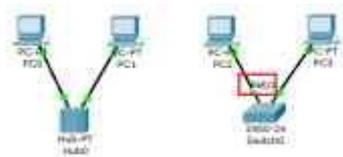


Рис. 4.12 Вид рабочей области

4. Настройка IP-адреса и маски подсети на хостах

Прежде чем мы сможем общаться между хостами по сети, нам нужно настроить IP-адреса и маски подсети на устройствах.

Щелкните мышью один раз на PC0. Откроется окно свойств конечного узла на вкладке Physical (рис. 4.13).



Рис. 4.13 Вкладка Physical конечного устройства (компьютера)

Физический вид устройства мы менять не будем, поэтому сразу переходим к настройке в вкладке Config (рис. 4.14).

Именно здесь вы можете изменить название PC0 (например, ввести IP-адрес этого компьютера, чтобы не подглядывать его каждый раз в настройках). Кроме того, здесь вы можете указать IP-адрес шлюза, также известный как шлюз по умолчанию, и IP-адрес DNS-сервера. Мы обсудим это позже, но это будет IP-адрес локального маршрутизатора. Если вы хотите, вы можете ввести IP-адрес шлюза 192.168.1.1 и IP-адрес DNS-сервера 192.168.1.100, хотя он не будет использоваться в этой лабораторной работе.



Рис. 4.14 Вкладка Config конечного устройства (компьютера)

Кликните мышью на интерфейсе FastEthernet (рис. 4.15). Укажите IP-адрес компьютера 192.168.1.10. Нажмите на поле для ввода маски подсети, она определится автоматически 255.255.255.0.



Рис. 4.15 Настройки интерфейса конечного устройства

Информация автоматически сохраняется после ввода.

Закройте окно настройки PC0 и повторите указанные выше действия для остальных узлов сети, используя информацию о IP-адресах и маски подсети, представленную в таблице 4.1

Таблица 4.1

| Хост | IP-адрес | Маска подсети |
|------|--------------|---------------|
| PC0 | 192.168.1.10 | 255.255.255.0 |
| PC1 | 192.168.1.11 | 255.255.255.0 |
| PC2 | 192.168.1.12 | 255.255.255.0 |
| PC3 | 192.168.1.13 | 255.255.255.0 |

После настройки узлов рабочая область симулятора будет выглядеть следующим образом (рис.

4.16):

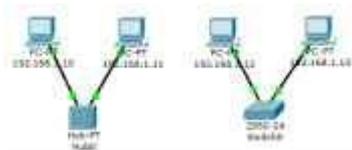


Рис. 4.16 Вид рабочей области

Можно проверить введенную вами информацию на узлах (рис. 4.17). Для этого наведите курсор на интересующее вас устройство.



Рис. 4.17 Проверка настроек конечного устройства (компьютера)

Если при построении сети какие-либо устройства или связи оказались лишними, их можно удалить при помощи инструмента Delete на боковой панели симулятора (Common Tools Bar). Для удаления нужно щелкнуть один раз на инструмент Delete, затем на элемент сети.

5. Соединение концентратора и коммутатора

Для подключения такого типа устройств, как коммутатора и концентратора, используется перекрестный кабель (рис. 4.18).



Рис. 4.18. Выбор соединения

Для подключения Hub0 к Switch0 выполните следующие действия:

- 1) Щелкните один раз на Hub0, выберите порт 2 (рис. 4.19).

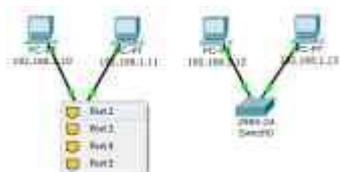


Рис. 4.19 Вид рабочей области

- 2) Переместите курсор на Switch0, щелкните на нем мышью и выберите интерфейс FastEthernet0/3 (рис. 4.20).

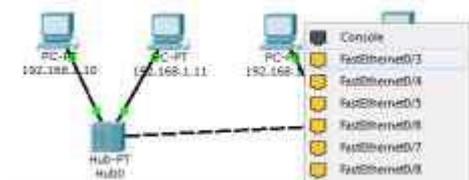


Рис. 4.20. Вид рабочей области

- 3) Когда оба устройства будут готовы к работе, индикаторы состояния станут зелеными (рис. 4.21).

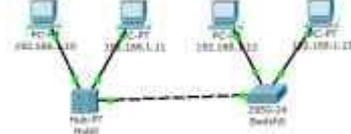


Рис. 4.21. Вид рабочей области

6. Выполним проверку в режиме реального времени
Убедитесь, что вы находитесь в режиме реального времени.



Сформируем простой пакет ping-запроса для проверки работы сети, воспользовавшись Add Simple PDU. Нажмите один раз на Add Simple PDU.



Теперь нужно выбрать два узла: источник и приемник ping-запроса. Наведите курсор на PC0 (192.168.1.10) и щелкните на нем мышью (источник ping-запроса), затем переместите курсор на PC3 (192.168.1.13) (приемник ping-запроса) и кликните на нем.

Так как все интерфейсы и связи сети настроены правильно (о чем говорят зеленые индикаторы состояния), то ping-запрос должен пройти успешно. В окне управления пакетами User Created Packet Window (см. рис. 3.6) появится соответствующая запись (рис. 4.22).



Рис. 4.22 Окно управления пакетами

Важно: измените IP-адрес 192.168.1.13 узла PC3 на IP-адрес 192.168.2.13, с той же маской подсети 255.255.255.0. Выполните ping-запрос от PC0 к PC3. Какой получился результат? Каковы причины?

Чтобы очистить список выполненных операций моделирования, необходимо удалить соответствующий сценарий симуляции.

Нажмите на кнопку Delete на панели User Created Packet Window (рис. 4.23).



Рис. 4.23 Окно управления пакетами

Все записи сценария удалятся.

7. Сохранение созданной топологии

Выберите в Menu Bar вкладку File, далее Save as. Выберите соответствующую директорию. Все файлы симулятора Cisco Packet Tracer имеют расширение .pkt.

8. Построение топологии сети, состоящей из двух подсетей

В результате первой работы мы изучили основные операции с устройствами. Для подготовки к выполнению следующей лабораторной работы у нас есть соответствующие знания и навыки для построения топологии сети следующего вида (рис. 4.24):

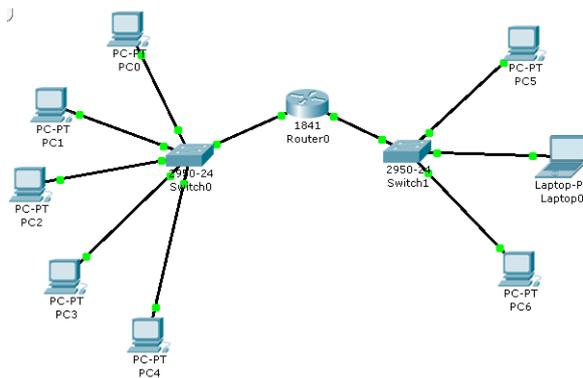


Рис. 4.24. Топология сети

Практикум 3. ПРОТОКОЛЫ ARP И ICMP

Цель работы: изучить режим симуляции Cisco Packet Tracer, протоколы ARP и ICMP на примере программ ping и tracer.

Программа работы:

1. Построение топологии сети, настройка конечных узлов;
2. Настройка маршрутизатора;
3. Проверка работы сети в режиме симуляции;
4. Посылка ping-запроса внутри сети;

5. Посылка ping-запроса во внешнюю сеть;
6. Посылка ping-запроса на несуществующий IP-адрес узла;
7. Выполнение индивидуального задания.

Выполнение работы:

1. Построение топологии сети

В конце вводной лабораторной работы мы создали следующую топологию сети, состоящую из конечных узлов (PC), коммутаторов и маршрутизатора (рис. 4.32):

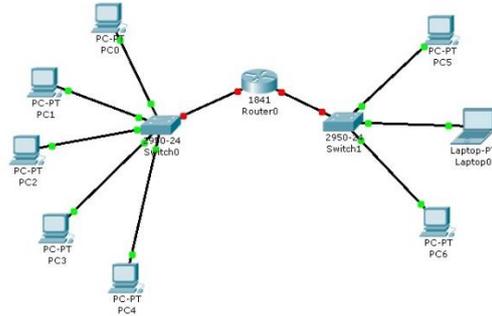


Рис. 4.32 Тестовая топология сети

Маршрутизатор Router0 имеет два интерфейса и соединяет две подсети. Произведем настройку конечных узлов.

2. Настройка конечных узлов

На устройствах PC0-PC4 установим заданные IP-адреса и маску подсети (таблица 4.2). IP-адрес шлюза для всех узлов – 192.168.3.1. IP-адрес DNS-сервера указывать необязательно, т.к. в данной работе он использоваться не будет.

Таблица 4.2

| Хост | IP-адрес | Маска подсети |
|------|-------------|---------------|
| PC0 | 192.168.3.3 | 255.255.255.0 |
| PC1 | 192.168.3.4 | 255.255.255.0 |
| PC2 | 192.168.3.5 | 255.255.255.0 |
| PC3 | 192.168.3.6 | 255.255.255.0 |
| PC4 | 192.168.3.7 | 255.255.255.0 |

На устройствах PC5, Laptop0, PC6 установим заданные IP-адреса и маску подсети (таблица 4.3). IP-адрес шлюза для всех узлов – 192.168.5.1. IP-адрес DNS-сервера указывать необязательно.

Таблица 4.3

| Хост | IP-адрес | Маска подсети |
|---------|-------------|---------------|
| PC5 | 192.168.5.3 | 255.255.255.0 |
| Laptop0 | 192.168.5.4 | 255.255.255.0 |
| PC6 | 192.168.5.5 | 255.255.255.0 |

Каждый узел переименуем его же IP-адресом, получится следующее (рис. 4.33):

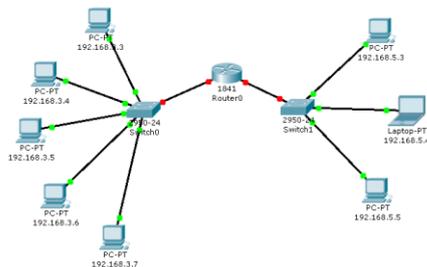


Рис. 4.33 Вид рабочей области

3. Настройка маршрутизатора

При настройке конечных узлов уже упоминалось о том, что маршрутизатор в данной топологии сети имеет два интерфейса. Произведем настройку интерфейса FastEthernet0/0:

- 1) Один клик по устройству (маршрутизатору);
- 2) Выбираем вкладку “Config”;
- 3) Находим интерфейс FastEthernet0/0, задаем нужный IP-адрес и маску подсети (рис. 4.34).

Важно: интерфейс маршрутизатора, по умолчанию, отключен; необходимо его включить, кликнув мышкой рядом с “On”.

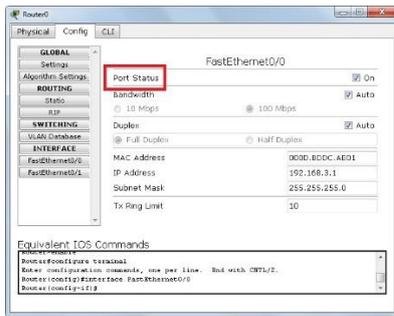


Рис. 4.34 Настройка интерфейса маршрутизатора

4) Закрываем окно, смотрим на всю топологию сети. Зеленые индикаторы состояния на линии связи между Router0 и Switch0 сигнализируют, что интерфейс подключен правильно (рис. 4.35).

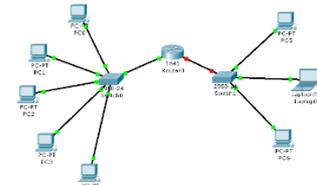


Рис. 4.35 Вид рабочей области

Аналогично производим настройку интерфейса FastEthernet0/1 (рис. 4.36).

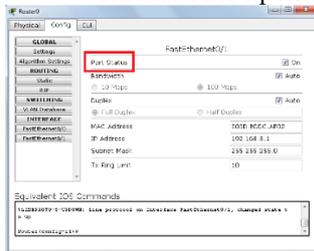


Рис. 4.36 Настройка интерфейса маршрутизатора

Сделать надписи к интерфейсам маршрутизатора, можно с помощью инструмента Place Note на панели Common Tools . Необходимо кликнуть на инструмент, затем сделать клик в нужном месте на рабочей области.

4. Режим симуляции Cisco Packet Tracer

Убедитесь, что вы находитесь в режиме симуляции. Для этого кликните на иконку симуляции в



правом нижнем углу рабочей области симулятора.

Откроется окно событий, в котором вы увидите список событий, управляющие кнопки, заданные фильтры (рис. 4.37). По умолчанию, фильтруются, т.е. будут отображаться, пакеты всех возможных протоколов, необходимо поправить и ограничить этот список до исследуемых протоколов.

Управляющие кнопки:

- Back – назад
- Auto Capture/Play – автоматический захват пакетов от источника до приемника и обратно
- Capture/Forward – захват пакетов только от одного устройства до другого



Рис. 4.37 Окно событий режима симуляции

В данной лабораторной работе нас интересуют пакеты двух типов ARP и ICMP.

Следовательно, нужно поставить фильтр только на сообщения заданного типа (рис. 4.38):

- 1) Нажимаем на кнопку “Edit Filters”
- 2) Снимаем метку с “Show All/None”
- 3) Выбираем ARP и ICMP

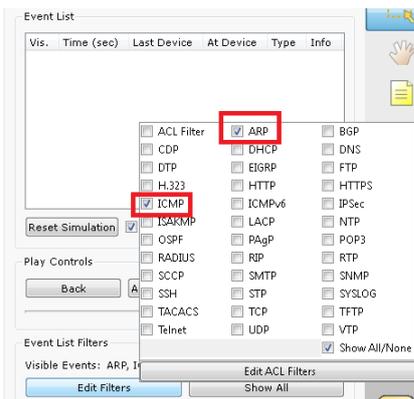


Рис. 4.38 Добавление фильтров на протоколы ARP и ICMP

- 4) Убедимся, что заданные протоколы для фильтрации назначены (рис. 4.39)



Рис. 4.39 Окно событий режима симуляции

5. Проверка работы сети в режиме симуляции

Отправим тестовый ping-запрос с конечного узла с IP-адресом 192.168.3.3 на хост с IP-адресом 192.168.3.5.

Важно: оба узла находятся в пределах одного сегмента сети

- 1) Один клик по выбранному устройству (рис. 4.40)

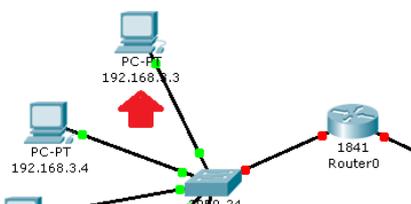


Рис. 4.40 Выбор узла 192.168.3.3

2) Выбираем вкладку Desktop, в которой содержатся симуляторы некоторых программ, доступных на компьютере (см. рис. 3.4).

- 3) Выбираем “Command Prompt”, программу, имитирующую командную строку компьютера.

- 4) С помощью утилиты ping отправляем ping-запрос (рис. 4.41). (Не забудьте нажать Enter).

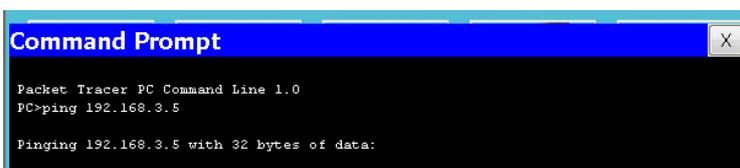


Рис. 4.41 Командная строка узла 192.168.3.3

На устройстве-источнике формируются два пакета протокола ARP и ICMP (рис. 4.42). ARP-запрос

возникает всегда, когда хост пытается связаться с другим хостом.

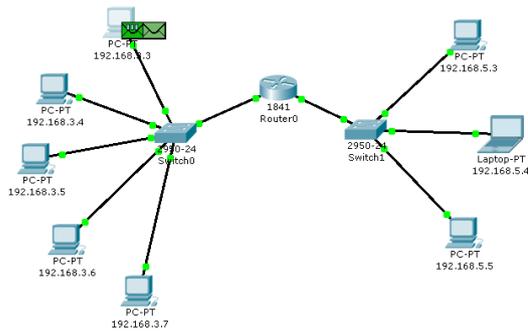


Рис. 4.42 Вид рабочей области

Нажимаем на кнопку “Auto Capture/play” или “Capture/Forward”, последняя позволит вам управлять движением пакетов от устройства к устройству самим. Видим, что первым отправляется пакет протокола ARP, так как ARP-таблица хоста 192.168.3.3 пуста, и он еще «не знает», кому отправлять ping-запрос. Сделайте один клик по самому пакету (конверту), ознакомьтесь, какие уровни модели OSI задействованы. Перейдите к вкладке “Inbound PDU Details”, которая содержит структуру пакета (рис. 4.43).

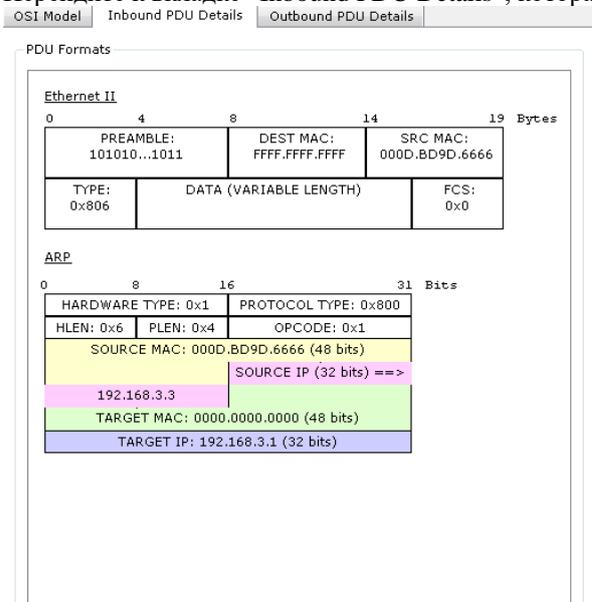


Рис. 4.43 Формат пакета ARP-запроса

Узел 192.168.3.3 построил запрос и посылает его широковещательным сообщением всем хостам подсети. Помимо IP-адреса назначения, запрос содержит IP-адрес и MAC-адрес отправителя, чтобы приемная сторона могла ответить.

При просмотре прохождения пакетов убедитесь, что на ARP-запрос ответит только хост 192.168.3.5. Каждый хост в подсети получает запрос и проверяет на соответствие свой IP-адрес. Если он не совпадает с указанным адресом в запросе, то запрос игнорируется (рис. 4.44).

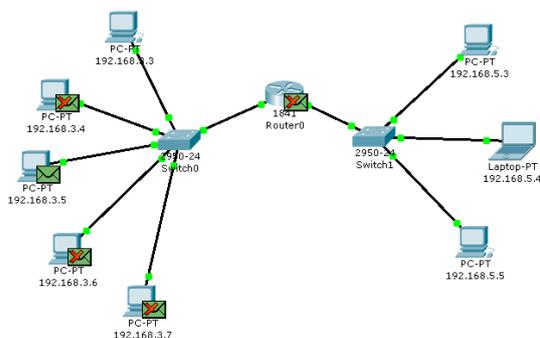


Рис. 4.44 Вид рабочей области

Посмотрите содержимое пакета ARP-ответа, пришедшего на хост 192.168.3.3 (рис. 4.45).

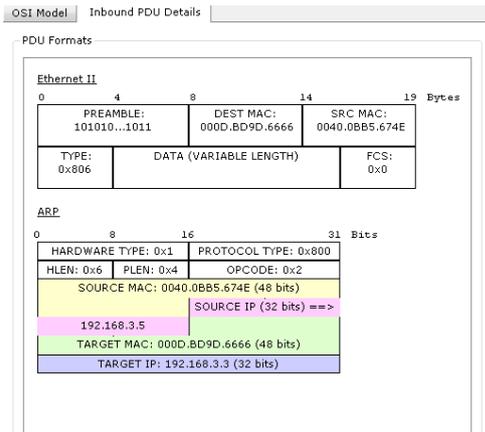


Рис. 4.45 Формат пакета ARP-ответа

Узел 192.168.3.5. послал ARP-ответ непосредственно отправителю, используя его MAC-адрес, с указанием собственного MAC-адреса в поле “Target MAC”.

Далее отправляется ICMP-сообщение ping-запроса. Посмотрите содержимое пакета, сделав клик по пакету (конверту) (рис. 4.46).

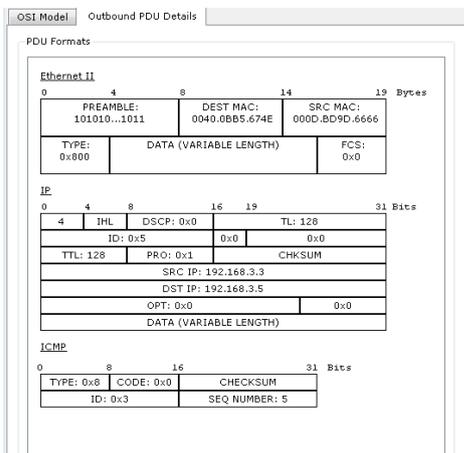


Рис. 4.46 Формат пакета ICMP-эхо-запроса

Физические адреса узлов известны. IP-адрес источника – 192.168.3.3. IP-адрес назначения – 192.168.3.5. Тип ICMP-сообщения – 8 (эхо-запрос).

Запрос производится на хост 192.168.3.5 через коммутатор (рис. 4.47).

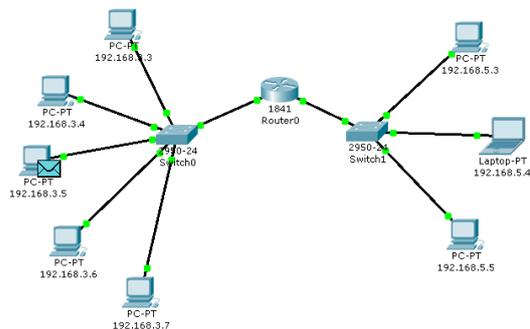


Рис. 4.47 Вид рабочей области

Посмотрите содержимое пакета ping-ответа, пришедшего на хост 192.168.3.3 (рис. 4.48).

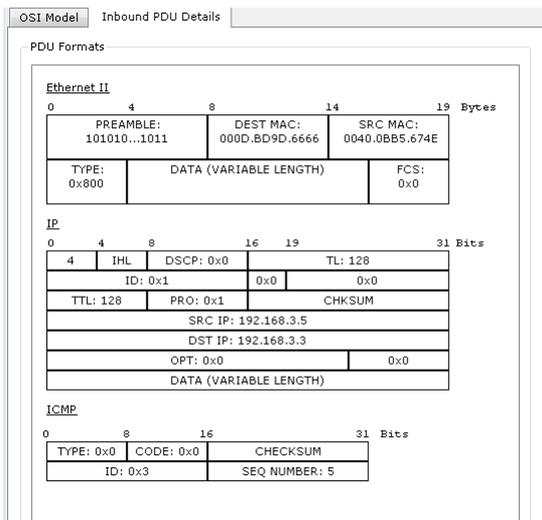


Рис. 4.48 Формат пакета ICMP-эхо-ответа

IP-адрес источника – 192.168.3.5. IP-адрес назначения – 192.168.3.3. Тип ICMP-сообщения – 0 (эхо-ответ).

Посмотрите ping-ответ в командной строке хоста 192.168.3.3 (рис. 4.49).

```

PC>ping 192.168.3.5

Pinging 192.168.3.5 with 32 bytes of data:

Reply from 192.168.3.5: bytes=32 time=8ms TTL=128
Reply from 192.168.3.5: bytes=32 time=4ms TTL=128
Reply from 192.168.3.5: bytes=32 time=4ms TTL=128
Reply from 192.168.3.5: bytes=32 time=4ms TTL=128

Ping statistics for 192.168.3.5:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 4ms, Maximum = 8ms, Average = 5ms
  
```

Рис. 4.49 Вывод программы ping

В окне событий так же указаны маршруты запроса ARP и ICMP: через какие устройства прошли пакеты (рис. 4.50).

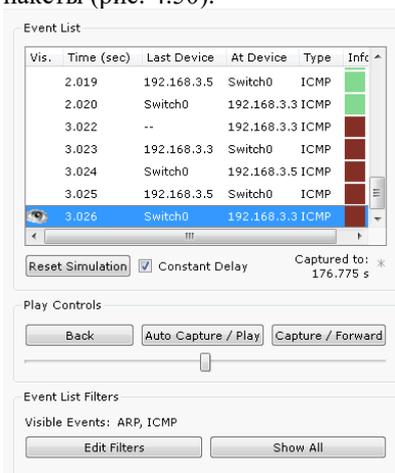


Рис. 4.50 Окно событий режима симуляции

Удалить сценарий симуляции можно с помощью кнопки “Reset Simulation” или воспользоваться кнопкой “Delete” в области User Created Packet Window.

Теперь ARP-таблицы хостов 192.168.3.3 и 192.168.3.5 не пусты, в них содержится одна запись. Чтобы просмотреть содержимое ARP-таблицы, нужно выполнить команду “arp -a” в командной строке.

Содержимое ARP-таблицы узла 192.168.3.3 (рис. 4.51):

```

PC>arp -a
Internet Address      Physical Address      Type
192.168.3.1          000d.bddc.ae01       dynamic
192.168.3.5          0040.0bb5.674e       dynamic

```

Рис. 4.51 ARP-таблица узла 192.168.3.3 в командной строке

Можно воспользоваться другим способом: нажать на кнопку «Inspect» , нажать на выбранное устройство, выбрать «ARP table» и просмотреть записи ARP-таблицы узла (рис. 4.52).

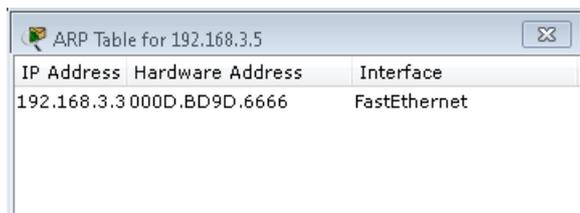


Рис. 4.52 ARP-таблица узла 192.168.3.5, показанная с помощью инструмента «Inspect»

Если снова задать ping-запрос на хост 192.168.3.5, то сразу будет сформирован только один пакет ICMP-сообщения, т.к. в ARP-таблице компьютера-источника уже хранится соответствующий локальный адрес.

Попробуйте отправить ping-запрос снова.

Чтобы удалить все записи ARP-таблицы, следует воспользоваться командой “arp -d”.

6. Псылка ping-запроса во внешнюю сеть

Отправим тестовый ping-запрос с конечного узла с IP-адресом 192.168.3.4 на хост с IP-адресом 192.168.5.5.

Важно: один узел пытается передать пакет другому узлу, находящемуся с ним в разных сетях.

В пункте 5 лабораторной работы был рассмотрен случай псылки ARP-запроса внутри локальной сети. Протокол ARP в этом случае определял непосредственно MAC-адрес узла-приемника запроса. Теперь рассмотрим ситуацию, когда узел-источник и узел-приемник находятся в разных сетях. Протокол ARP работает в пределах сегмента сети, поэтому в данном случае он будет использоваться для определения MAC-адреса маршрутизатора. Таким образом, пакет будет передан маршрутизатору для дальнейшей ретрансляции.

Открываем “Command Prompt”, имитирующую командную строку, на компьютере 192.168.3.4 и псылаем на хост 192.168.5.5. ping-запрос (рис. 4.53).

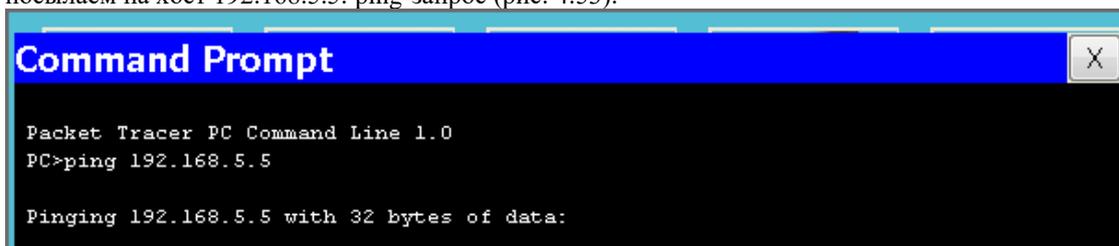


Рис. 4.53 Командная строка узла 192.168.3.4

В этом случае инициируется ARP-запрос маршрутизатору, который пересылает пакеты в сеть назначения. На узле-источнике формируются два пакета протокола ARP и ICMP (рис. 4.54).

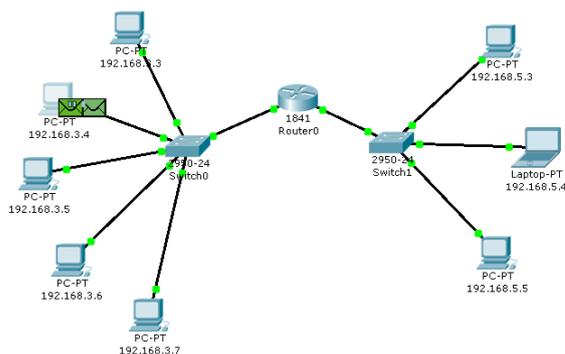


Рис. 4.54 Вид рабочей области

Формат пакета ARP-запроса содержит те же сведения, что и для разрешения локального адреса устройства, и рассылается широковещательно всем узлам подсети (рис. 4.55).

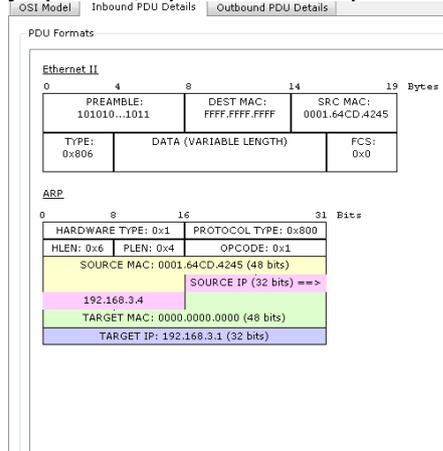


Рис. 4.55 Формат пакета ARP-запроса

Все узлы игнорируют пакет, кроме маршрутизатора, которому этот пакет предназначался (рис. 4.56).

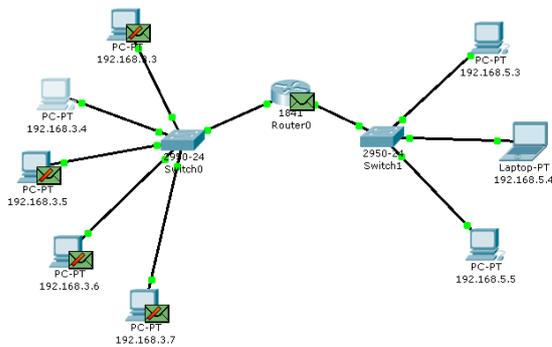


Рис. 4.56 Вид рабочей области

Маршрутизатор формирует ARP-ответ, указывая свой физический адрес, и отправляет его узлу 192.168.3.4 (рис. 4.57).

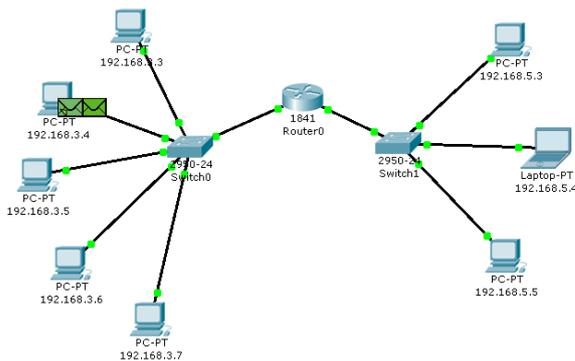


Рис. 4.57 Вид рабочей области

После получения ARP-ответа хост 192.168.3.4 посылает ICMP-сообщение ping-запроса через маршрутизатор в сеть назначения.

Посмотрите содержимое пакета, сделав клик по пакету (конверту) (рис. 4.58).

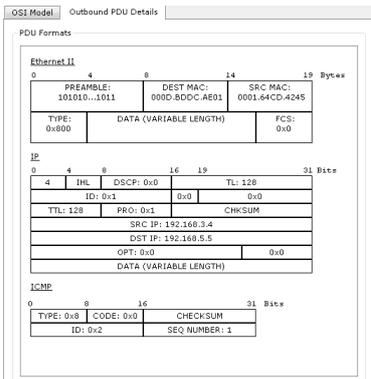


Рис. 4.58 Формат пакета ICMP-эхо-запроса

IP-адрес источника – 192.168.3.4. IP-адрес назначения – 192.168.5.5. Тип ICMP-сообщения – 8 (эхо-запрос).

Когда запрос приходит в сеть назначения, то маршрутизатор определяет MAC-адрес получателя, если такого нет в ARP-таблице маршрутизатора. Таким образом, снова решается задача разрешения локального адреса (рис. 4.59).

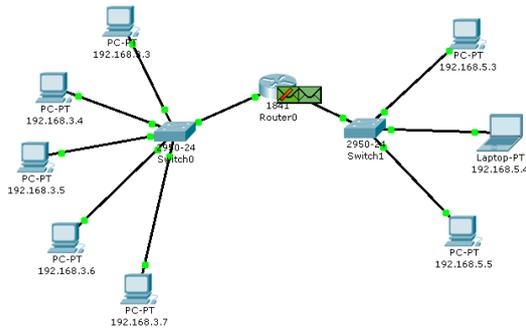


Рис. 4.59 Вид рабочей области

Маршрутизатор вынужден сперва узнать физический адрес получателя, прежде чем он сможет отправить ring-запрос по назначению, поэтому пакет с ring-запросом, пришедший на маршрутизатор, отклонен.

Новый ARP-запрос отправляется широковещательным сообщением от маршрутизатора, содержит его IP-адрес и MAC-адрес (рис. 4.60). IP-адрес назначения – узел 192.168.5.5.

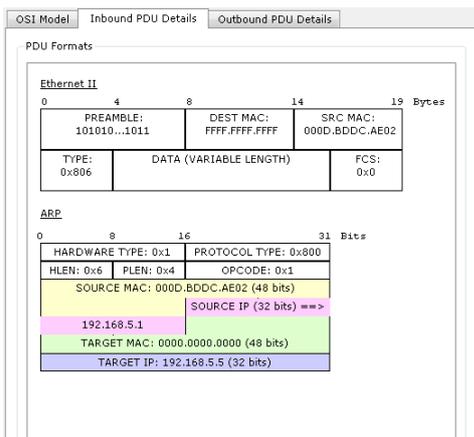


Рис. 4.60 Формат пакета ARP-запроса

Узлы подсети, которым пакет не предназначен, его игнорируют (рис. 4.61).

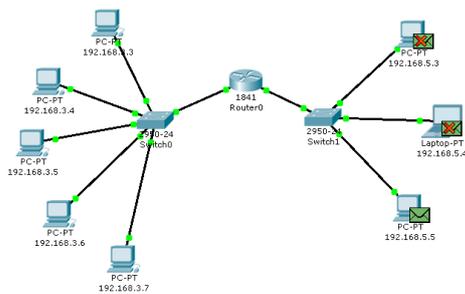


Рис. 4.61 Вид рабочей области

Узел 192.168.5.5. формирует ARP-ответ и отправляет его обратно маршрутизатору (рис. 4.62), указав свой MAC-адрес, о чем свидетельствует содержимое пакета (рис. 4.63).

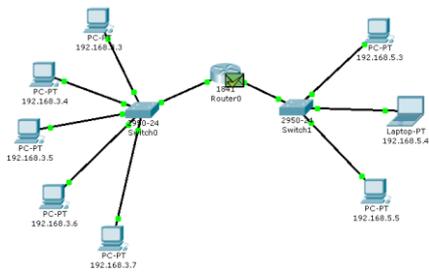


Рис. 4.62 Вид рабочей области

После того, как маршрутизатор определил MAC-адрес получателя входящего ping-запроса, он посылает ICMP-ответ маршрутизатору хоста отправителя. (В данном случае это тот же маршрутизатор Router0).

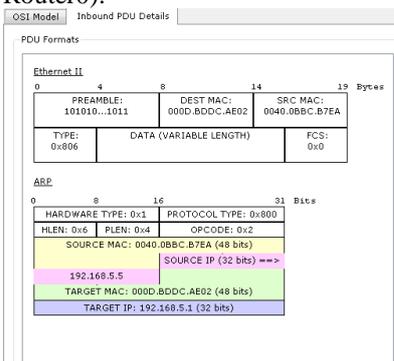


Рис. 4.63 Формат пакета ARP-ответа

Узел 192.168.3.4. снова пытается отправить ping-запрос во внешнюю сеть узлу 192.168.5.5. Его маршрут должен лежать через коммутатор Switch0, маршрутизатор Router0, коммутатор Switch1 и достигнуть узла назначения (рис. 4.64). Проследите маршрут пакета самостоятельно.

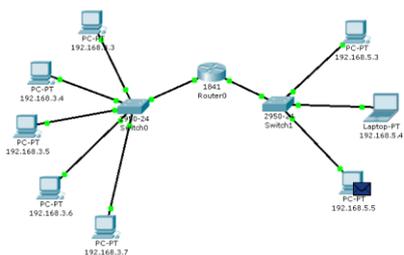


Рис. 4.64 Вид рабочей области

Узел формирует ping-ответ, который отправляется обратно узлу 192.168.3.4 (рис. 4.65).

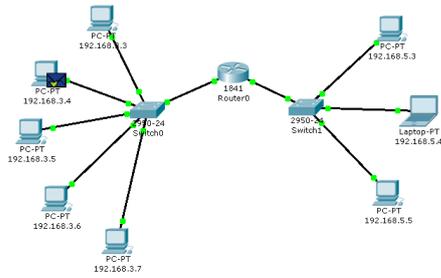


Рис. 4.65 Вид рабочей области

Посмотрите содержимое пакета ping-ответа, пришедшего на хост 192.168.3.4 (рис. 4.66).

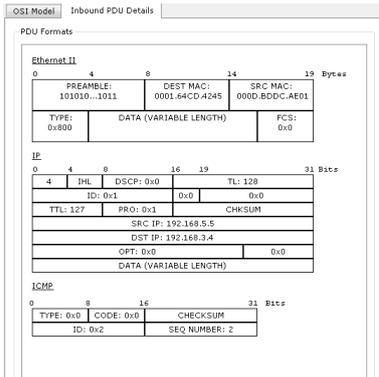


Рис. 4.66 Формат пакета ICMP-эхо-ответа

IP-адрес источника – 192.168.5.5. IP-адрес назначения – 192.168.3.4. Тип ICMP-сообщения – 0 (эхо-ответ).

Посмотрите ping-ответ в командной строке хоста 192.168.3.4 (рис. 4.67).

```

Command Prompt
Packet Tracer PC Command Line 1.0
PC>ping 192.168.5.5

Pinging 192.168.5.5 with 32 bytes of data:

Request timed out.
Reply from 192.168.5.5: bytes=32 time=8ms TTL=127
Reply from 192.168.5.5: bytes=32 time=8ms TTL=127
Reply from 192.168.5.5: bytes=32 time=8ms TTL=127

Ping statistics for 192.168.5.5:
    Packets: Sent = 4, Received = 3, Lost = 1 (25% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 8ms, Maximum = 8ms, Average = 8ms
  
```

Рис. 4.67 Вывод программы ping

Маршрут пакета можно посмотреть с помощью команды tracert. Выполним эту команду, например, в командной строке компьютера 192.168.3.5 (рис. 4.68):

```

PC>tracert 192.168.5.4

Tracing route to 192.168.5.4 over a maximum of 30 hops:

  0  4 ms    4 ms    4 ms    192.168.3.1
  1  8 ms    8 ms    8 ms    192.168.5.4

Trace complete.
  
```

Рис. 4.68 Вывод программы tracert

На пути пакета до хоста 192.168.5.4 один промежуточный маршрутизатор.

7. Посылка ping-запроса на несуществующий хост

Отправим ping-запрос на несуществующий адрес в сеть 192.168.5.0/24.

Откроем программу “Command Prompt” на узле 192.168.3.7 и попробуем отправить ping-запрос на несуществующий хост с IP-адресом 192.168.5.6 (рис. 4.69).

```
Command Prompt
Packet Tracer PC Command Line 1.0
PC>ping 192.168.5.6

Pinging 192.168.5.6 with 32 bytes of data:
```

Рис. 4.69 Командная строка узла 192.168.3.7

ARP-таблица на узле-источнике не содержит соответствующей записи о MAC-адресе узла 192.168.5.6, поэтому формируется ARP-запрос (рис. 4.70).

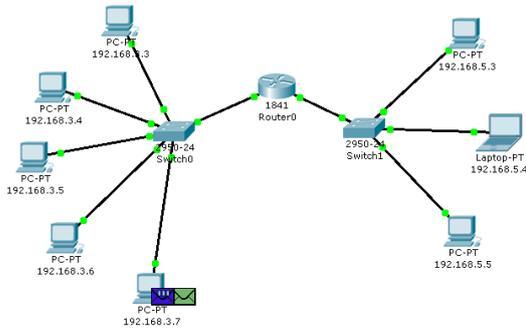


Рис. 4.70 Вид рабочей области

Все узлы игнорируют пакет, кроме маршрутизатора, которому этот пакет предназначался (рис. 4.71).

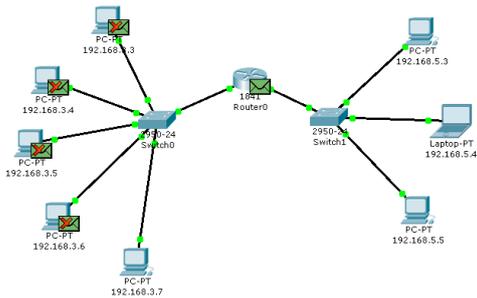


Рис. 4.71 Вид рабочей области

Узел 192.168.3.7 получает ARP-ответ с MAC-адресом маршрутизатора. Теперь, зная его аппаратный адрес, хост отправляет ping-запрос на узел 192.168.5.6 (рис. 4.72).

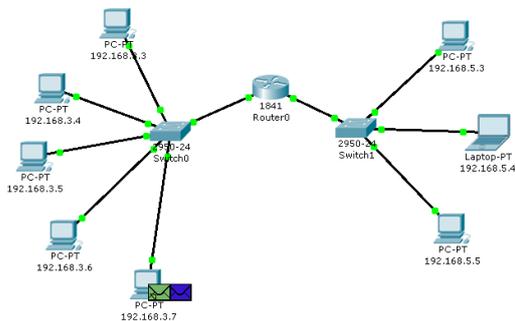


Рис. 4.72 Вид рабочей области

Маршрутизатор пришедший пакет уничтожает, т.к. не может его перенаправить на указанный адрес, потому что соответствующего MAC-адреса он «не знает». В связи с этим маршрутизатор формирует ARP-запрос по адресу 192.168.5.6 (рис. 4.73).

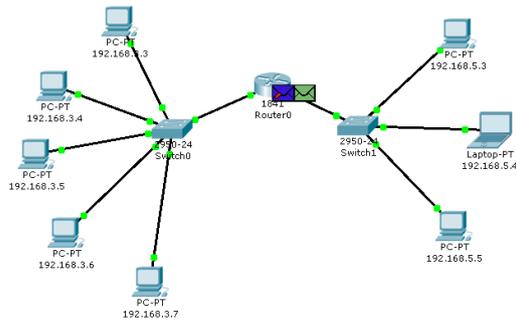


Рис. 4.73 Вид рабочей области

Все узлы подсети игнорируют пакет, потому что IP-адрес в запросе не соответствует их собственным (рис. 4.74). Маршрутизатор ни какого ответа ни от кого не получает.

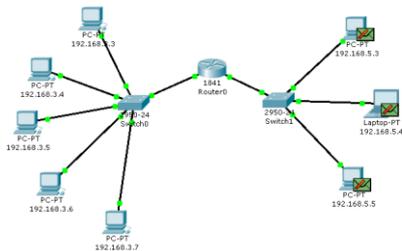


Рис. 4.74 Вид рабочей области

Процедура прохождения пакетов повторяется в течение всего сценария симуляции: маршрутизатор по-прежнему «не знает» MAC-адрес указанного в ping-запросе IP-адреса 192.168.5.6 и продолжает рассылать ARP-запросы. Ни один из узлов подсети на эти запросы не реагирует. Не получив ответа, маршрутизатор и сам «молчит», никак не уведомляя об ошибке хост-источник ping-запроса.

Примечание: на самом деле в данном случае маршрутизатору следует отправить ICMP-сообщение «хост недостижим»: сообщение типа 3 с кодом 1. Однако проведенный эксперимент с теорией разошелся.

Посмотрим ответ на ping-запрос в командной строке узла-источника 192.168.3.7: «превышено время ожидания» (рис. 4.75).

```

Command Prompt
PC>ping 192.168.5.6

Pinging 192.168.5.6 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 192.168.5.6:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
  
```

Рис. 4.75 Вывод программы ping

Попробуем отправить ping-запрос, содержащий IP-адрес узла, в сеть, на которую нет маршрута.

Откроем программу “Command Promt” на узле 192.168.3.6 и попробуем отправить ping-запрос на несуществующий хост с IP-адресом 192.168.6.6 (рис. 4.76).

```

Command Prompt

Packet Tracer PC Command Line 1.0
PC>ping 192.168.6.6

Pinging 192.168.6.6 with 32 bytes of data:
  
```

Рис. 4.76 Командная строка узла 192.168.3.6

Так как ARP-таблица узла-источника соответствующей записи не имеет, формируется ARP-запрос на заданный узел с IP-адресом 192.168.6.6 (рис. 4.77).

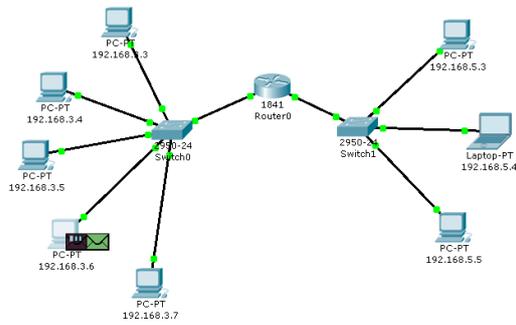


Рис. 4.77 Вид рабочей области

Все узлы игнорируют пакет, кроме маршрутизатора, которому этот пакет предназначался (рис. 4.78).

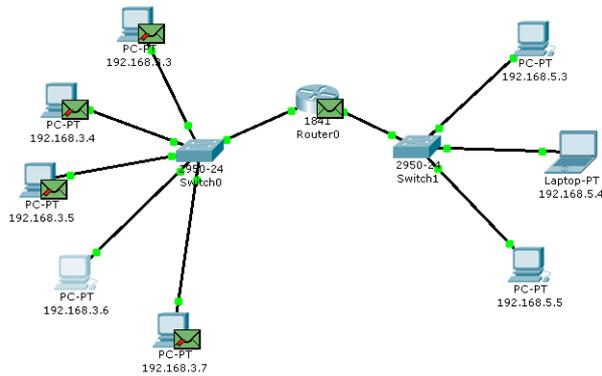


Рис. 4.78 Вид рабочей области

Узел 192.168.3.6 получает ARP-ответ с MAC-адресом маршрутизатора. Теперь, зная его аппаратный адрес, хост отправляет ping-запрос (рис. 4.79).

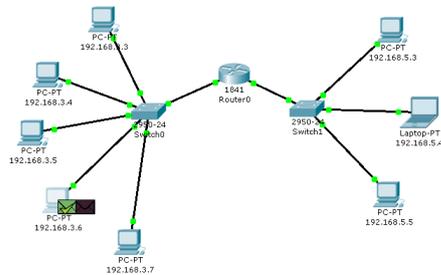


Рис. 4.79 Вид рабочей области

Когда ping-запрос попадает на маршрутизатор, тот не может его перенаправить на какой из своих интерфейсов, т.к. IP-адреса его интерфейсов не совпадают с тем адресом, который указан в ping-запросе. Соответственно, этот пакет уничтожается и формируется новое ICMP-сообщение (рис. 4.80).

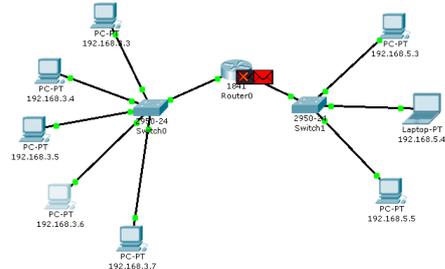


Рис. 4.80 Вид рабочей области

Посмотрим содержимое пакета, сформированного маршрутизатором (рис. 4.81).

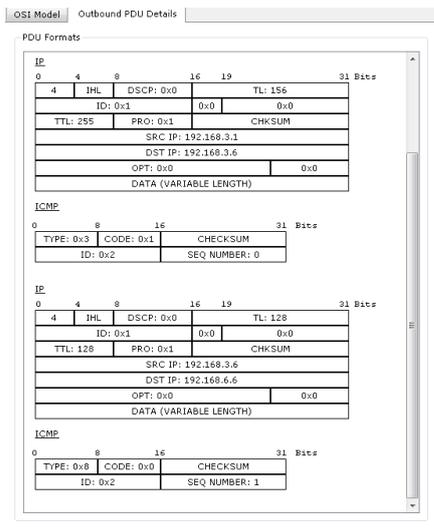


Рис. 4.81 Формат пакета ICMP «хост недоступен»

IP-адрес источника – 192.168.3.1. IP-адрес назначения – 192.168.3.6. Тип ICMP-сообщения – 3 с кодом 1, что означает «хост недоступен». Этот пакет приходит на узел 192.168.3.6.

Результат ping-запроса в командной строке узла 192.168.3.6: «хост назначения недоступен» (рис. 4.82).

```

Command Prompt
Packet Tracer PC Command Line 1.0
PC>ping 192.168.6.6

Pinging 192.168.6.6 with 32 bytes of data:

Reply from 192.168.3.1: Destination host unreachable.

Ping statistics for 192.168.6.6:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
  
```

Рис. 4.82 Вывод программы ping

Таким образом, маршрутизатор «ответил» на ping-запрос, для которого у него не было соответствующего маршрута, новым ICMP-сообщением «хост недоступен».

Примечание: корректно ли отреагировал маршрутизатор в данной ситуации, отправив на хост-источник ping-запроса ICMP-сообщение «хост недоступен»? Чтобы ответить на этот вопрос, необходимо обратиться к спецификации протокола ICMP RFC 792 и ознакомиться с другими типами ICMP-сообщений. [Электронный ресурс]. URL: <http://tools.ietf.org/html/rfc792>.

8. Индивидуальные задания

В соответствии с вариантом отфильтруйте ARP и ICMP сообщения для указанных пар «источник – приемник». В каждом варианте предусмотрены 2 варианта ping-запроса: внутри сети и во внешнюю сеть. С помощью команды tracerp посмотрите маршрут пакета, адресованного во внешнюю сеть.

В отчете для каждого теста приведите маршруты пакетов, их содержимое и объясните полученные результаты.

Варианты заданий представлены в таблице 1.

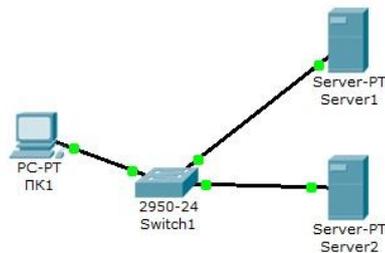
Таблица 1

| Вариант | Источник | Приемник |
|---------|-------------|-------------|
| 1 | 192.168.3.3 | 192.168.3.4 |
| | 192.168.3.4 | 192.168.3.6 |
| 2 | 192.168.3.4 | 192.168.3.7 |
| | 192.168.3.5 | 192.168.5.3 |
| 3 | 192.168.3.5 | 192.168.3.6 |
| | 192.168.3.6 | 192.168.3.7 |
| 4 | 192.168.3.6 | 192.168.5.4 |
| | 192.168.3.7 | 192.168.3.4 |
| 5 | 192.168.3.3 | 192.168.3.7 |
| | 192.168.3.7 | 192.168.5.5 |
| 6 | 192.168.5.3 | 192.168.5.4 |

| | | |
|----|----------------------------|----------------------------|
| | 192.168.3.6 | 192.168.3.4 |
| 7 | 192.168.3.3 192.168.3.5 | 192.168.5.3 192.168.3.7 |
| 8 | 192.168.3.3 192.168.3.4 | 192.168.5.4 192.168.3.5 |
| 9 | 192.168.3.4 192.168.3.5 | 192.168.5.3 192.168.3.4 |
| 10 | 192.168.5.4 192.168.3.6 | 192.168.5.5 192.168.3.3 |
| 11 | 192.168.3.4 192.168.3.7 | 192.168.5.3 192.168.5.4 |
| 12 | 192.168.3.5 192.168.3.6 | 192.168.5.5 192.168.3.7 |
| 13 | 192.168.3.5 192.168.3.7 | 192.168.5.4 192.168.3.3 |
| 14 | 192.168.3.6 192.168.3.7 | 192.168.5.3 192.168.5.5 |

Практикум 4. DNS и Web сервер

1. Создайте схему сети, как представлено на рисунке:

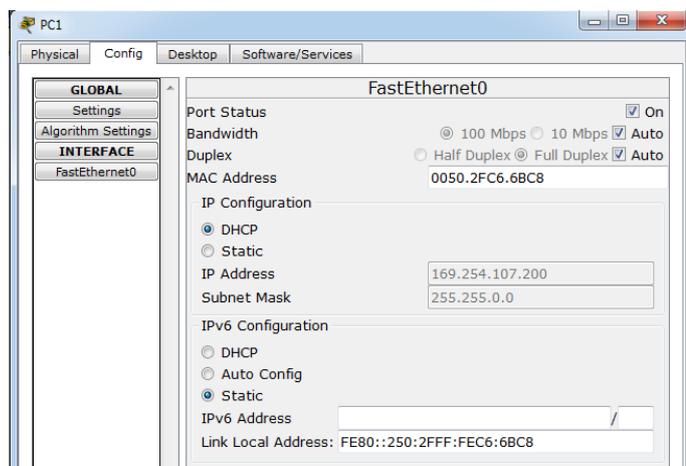


Настроить сеть следующим образом:

- 1 - Server1 – DNS и Web сервер;
- 2 - Server2 – DHCP сервер;
- 3 - Компьютер ПК1 получает параметры протокола TCP/IP с DHCP сервера и открывает сайт www.rambler.ru на Server1.

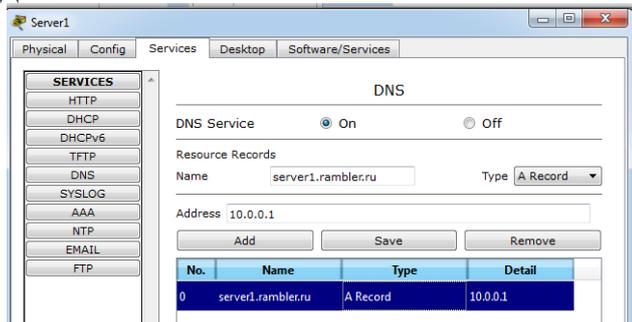
Выполнение:

- Задайте параметры протокола TCP/IP на ПК1 и серверах. Для этого войдите в конфигурацию ПК1 и установите настройку IP через DHCP сервер.

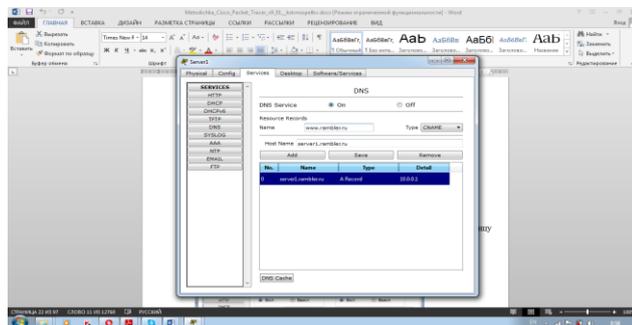


- Задайте в конфигурации серверов следующие настройки IP:
 Server1: IP адрес – 10.0.0.1, маска подсети – 255.0.0.0
 Server2: IP адрес – 10.0.0.2, маска подсети – 255.0.0.0
Серверам допишите DNS сервер - 10.0.0.1

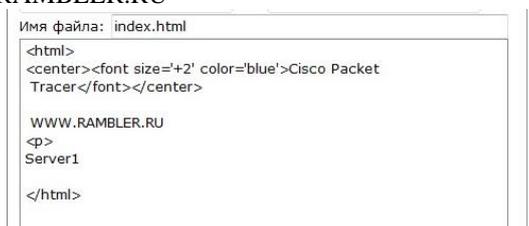
- Настройте службу DNS на Server1. Для этого на вкладке Сервисы Server1 в разделе DNS и задайте две ресурсные записи в прямой зоне DNS:
 1 – в ресурсной записи типа A свяжите доменное имя компьютера с его IP адресом и нажмите кнопку **Добавить**



- 2 – в ресурсной записи типа CNAME свяжите псевдоним сайта с компьютером



- В конфигурации Server1 водите на вкладку HTTP и задайте стартовую страницу сайта WWW.RAMBLER.RU



- Перейдите в командную строку на Server1 и проверьте работу службы DNS. Для проверки прямой зоны DNS сервера введите команду

SERVER>nslookup www.rambler.ru

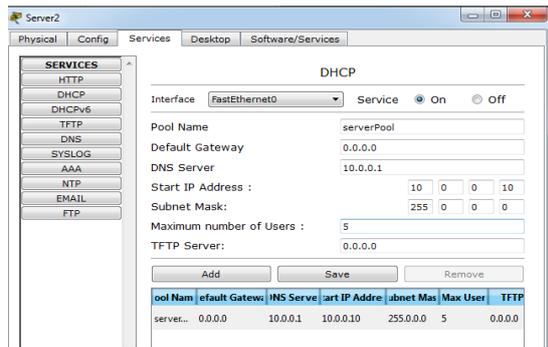
- Если все правильно, то вы получите отклик, представленный на рисунке с указанием полного доменного имени DNS сервера в сети и его IP адрес.

```
SERVER>nslookup www.rambler.ru
Server: [10.0.0.1]
Address: 10.0.0.1

Non-authoritative answer:
Name:   server1.rambler.ru
Address: 10.0.0.1

Aliases: server1.rambler.ru
SERVER>
```

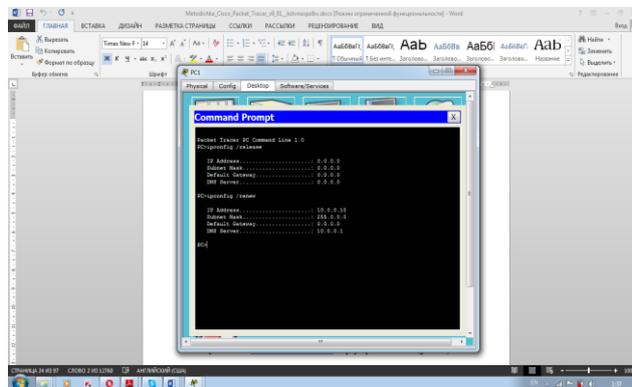
- Настройте DHCP службу на Server2. Для этого на вкладке в конфигурация Server2 и на вкладке DHCP настройте службу



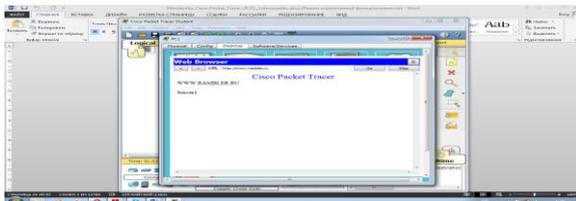
- В конфигурации хоста PC1 на **рабочем столе** и в командной строке сконфигурируйте протокол TCP/IP.

Командой **PC>ipconfig /release**

Сбросьте старые параметры IP адреса, а командой: **PC>ipconfig /renew** получите новые параметры с DHCP сервера



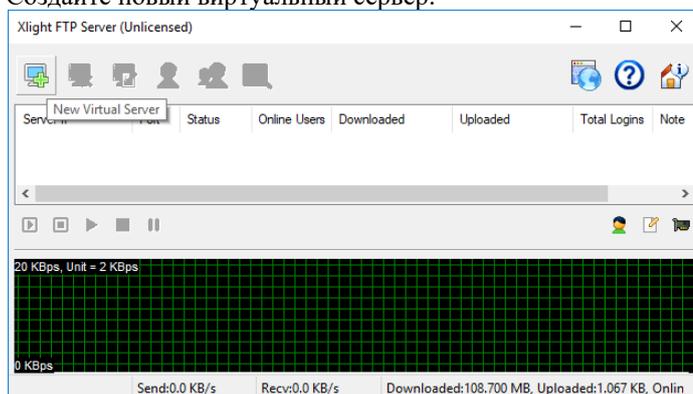
Откройте сайт **WWW.RAMBLER.RU** в браузере на клиенте



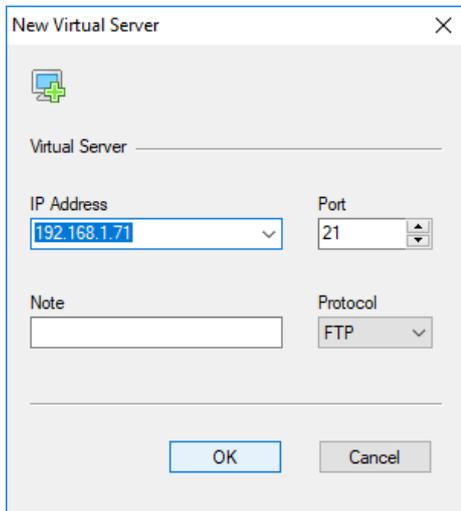
Практикум 5. Протоколы прикладного уровня

Задание 1. FTP сервер.

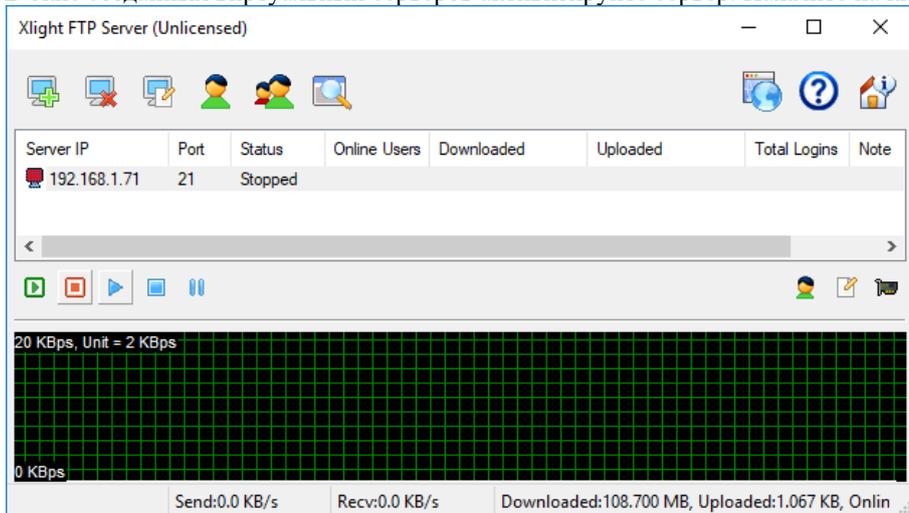
1. Скачайте «Xlight - Windows FTP and SFTP Server» со страницы <https://www.xlightftpd.com/> (портативную версию) или из приложения к лабораторной работе.
2. Создайте новый виртуальный сервер.



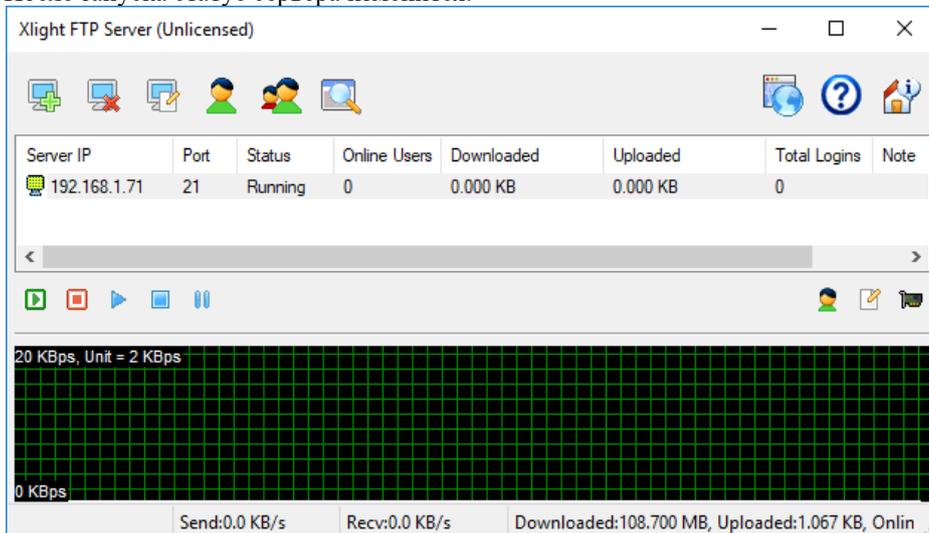
3. Настройте сервер



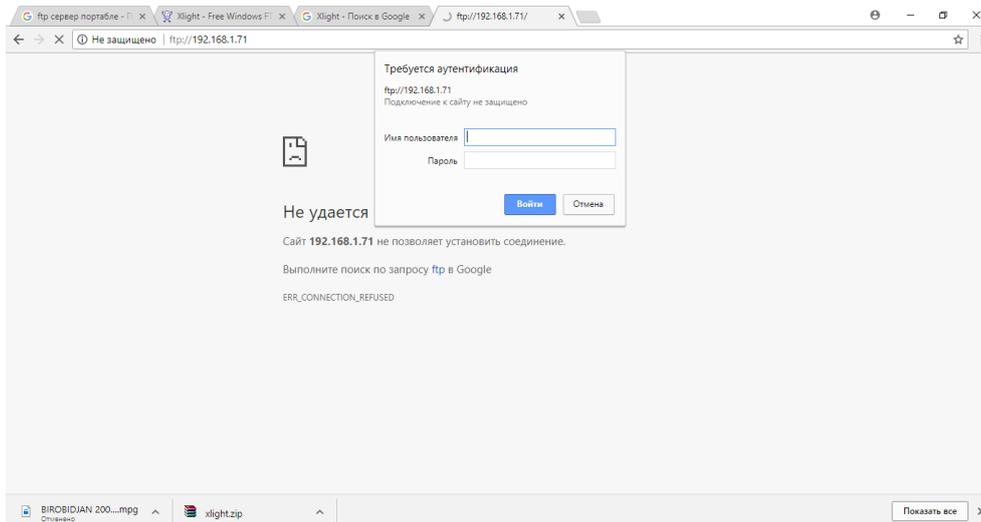
4. В окне созданных виртуальный серверов активизируйте сервер. Нажмите на кнопку – Start Server



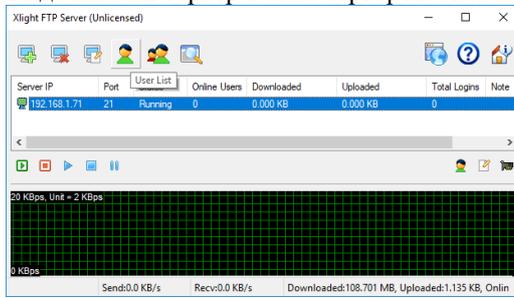
5. После запуска статус сервера изменится.



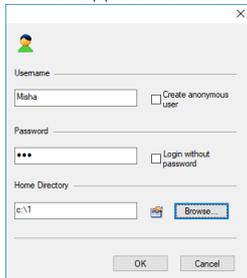
6. Проверьте работу сервера. Запустите браузер на любом или текущем узле сети. Наберите ftp://<ip-address server>
7. Если появится окно аутентификации, значит сервер настроен правильно.



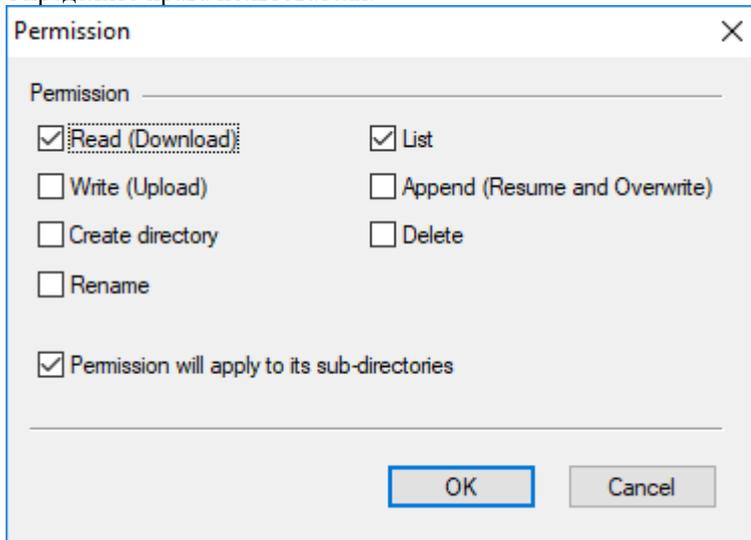
8. Создайте на диске папку с файлами, к которым будет давать доступ сервер: 2 текстовых, 2 графических файла. Внутри папки создайте еще две папки, в каждую из которых поместите по одному текстовому файлу.
9. Создадим пользователей, которые смогут получить доступ к вашей папке.
10. Выделив ваш сервер в окне серверов. Нажмите на кнопку – User List



11. Создайте пользователя: определите логин, пароль и доступ к папке.

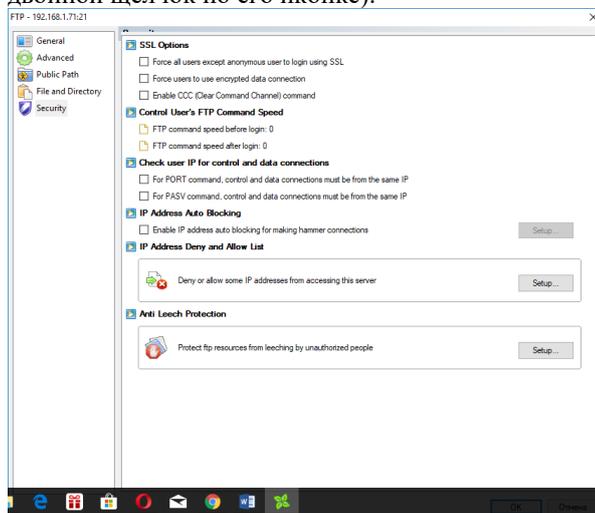


12. Определите права пользователя.



13. Пройдите аутентификацию через браузер

14. Создайте еще трех пользователей: анонимного пользователя, пользователя которому запрещен просмотр поддиректориев, и пользователь которому запрещено открывать файлы (только просмотр списка файлов).
15. Посмотрите содержимое логов (найдите соответствующую кнопку на главном интерфейсе). Опишите в отчете основные команды которые посылает клиент и сервер друг другу при работе.
16. Запретите доступ с ip адреса вашего соседнего компьютера (вызовите опции вашего сервера – двойной щелчок по его иконке).



17. Опишите в отчете какие еще функции выполняет сервер.
18. Осуществите доступ к вашему серверу из командной строки.

```

Командная строка - ftp 192.168.1.71
Microsoft Windows [Version 10.0.16299.248]
(c) Корпорация Майкрософт (Microsoft Corporation), 2017. Все права защищены.

C:\Users\Илья>ftp 192.168.1.71
Связь с 192.168.1.71.
220 Xlight FTP Server 3.8 ready...
530 Not login, please login first
Пользователь (192.168.1.71:(none)): w
331 Password required for w
Пароль:
230 Login OK
ftp> dir
200 PORT command successful
150 Opening ASCII mode data connection for /bin/ls (2119 bytes).
drwx-rw-rw- 1 ftp ftp          0 Sep 24  2017 $AV_ASW
drwx-rw-rw- 1 ftp ftp          0 Sep 03  2017 $GetCurrent
drwx-rw-rw- 1 ftp ftp          0 Sep 04  2017 $Recycle.Bin
drwx-rw-rw- 1 ftp ftp          0 Feb 11  19:06 1
drwx-rw-rw- 1 ftp ftp          0 Jan 25  12:33 1C
drwx-rw-rw- 1 ftp ftp          0 Nov 09  2016 Boot
-r--r--r-- 1 ftp ftp    389400 Oct 15  2016 bootmgr
-rw-rw-rw- 1 ftp ftp          1 Jul 16  2016 BOOTNXT
-rw-rw-rw- 1 ftp ftp    11577 May 24  2017 devlist.txt
drwx-rw-rw- 1 ftp ftp          0 May 24  2017 Documents and Settings
-rw-rw-rw- 1 ftp ftp    107520 Feb 25  11:09 DUMP900b.tmp
drwx-rw-rw- 1 ftp ftp          0 May 23  2017 eSupport
-rw-rw-rw- 1 ftp ftp          9 May 24  2017 Finish.log
drwx-rw-rw- 1 ftp ftp    2975358976 Mar 04 09:59 hiberfil.sys
drwx-rw-rw- 1 ftp ftp          0 Sep 22  2017 MediaServer_Temp
-rw-rw-rw- 1 ftp ftp    921636 Jan 18  22:53 PA7302.DAT

```

Используйте команду dir для просмотра содержимого каталога

Для смены каталога используйте команду CD <path>

Сделайте скриншот вашего каталога.

При помощи команды get вы можете скачать файл (он помещается в папку вашего профиля). Скачайте любой файл. Сделайте скриншот.

При помощи команды put вы можете отправить любой файл из папки вашего профиля в папку сервера. Отправьте любой файл. Сделайте скриншот.

Закройте соединение при помощи команды bye.

19. Какие еще существуют программные комплексы при помощи которых можно организовать в локальной сети FTP сервер?
20. Найдите FTP каталоги размещенные в сети интернет. Произведите их поиск в любой из поисковых систем по фразе «FTP адреса». Приведите примеры каталогов (не менее 2-х) в которые разрешен анонимный доступ.
21. В файловой поисковой системе FileSearch.ru осуществите поиск изображения кремля в ftp каталогах.

Задание 2. HTTP – запросы

1. Используя сервис <https://www.bertal.ru/> - просмотрите HTTP заголовки при обращении к сайту mail.ru. В отчете поясните значения некоторых заголовков (не менее 3-х в запросе и ответе сервера)

2. Осуществите поисковый запрос в mail.ru. Разберите на составляющую сформированную поисковую строку в адресном окне.
3. Перечислите, в отчете, основные методы протокола и их назначение.

Практикум 6. Создаем веб-браузер

WebBrowser предоставляет функции интернет-браузера, позволяя загружать и отображать контент из сети интернет. В то же время важно понимать, что данный элемент не является полноценным веб-браузером, и возможности по его настройке и изменению довольно ограничены.

Рассмотрим основные его свойства:

- AllowWebBrowserDrop: при установке для данного свойства значения true можно будет с помощью мыши переносить документы в веб-браузер и открывать их.
- CanGoBack: определяет, может ли веб-браузер переходить назад по истории просмотров
- CanGoForward: определяет, может ли веб-браузер переходить вперед
- Document: возвращает открытый в веб-браузере документ
- DocumentText: возвращает текстовое содержание документа
- DocumentTitle: возвращает заголовок документа
- DocumentType: возвращает тип документа
- IsOffline: возвращает true, если отсутствует подключение к интернету
- ScriptErrorsSuppressed: указывает, будут ли отображаться ошибки javascript в диалоговом окне
- ScrollBarsEnabled: определяет, будет ли использоваться прокрутка
- URL: возвращает или устанавливает URL документа в веб-браузере

Кроме того, WebBrowser содержит ряд методов, которые позволяют осуществлять навигацию между документами:

- GoBack(): осуществляет переход к предыдущей странице в истории навигации (если таковая имеется)
- GoForward(): осуществляет переход к следующей странице в истории навигации
- GoHome(): осуществляет переход к домашней странице веб-браузера
- GoSearch(): осуществляет переход к странице поиска
- Navigate: осуществляет переход к определенному адресу в сети интернет

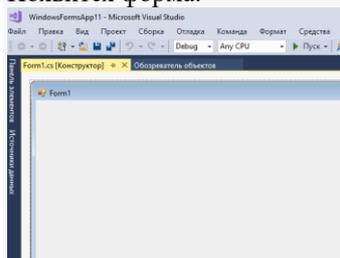
Таким образом, чтобы перейти к определенному документу, надо использовать метод Navigate:

```
// перейти к адресу в интернете
webBrowser1.Navigate("http://google.com");
// открыть документ на диске
webBrowser1.Navigate("C://Images//24.png");
```

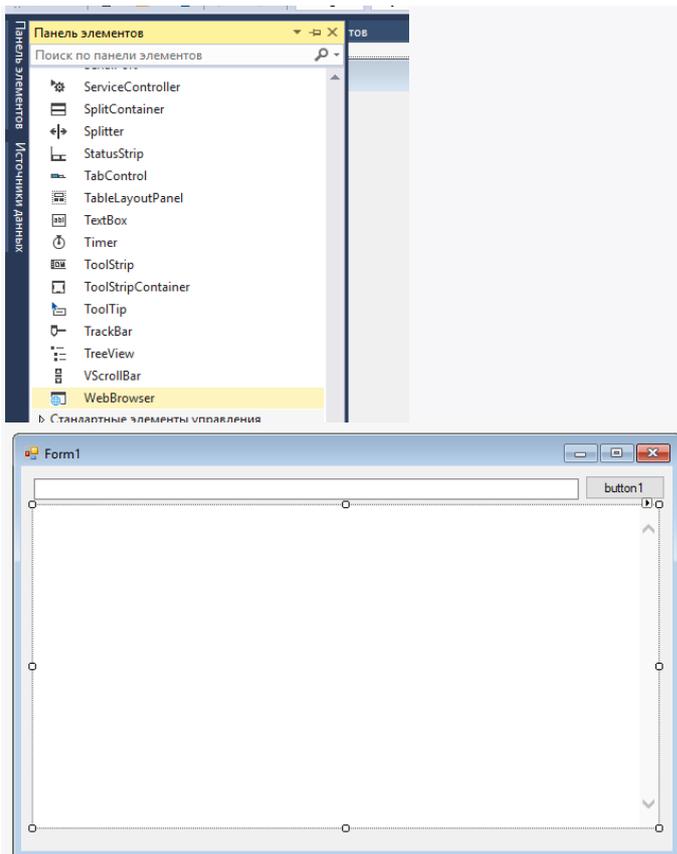
Создадим небольшой веб-браузер.

1. Откроем Microsoft Visual Studio
2. Файл – Создать – Проект – C# - Приложение Windows Forms

Появится форма.



Поместим на форму элементы WebBrowser, TextBox (в него будем вводить адрес) и Button. Элементы находятся на панели инструментов.



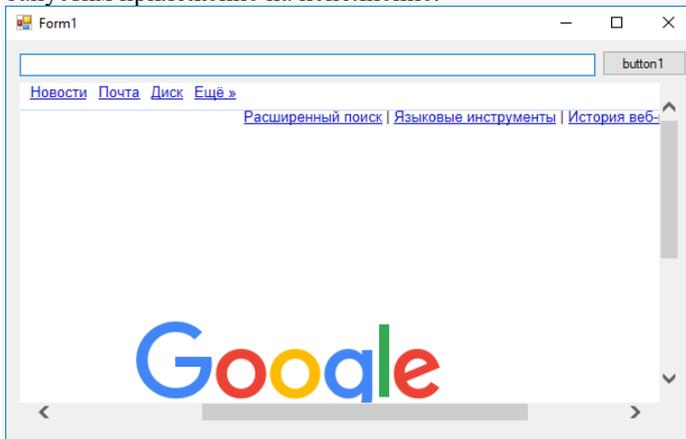
В файле формы (через контекстное меню – перейти к коду) пропишем следующий код:

```
public partial class Form1 : Form
{
    public Form1()
    {
        InitializeComponent();
        // запретим сообщения об ошибках скриптов
        webBrowser1.ScriptErrorsSuppressed = true;
        // установка начального адреса
        webBrowser1.Url = new Uri("http://google.com");
        button1.Click += button1_Click;
    }
}
```

Сделаем двойной щелчок по кнопке и впишем код.

```
private void button1_Click(object sender, EventArgs e)
{
    webBrowser1.Navigate(textBox1.Text);
}
}
```

Запустим приложение на исполнение.



Наберите в строке адреса – новый интернет адрес и нажмите кнопку.

Дополнительно.

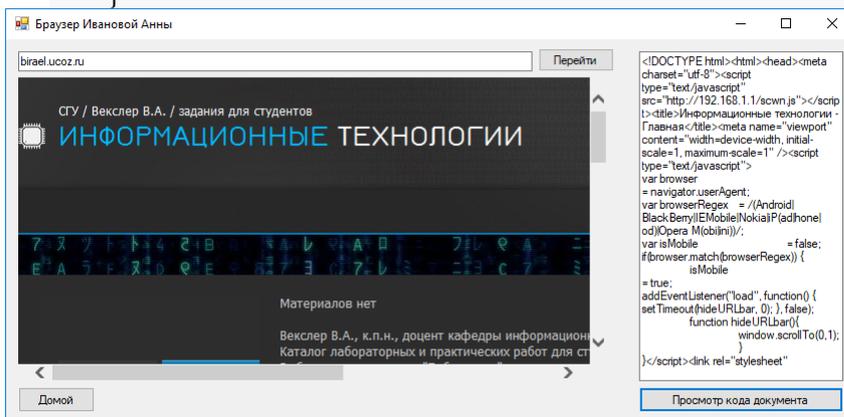
1. Изменим название кнопки на «Перейти». В окне конструктора вызовем свойства кнопки и в поле Текст введем новое название.
2. Изменим название формы с Form1 на «Браузер Фамилия Имя» (Фамилия и Имя – студента, делающего лабораторную работу). В окне конструктора вызовем свойства формы и в поле Текст введем новые данные.

3. Создадим кнопку для возврата к домашней странице.

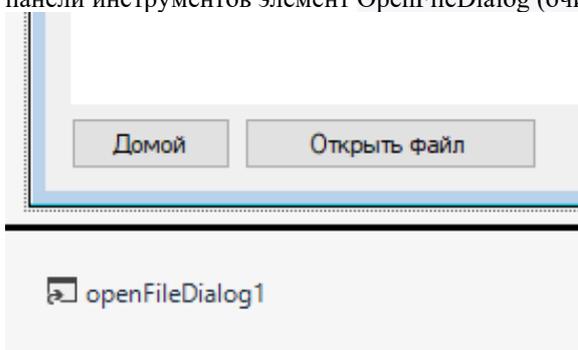
```
private void button2_Click(object sender, EventArgs e)
{
    webBrowser1.GoHome();
}
```

4. Добавьте большой текстовый блок (TextBox, свойство Multiline). Внизу разместите кнопку «Просмотр кода документа». Для кнопки напишите код:

```
private void button3_Click(object sender, EventArgs e)
{
    textBox2.Text = webBrowser1.DocumentText;
}
```



5. Добавим кнопку для открытия в окне файлов необходимых нам типов. Установим новую кнопку «Открыть файл» и перетащим с панели инструментов элемент OpenFileDialog (очистите свойство FileName).



Для кнопки напишем код.

```
private void button4_Click(object sender, EventArgs e)
{
    openFileDialog1.Filter = "Text files(*.txt)*.txt|Html files(*.html)*.html|Htm files(*.htm)*.htm";
    if (openFileDialog1.ShowDialog() == DialogResult.Cancel)
        return;
    // получаем выбранный файл
    string filename = openFileDialog1.FileName;
    webBrowser1.Url = new Uri(openFileDialog1.FileName);
}
```

«DNS»

DNS-сервер хранит таблицу, отображающую имена хостов на IP-адреса для всех известных ему компьютеров, а также IP-адреса других DNS-серверов, где можно искать имена хостов, которые ему неизвестны. Локальный компьютер должен всегда знать, по крайней мере, один DNS-сервер. Сетевые администраторы конфигурируют эту информацию при настройке компьютера.

Прежде чем отправлять запрос, компьютер сначала спрашивает у DNS-сервера IP-адрес, соответствующий введенному имени хоста. Получив корректный IP-адрес, компьютер может отправить ему запрос через сеть. Все это нормально работает "за кулисами", пока пользователь путешествует по Интернету.

В .NET Framework предлагается множество классов, которые помогают в процессе поиска IP-адресов и нахождении информации о компьютерах-хостах.

Класс Dns способен взаимодействовать с DNS-сервером по умолчанию для извлечения IP-адреса. Он имеет два важных статических метода — **GetHostEntry()**, который использует DNS-сервер для получения деталей хоста по заданному его имени, и **GetHostByAddress()**, который также возвращает детали хоста, но на этот раз используя IP-адрес.

Класс IPHostEntry инкапсулирует информацию об определенном компьютере-хосте. Этот класс делает имя хоста доступным через свойство **HostName** (которое возвращает строку), а свойство **AddressList** возвращает массив объектов **IPAddress**.

Допишем в перечисление пространств имен

`using System.Net;`

Добавим кнопку «DNS».

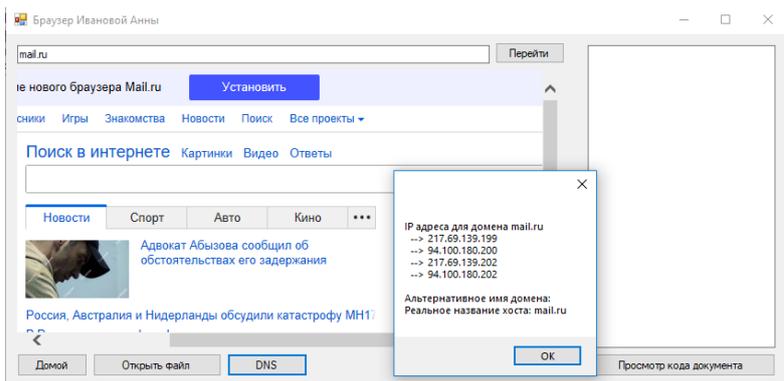
Код кнопки.

```
private void button5_Click(object sender, EventArgs e)
{
    string message = "IP адреса для домена " + textBox1.Text + "\n";
    IPHostEntry entry = Dns.GetHostEntry(textBox1.Text);

    foreach (IPAddress a in entry.AddressList)
        message += " --> " + a.ToString() + "\n";

    message += "\nАльтернативное имя домена: ";
    foreach (string aliasName in entry.Aliases)
        message += aliasName + "\n";

    message += "\nРеальное название хоста: " + entry.HostName;
    MessageBox.Show(message);
}
```



«Ping»

Добавим возможность пинговать адрес в доменной форме или в виде IP-адреса из строки адреса.

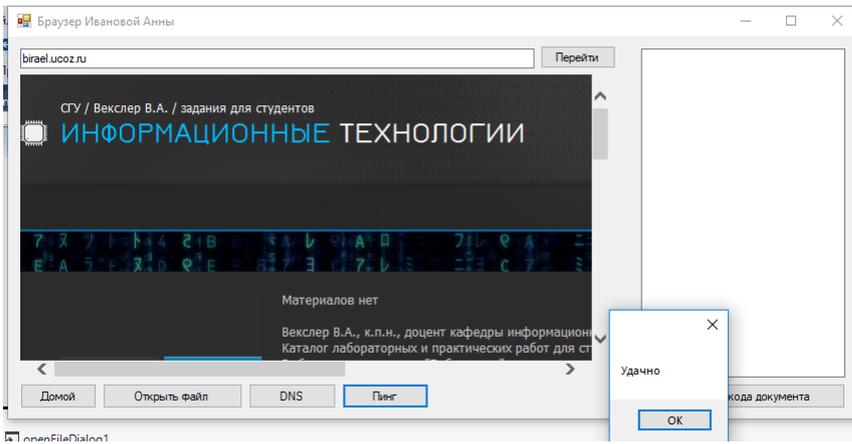
Установим кнопку «Пинг».

Добавим пространство имен.

`using System.Net.NetworkInformation;`

Напишем код для кнопки.

```
private void button6_Click(object sender, EventArgs e)
{
    PingReply pr;
    Ping ping = new Ping();
    pr = ping.Send(textBox1.Text, 2);
    if (pr.Status == IPStatus.Success)
        MessageBox.Show("Удачно");
    else
        MessageBox.Show("Неудачно");
}
```



«HTTP»

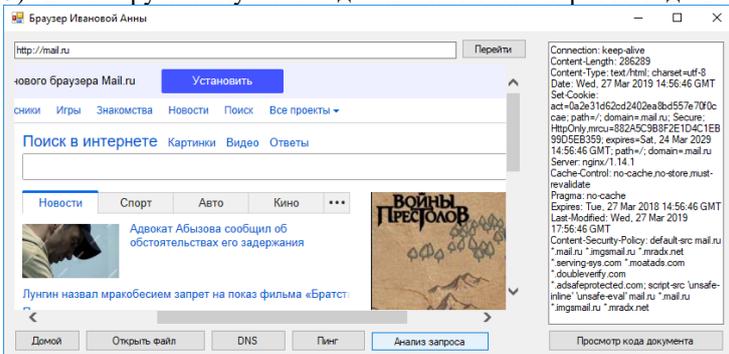
Компьютеры взаимодействуют по сети, используя различные сетевые протоколы, определяющие форматы и способ передачи данных. В WEB основным протоколом является HTTP.

Различные куски программы могут взаимодействовать друг с другом по этому протоколу. Такая программа называется распределенной. Компоненты такой программы могут состоять из служб, каких-то сервисов, клиентов и располагаться на различных серверах. Для того, чтобы использовать данный протокол, можно воспользоваться классами из пространства имен "System.Net". В данном пространстве имен есть 2 абстрактных класса: "WebRequest" (запрос) и "WebResponse" (ответ).

Как реализован механизм "Запрос-ответ": некий объект создаёт запрос и отправляет его на приёмник (можно назвать его сервером). Сервер принимает данные из запроса и отправляет ответ. Данные классы являются абстрактными, потому что данная схема используется не только протоколом HTTP, но и другими протоколами (ftp, file).

Разберем работу данного механизма на примере классов "HttpWebRequest" и "HttpWebResponse":

- 1) В первую очередь необходимо знать адрес сервиса или ресурса, на которой мы хотим обратиться. В начале адреса обычно идёт название протокола.
- 2) Создаём объект класса "HttpWebRequest".
- 3) Получаем объект класса "HttpWebResponse". Данный объект получается из объекта "HttpWebRequest" с использованием метода "GetResponse()". Как раз в тот момент, когда мы вызываем метод "GetResponse()", осуществляется запрос.
- 4) Анализируем код возврата, чтобы понять, что мы получили ответ. Необязательный шаг.
- 5) Анализируем полученные данные из потока через метод "GetResponseStream()".



Создадим кнопку «Анализ запроса».

Адрес запроса берется из textBox1, результат выводится в textBox2.

Добавим пространство имен.

`using System.IO;`

Код кнопки.

`private void button7_Click(object sender, EventArgs e)`

```
{
    try {
        // uri обязательно начинается с http
        string uri = textBox1.Text;
        // Если адрес непустой, то делаем запрос к этому ресурсу
        if (!string.IsNullOrEmpty(uri))
        {
            HttpWebRequest request = WebRequest.Create(uri) as HttpWebRequest;
            if (request != null)
```

```

    {
        // Запрашиваем данные методом GET
        request.Method = "GET";
        HttpResponseMessage response = request.GetResponse() as HttpResponseMessage;
        textBox2.Text = string.Empty;
        textBox2.Text = response.Headers.ToString();
        // Будем определять кодировку из заголовка ответа "Content-Type"
        string encoding = "utf-8";
        string ct = response.Headers["Content-Type"];
        if (ct != null)
        {
            // После "charset=" идёт кодировка
            encoding = ct.Substring(ct.IndexOf("charset=") + 8);
        }
        // Создаём StreamReader, в котором читаем все данные из потока в нужной кодировке
        StreamReader reader = new StreamReader(response.GetResponseStream(),
        Encoding.GetEncoding(encoding));

        textBox2.Text += reader.ReadToEnd();
        reader.Close();
    }
}
catch
{
    MessageBox.Show("uri обязательно начинается с http");
}
}

```

Практикум 7. Создание простейшего HTML-документа. Форматирование шрифта и абзаца

Цель работы:

- Научиться создавать простейший гипертекстовый документ средствами текстового редактора Блокнот.
- Научиться использовать теги форматирования шрифта и абзаца.

Ход работы:

1. Изучить теоретический материал. Письменно ответить на контрольные вопросы.
2. Выполните практикум

Контрольные вопросы.

1. Что представляет собой документ, написанный на языке HTML?
2. Что такое валидация?
3. Что такое теги?
4. Какие бывают теги?
5. Что такое исходный код?
6. Как обозначают начало и конец тега?
7. Что такое заголовок документа? Какими тегами он выделяется?
8. Что такое тело документа? Какими тегами оно выделяется?
9. Какими буквами следует писать теги, прописными или строчными?
10. Какой тег позволяет перейти на новую строку?
11. Какой тег используют для разбиения текста на параграфы?
12. Какие теги используют для выравнивания текста по левому, правому краю и центру?
13. Как выделить текст полужирным шрифтом?
14. Как выделить текст курсивом?

15. Как выделить текст подчеркиванием?
16. Какие теги изменяют размер шрифта?
17. Как оформить маркированные, нумерованные и вложенные списки?

Практикум

1. Создать файл с гипертекстовым документом:

- Запустить редактор Блокнот, ввести текст:

Приветствую Вас на моей первой web-страничке!

- Сохранить файл в созданной папке. При сохранении, в окне диалога **Сохранить как...** в строке **Тип файла:** выбрать вариант **Все файлы (*.*)**, а в строке **Имя файла** задать имя с расширением **.htm**, например **1_name.htm** (где **name** – ваше имя)
 - Закрывать документ, найти его пиктограмму в окне **Мой компьютер** или в окне программы **Проводник**.
 - Открыть файл. Проанализировать, с помощью какого приложения отображается файл и как выглядит введенная фраза.
2. Ввести теги, определяющие структуру html-документа:
- С помощью контекстного меню открыть файл с помощью редактора Блокнот. Ввести приведенные ниже теги, в разделе заголовка документа (между тегами **<TITLE>** **</TITLE>**) указать свою фамилию.

<HTML>

<HEAD> <TITLE> фамилия </TITLE>

</HEAD>

<BODY>

Приветствую Вас на моей первой web-страничке!

</BODY>

</HTML>

- **Сохранить** документ под тем же именем, обновить его отображение в браузере (выполнить **Вид/Обновить** или нажать кнопку **Обновить** на панели инструментов). Проанализировать произошедшие изменения в отображении документа.
 - **Проверьте код на валидацию. Исправьте при необходимости код.**
3. Отредактировать документ:

- Вызвать меню браузера **Вид/Просмотр HTML-кода** и добавить после текста **«Приветствую Вас на моей первой web-страничке!»** текст подписи:

Студент группы NNN Фамилия Имя

Сохранить документ (но не закрывать) и обновить его просмотр в браузере.

- Используя одиночный тег **
, отредактировать документ так, чтобы подпись начиналась с новой строки, а **Фамилия Имя – в следующей строке. Просмотреть в браузере новый вариант.

Внимание! После каждого изменения документ нужно сохранять, а просмотр в браузере начинать с обновления загрузки документа с помощью кнопки **«Обновить»** на панели инструментов.

4. Оформить фрагменты текста с помощью стилей **Заголовков**:

- Первую строку документа оформить стилем **Заголовок 1-го уровня** с помощью парного тега **<H1> ...</H1>**. Вторую строку оформить как **Заголовок 6-го уровня**, а третью как **Заголовок 4-го уровня**.
- Просмотреть документ в браузере, изменяя настройку отображения шрифтов (меню **Вид / Размер шрифта / Самый крупный, Средний, Мелкий и Самый мелкий**).

- Поменять стиль оформления первой строки на **Заголовок 2 уровня**, второй строки - на **Заголовок 5 уровня**, последней строки - на **Заголовок 3-го уровня**.
5. Выполнить форматирование шрифта:
- После строки **Фамилия Имя** добавить еще одну строку текста
Нас утро встречает прохладой
 - Оформить фразу по приведенному ниже образцу.

Нас утро встречает прохладой

В слове УТРО все буквы должны иметь **разные цвета**. В слове ПРОХЛАДОЙ оформить буквы ПРО – **красным** цветом, ОЙ – **синим**.

- Оформить строку с подписью (Студент группы NNN **Фамилия Имя**) курсивом, размер шрифта задать относительным изменением. Использовать теги **** и **<I>**
 - Просмотреть полученный документ в браузере.
6. Выполнить форматирование абзацев:
- Создать новый документ **2_name.htm**, сохранить его в той же рабочей папке.
 - Ввести текст (использовать копирование текста из документа **1_name.htm**):

```
<HTML>
  <HEAD> <TITLE> фамилия </TITLE>
</HEAD>
<BODY>
Приветствую Вас на моей второй web-страничке! <BR> Монолог Гамлета
</BODY>
</HTML>
```

- Выровнять текст **по центру**.
 - Ввести текст:
Быть или не быть - вот в чем вопрос. Что благороднее: сносить удары неистовой судьбы - или против моря невзгод вооружиться, в бой вступить. И все покончить разом...
 - Оформить выравнивание абзаца **по ширине**.
 - Ограничить абзац горизонтальными разделительными линиями сверху и снизу, используя тег **<HR>**.
 - Скопировать монолог и разбить его на абзацы. Выровнять **по центру**.

Быть или не быть - вот в чем вопрос.
 Что благороднее: сносить удары
 Неистойвой судьбы - или против моря
 Невзгод вооружиться, в бой вступить
 И все покончить разом...
 - Раскрасьте строки по своему выбору
 - Добавьте строку где бы присутствовал текст нижнего и верхнего индекса.
 - Сохранить документ.
 - Просмотреть документ в окне браузера, изменяя размер окна.
7. Выполнить оформление списков:
- Создать новый документ **3_name.htm**, сохранить его в той же рабочей папке жесткого диска.
 - Ввести текст:

```

<HTML>
  <HEAD> <TITLE> фамилия </TITLE>
  </HEAD>
  <BODY>

      Приветствую Вас на моей третьей web-страничке!

  </BODY>
</HTML>

```

- Дополнить текст документа (между тегами <BODY>...</BODY>) следующим текстом:

Я знаю как оформлять:

**Шрифты,
Заголовки,
Абзацы**

- Оформить три последние строки как **список нумерованный**. Для этого использовать следующую конструкцию тегов:

```

<OL>
  <LI> Шрифты, </LI>
  <LI> Заголовки, </LI>
  <LI> Абзацы </LI>
</OL>

```

-
- Поменять оформление списка на **список маркированный**. Использовать теги ,

- Создать «смешанный» список:

Я знаю как оформлять:

1. Шрифты

- Размер
- Цвет
- Гарнитуру
- Индексы

2. Заголовки

- От 1-го до 6-го уровня

3. Абзацы

- Выравнивание
- Разрыв строк внутри абзаца
- С использованием переформатирования.

8. Проверить все страницы на валидацию. Исправить ошибки.

9. Предъявить результаты работы преподавателю.

Практикум 8. Вставка в HTML-документ рисунков. Создание закладок и гиперссылок

Цель работы:

- Научиться выполнять вставку рисунков в HTML-документ
- Научиться создавать закладки и гиперссылки

Упражнение 1

1. Скопировать из Интернета или какой-либо папки в личную папку файлы три графических файла (например, **Arrows1.wmf**, **Arrows2.wmf**, **Arrows3.wmf**).

Убедиться, что созданные ранее документы **1_name.htm**, **2_name.htm** и **3_name.htm** также находятся в вашей папке на жёстком диске.

2. Вставка рисунков в документ.

- Открыть в Блокноте документ **2_name.htm**.
- Вставить рисунок **Arrows1.wmf** в начало документа **2_name.htm**. Для вставки использовать тег

IMG с параметрами **WIDTH** и **HEIGHT** для установки размеров рисунка 50 пикселей по горизонтали и по вертикали.

- Сохранить документ под именем **4_name.htm**.
 - Просмотреть в браузере полученный результат.
 - Ввести в тег рисунка параметр **ALIGN** для выравнивания рисунка по правому краю. Просмотреть результат в браузере.
 - Вставить рисунок **Arrows2.wmf** в конец документа **4_name.htm**, подобрать тип выравнивания рисунка на свое усмотрение. Установить размер рисунка 100 пикселей по горизонтали и по вертикали. С помощью параметра **ALT** создать всплывающую подсказку «Рисунок 2», появляющуюся при наведении курсора мыши на рисунок.
 - Просмотреть в браузере полученный результат.
3. Создание гиперссылок и закладок.
- В документе **3_name.htm** закрепить гиперссылки за следующими элементами списка:
 - За словом **Шрифт** – гиперссылка на документ **1_name.htm**.
 - За словом **Заголовки** – на документ **1_name.htm**.
 - За словом **Абзацы** - на документ **2_name.htm**.

Предъявить результат преподавателю.

Упражнение 2. Создание заголовка и разбиение на абзацы

1. Откройте Блокнот.
2. Введите:

```
<html>
<head>
<title>Моя первая страничка </title>
</head>
<body>
<p>
12 апреля 2003 года <br>
Саратов </p>
<p>
Уважаемый Иван Иванович!</p>
<p>
Прошу Вас рассмотреть возможность участия представителя Вашего <br>
педагогического коллектива в семинаре по проблеме применения <br>
новых информационных технологий в образовании, который будет <br>
проходить 7 мая в нашем центре.</p>
<p>П.П.Петров</p>
</body>
</html>
```
3. Сохраните данный документ в гипертекстовом формате с расширением .html
4. Закройте программу Блокнот и просмотрите созданный файл с расширением .html при помощи браузера.

Упражнение 3. Раздел BODY. Параметры фона

1. Откройте созданный Вами файл с расширением .html. В Проводнике сделать щелчок правой кнопкой мыши на имени файла. Открыть с помощью – Блокнот (Notepad).
2. Изменит цвет фона на всей странице. Для этого надо изменить тег **<body>** следующим образом:
<body bgcolor="FF0066">
3. Сохраните и закройте данный файл.
4. Просмотрите его в браузере.

Внимание!

Параметр **background="back.jpg"** позволяет накладывать на фон документа графическое изображение.

5. Добавьте на фон любое графическое изображение.

Упражнение 4. Создание ссылки на другой документ

1. Создайте еще один файл с расширением .html. Пусть он у нас называется file.html.

```
<html>
<head>
<title>Это информация о семинаре</title>
</head>
<body>
Это информация о семинаре
</body>
</html>
```

2. Создайте ссылку из первого файла со слова семинар на данный файл.
3. Для этого изменим данную строку следующим образом:
...в ****семинаре**** по...
4. Сохраните и закройте данный файл.
5. Просмотрите его в браузере.

Внимание!

Данная ссылка будет работать, если файл, на который происходит ссылка, находится в том же каталоге, что и первый файл.

Если ссылка должна быть на ресурс в Интернете, то тогда строка приобретает вид: ...в <http://www.sgu.ru> семинаре по...

Правила составления гиперссылок на различные Интернет-сервисы

| Название информационного сервиса | Пример гиперссылки |
|----------------------------------|---|
| World Wide Web | http://www.site.ru/ |
| E-mail | mailto:mail@site.ru |
| FTP | ftp://ftp.site.ru |

6. Создайте ссылку на странице на интернет сайт.

Упражнение 5. Создание внутренней гиперссылки

Если HTML-документ слишком большого размера и нет возможности разбить его на несколько отдельных страниц, то можно прибегнуть к помощи внутренних гиперссылок в пределах одной страницы. Структура внутренней гиперссылки имеет две части – сама ссылка и ее именной идентификатор по месту перемещения.

1. Добавьте в документ произвольную информацию в несколько абзацев.
2. Создайте внутреннюю гиперссылку с конца документа на его начало. Для этого в начале страницы создадим именной идентификатор.

```
<body>
<a name="#begin"></a>...
```

3. Создайте в конце страницы ссылку возврата на начало страницы:

```
...<p><a href="#begin">В начало страницы</a></p>
</body>
</html>
```

4. Сохраните и откройте данный файл.
5. Просмотрите его в браузере.

Упражнение 6. Вставка иллюстраций

1. Вставим в наш файл после заголовка иллюстрацию, добавив следующий тег:

```
.
```

В этом теге должно быть точно указано месторасположение и имя файла с рисунком.

Файл с рисунком предварительно расположите рядом с файлом проекта.

2. Сохраните и закройте данный файл.
3. Просмотрите его в браузере.
4. Добавьте на страницу несколько картинок используя описанные ниже параметры.

Параметры тега IMG

| Значение параметра | Действие параметра |
|--|--|
| src | Задаёт путь (относительный или абсолютный) к рисунку |
| border | Предназначен для отображения вокруг рисунка рамки определенной толщины. Толщина указывается в пикселях: border="5" |
| width height | Определение размеров графического изображения (ширина и высота в пикселях) |
| align left right top texttop middle bottom | <ul style="list-style-type: none"> • Определение типа выравнивания рисунка • Текст обтекает рисунок по правому краю (изображение слева) • Текст обтекает рисунок по левому краю (изображение справа) • Верхний край рисунка выравнивается по самому высокому элементу строки • Верхний край рисунка выравнивается по самому высокому текстовому элементу строки • Выравнивание середины рисунка по базовой линии строки • Выравнивание нижнего края рисунка по базовой линии строки |
| hspace vspace | Определяют размер горизонтального и вертикального отступов от встраиваемого изображения до текста. Формат записи значений – в пикселях. |
| alt | Вывод смысловых подсказок для графических элементов |

Практикум 9. Создание и форматирование таблиц

Цель работы:

- Научиться использовать таблицы для оформления WEB-страниц

1. Откройте «Блокнот». Наберите следующий исходный код:

```
<html>
<head>
  <title>Тег TABLE</title>
</head>
<body>
  <table border="1" width="100%" cellpadding="5">
    <tr>
      <th>Ячейка 1</th>
      <th>Ячейка 2</th>
    </tr>
    <tr>
      <td>Ячейка 3</td>
      <td>Ячейка 4</td>
    </tr>
  </table>
</body>
</html>
```

Сохраните документ под именем **f_tabl_name.htm**. Просмотрите результат. Исправьте таблицу таким образом, что появилась новая первая строка с объединенными ячейками. Текст в строке: «Фамилия». Залейте ячейки разными цветами.

2. Создайте таблицу состоящую из 3-х строк и 4-х столбцов. В каждую ячейку поместите произвольный рисунок. Сохраните документ под именем **s_tabl_name.htm**.

3. Откройте документ **s_tabl_name.htm** в «Блокноте». Наберите код следующих таблиц.

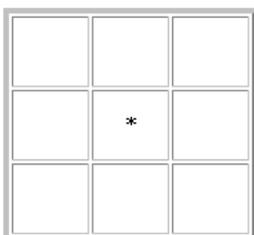
```
<table border="5" bordercolor="#FF0000">
  <tr>
    <td>Первая ячейка</td>
    <td>Вторая ячейка</td>
  </tr>
</table>
```

```
<table border="1" cellspacing="10">
  <tr>
    <td>Первая ячейка</td><td>Вторая ячейка</td><td>Третья ячейка</td>
  </tr>
  <tr>
    <td>Четвертая ячейка</td><td>Пятая ячейка</td><td>Шестая ячейка</td>
  </tr>
</table>
```

```
<table border="1" cellpadding="10">
  <tr>
    <td>Первая ячейка</td><td>Вторая ячейка</td><td rowspan="2">Третья и
    шестая ячейки</td>
  </tr>
  <tr>
    <td>Четвертая ячейка</td><td>Пятая ячейка</td>
  </tr>
</table>
```

Опишите особенности этих таблиц. Добавьте к таблицам произвольные заголовки.

4. Создать таблицу по приведенному образцу, сохранить документ под именем **tabl_name.htm**. Сверху над таблицей разместить заголовок **Таблица №1**



| | | |
|--|---|--|
| | | |
| | * | |
| | | |

При отображении таблицы в браузере должны удовлетворяться следующие условия:

- в центральной ячейке поместить символ * (звездочка), остальные ячейки должны быть пустыми.

Примечание. Для отображения пустых ячеек в них нужно поместить символьный

примитив пробела ` `;

5. В этом же документе создать копию таблицы №1, ввести заголовок **Таблица №2** и модифицировать ее:

- В центральной ячейке разместить рисунок **Arrows3.wmf**
- «Раскрасить» все остальные ячейки в различные цвета.

6. Создать еще одну копию таблицы – **Таблица №3** и отредактировать теги таблицы так, чтобы она соответствовала приведенному ниже образцу.



Примечание. Для объединения ячеек в тегах `<TD>` необходимо использовать параметры `colspan=` и `rowspan=`

7. Создать новый HTML-документ - **rasp_name.htm** с расписанием занятий.

- Документ должен начинаться заголовком

**Расписание занятий гр. NNN на осенний семестр 2015 г (составил
Фамилия Имя Отчество).**

- Первая строка таблицы должна быть оформлена как заголовки полей (с использованием тегов `<TH>`).
- Таблица по ширине должна занимать полный размер окна. Ширину отдельных столбцов задать в относительных единицах (в %), с тем, чтобы при изменении ширины окна пропорции таблицы сохранялись.

| День недели | Время | Предмет | Преподаватель | Аудитория |
|-------------|-------------|------------------|--------------------|-----------|
| Понедельник | 8:30-10:05 | Математика (лек) | доц. Иванов А.А. | 320 |
| | 10:15-11:50 | Математика (пр) | преп. Петрова И.А. | 302 |
| | 12:30-14:05 | Физика (лаб) | доц. Сидоров О.И. | 307 |
| Вторник | 8:30-10:05 | История (лек) | проф. Громова О.А. | 310 |
| | 10:15-11:50 | История (сем) | преп. Попов М.А. | 302 |
| | 12:30-14:05 | Физика (лаб) | доц. Сидоров О.И. | 307 |
| ... | ... | ... | ... | ... |

- Просмотреть созданный документ в браузере при различных размерах окна и различных настройках размера шрифта.

8. Сохранить файл с расписанием под именем **rasp_menu_name.htm** и модифицировать его.

9. После заголовка создать таблицу, состоящую из одной строки меню с названиями дней недели.

Расписание

| | | | | | |
|------------------------|-------------------------|-----------------------|-------------------------|-----------------------|-----------------------|
| Понед. | Вторник | Среда | Четверг | Пятн. | Субб. |
|------------------------|-------------------------|-----------------------|-------------------------|-----------------------|-----------------------|

10. В таблице с расписанием установить закладки на названия дней недели.

11. В таблице меню создать гиперссылки на соответствующие дни недели.

12. Выполнить цветовое оформление каждой ячейки меню.

13. Проверить правильность выполнения переходов по гиперссылкам.

14. Предъявить результат преподавателю.

Практикум 10. Фреймы

Окно просмотра браузера можно разбить на несколько прямоугольных областей, называемых фреймами. Области соприкасаются друг с другом и в каждую из областей можно загрузить отдельный HTML-документ, и работать с ним независимо от документов, загруженных в другие области окна или фрейма. Между фреймами можно организовать взаимодействие, например, выбор ссылки в одном из фреймов позволит изменить содержимое других фреймов. Фреймы часто используются в случаях, когда возникает необходимость загрузить документ в одну из областей при работе в другой области, или когда следует отобразить информацию, которая должна постоянно находиться на экране.

Задание 1. Простая фреймовая структура

Создадим документ, который разбивает область экрана на две части. Левая часть содержит оглавление разделов документа, который располагается в правой части. При выборе пункта оглавления в левой части появляется соответствующий раздел документа в правой части. Разобьем область экрана на два фрейма. Левый фрейм занимает 25% ширины всего окна и будет содержать оглавление разделов документа, который загрузим в правый фрейм. Пусть имя файла, содержащего оглавление - *contents.htm*, а имя документа - *ch.htm*. Фреймовая структура задает способ организации экрана и определяет, какие документы должны быть первоначально загружены во фреймы. Создать

описанную фреймовую структуру можно, если использовать документ, содержащий HTML-код, представленный в листинге:

```
<HTML>
<HEAD>
<TITLE>Простая фреймовая структура</TITLE>
</HEAD>
<frameset cols="25%,75%">
<frame src=contents.htm name=left>
<frame src= ch.htm name=right>
</frameset>
</html>
```

Параметр *cols* тега `<frameset>` имеет вид *cols*="список значений". В списке через запятую перечисляются значения, которые определяют размеры фреймов. Список должен содержать не менее двух значений. Значения могут задаваться в процентах, в пикселах, в относительных единицах.

Элемент <FRAMESET>

Описание: определяет вывод главного пользовательского окна в виде набора прямоугольников (фреймов).

Начальный тег: **необходим.**

Конечный тег: **необходим.**

Определения атрибутов:

rows = "число" – определяет вывод горизонтальных фреймов (разделённый запятыми список пикселей, процентов и относительных размеров). Если атрибут не установлен, каждый столбец занимает всю высоту страницы;

cols = "число" – определяет вывод вертикальных фреймов (разделённый запятыми список пикселей, процентов и относительных размеров). Если не установлен атрибут, каждый ряд занимает всю ширину страницы.

Пример:

| | |
|---|--|
| <code><FRAMESET rows="50%, 50%"></code> ...остальная часть определения... <code></FRAMESET></code> | Экран делится надвое по вертикали (т.е. создаются верхняя и нижняя половины). |
| <code><FRAMESET cols="1*,250,3*"></code> ...остальная часть определения... <code></FRAMESET></code> | Создано три столбца: второй имеет фиксированную ширину 250 пикселей, первый занимает 25% оставшегося пространства, а третий - 75%. |

Тег `<frame>` определяет один фрейм. Он должен располагаться внутри парного тега `<frameset>` и `</frameset>`. Число тегов `<frame>` должно совпадать с количеством тегов, определенных при описании фреймовой структуры. В рассматриваемом примере в теге `<frameset cols="25%,75%">` определено два фрейма, поэтому в дальнейшем следует описание каждого из фреймов с помощью тега `<frame>`.

Значение параметра *src* тега `<frame>` определяет адрес документа, который первоначально загружается во фрейм. В рассматриваемом случае в левый фрейм загружается документ с именем *contents.htm*, а в правый фрейм - документ с именем *ch.htm*. В теге параметр *name* определяет имя фрейма, необходимое для указания, в какой фрейм загрузить документ. Если имя фрейма не задавать, то будет создан фрейм без имени, но сослаться на него из других фреймов будет нельзя.

Фреймовая структура с загружаемыми документами

Создадим документ, левая часть которого представляет оглавление, а в правую часть загружаются документы выбранного пункта оглавления. Документы, соответствующие пунктам оглавления, хранятся в разных файлах.

При решении задачи экран по-прежнему разбивается на два фрейма. Левый фрейм занимает 25% ширины всего окна и будет содержать оглавление документов, которые могут быть просмотрены пользователем при выборе соответствующего пункта. Правый фрейм занимает большую часть окна просмотра и предназначен для отображения самих документов. При первоначальной загрузке оба фрейма делят окно просмотра по вертикали в соотношении 25% и 75%. Данное соотношение может меняться при просмотре.

Каждый из фреймов имеет свою полосу прокрутки, обеспечивающую просмотр всего документа. При выборе ссылки в левом фрейме соответствующий документ будет загружен в правый фрейм. Такая структура позволяет одновременно видеть на экране и оглавление документов, и сами документы.

Пусть оглавление документа содержит шесть пунктов и располагается в файле с именем *contents.htm*. Требуется, чтобы файл, содержащий оглавление, загружался в левый фрейм. Файлы с именами *ch1.htm*, *ch2.htm*, ..., *ch6.htm* содержат документы, соответствующие пунктам оглавления.

В правый фрейм первоначально загружается файл с именем *ch.htm*. Если сразу неизвестно, какой файл загружать во фрейм, то можно использовать файл, содержащий HTML-код :

```
<TITLE>ПУСТОЙ ДОКУМЕНТ</TITLE>
<HEAD>
<HEAD>
<BODY>
<BODY>
<HTML>
```

В левый фрейм помещается **Оглавление**, которое содержит ссылки на документы, расположенные в различных файлах. Оглавление может быть сформировано так, как указано в листинге *contents.htm*:

```
<HTML>
<HEAD>
<TITLE>Оглавление</TITLE>
</HEAD>
<BODY >
<base target=right> ← указывает имя окна(фрейма) куда будет загружаться документ
<h3>Оглавление</h3>
<OL>
<LI><A href="ch1.htm">Информация</A>
<LI><A href="ch2.htm">Двоичная система счисления </A>
<LI><A href="ch3.htm">Моделирование</A>
<LI><A href="ch4.htm">Алгоритм</A>
<LI><A href="ch5.htm">Архитектура ПК </A>
<LI><A href="ch6.htm">Программное обеспечение </A>
</OL>
</body>
</HTML>
```

Другой вариант:

```
<HTML>
<HEAD>
<TITLE>Оглавление</TITLE>
</HEAD>
<BODY > <h3>Оглавление</h3>
<OL>
<LI><A href="ch1.htm" target=right >Информация</A>
<LI><A href="ch2.htm" target=right >Двоичная система счисления </A>
<LI><A href="ch3.htm" target=right >Моделирование</A>
<LI><A href="ch4.htm" target=right >Алгоритм</A>
<LI><A href="ch5.htm" target=right >Архитектура ПК </A>
<LI><A href="ch6.htm" target=right >Программное обеспечение </A>
</OL>
</body> </HTML>
```

Дополнительно: Создайте файлы с именами *ch1.htm*, *ch2.htm*, ..., *ch6.htm* содержащие соответствующие оглавлению документы (2-3 абзаца текста с изображением).

1. Откройте **Блокнот**.

2. Создайте документ с фреймовой структурой:

```
<FRAMESET ROWS="20%, 80%">

<FRAME SRC=frame1.html>

<FRAMESET cols="30%, 70%">

<FRAME SRC=frame2.html>

<FRAME SRC= spisok.html name="pr">

</FRAMESET>

</FRAMESET>
```

3. Сохраните файл под именем *index.html* .

4. Откройте **Paint**.

5. В **Paint** создайте рисунок размером 600x100 точек и сохраните его в формате *JPG* под именем *pic*.

6. Создайте *HTML*-документ с именем *frame1.html* . в нём:

- измените фоновый цвет Цвет фона устанавливается в теге `<body>`:
например, `<body bgcolor="#000000">`;
- вставьте рисунок *pic*, сделайте, чтобы он выровнился по центру.

7. Создайте *HTML*-документ с именем *frame2.html* с текстом:

Список

Таблица

Разработчик

8. Надпись "Список" оформите в виде гиперссылки на файл *spisok.html* .

9. Надпись "Таблица" оформите в виде гиперссылки на файл *tabl.html* .

10. Надпись "Разработчик" оформите в виде гиперссылки на файл *razrab.html*

11. Создайте *HTML*-документ с именем *spisok.html* . В документе:

- создайте заголовок «Список», сделайте, чтобы он выровнился по центру;
- создайте нумерованный список, состоящий из 10 произвольных строк;
- сделайте, чтобы документ загружался в правый фрейм.

12. Создайте *HTML*-документ с именем *tabl.html*. В документе создайте:

- заголовок «Таблица»;
- видимую таблицу вида:

| | | |
|--|--|--|
| | | |
| | | |
| | | |
| | | |

- измените цвет заливки первой ячейки;
- в ячейки таблицы введите произвольный текст;
- сделайте, чтобы документ загружался в правый фрейм.

13. Создайте *HTML*-документ с именем *razrab.html*. В документе:

- создайте заголовок «Работу выполнил»;
- заголовок сделайте курсивным;
- напишите свою Ф.И.О., № группы;
- создайте горизонтальную линию. Для линии задайте цвет, длину и толщину. Сделайте, чтобы она выровнилась по центру;
- напишите текущую дату;
- сделайте, чтобы документ загружался в правый фрейм.

Практикум 11. Каскадные таблицы стилей

Цель работы: познакомиться с возможностями и способами создания каскадных таблицей стилей.

Порядок выполнения работы

1. Откройте **Paint**.

2. В **Paint** создайте рисунок размером 100x100 точек и сохраните его в формате *JPG* под именем *bgpic*.

3. В **Paint** создайте второй рисунок размером 200x400 точек и сохраните его в формате *JPG* под именем *risunok*. В **Paint** создайте третий рисунок размером 200x400 точек и сохраните его в формате *JPG* под именем *risunok1*.
4. Откройте **Блокнот**.
5. Напишите следующий код:

```
body {background-color: lightsteelblue;
background-image: url(bgpic.jpg)}
P {font: 14pt "Times Cyr", serif}
P {text-indent: 20}
H1 {color: red}
H2 {color: blue}
```

Сохраните файл с именем *mycss.css*.

6. Создайте *HTML*-документ, в нём создайте:
 - заголовок первого уровня "Краткая информация";
 - абзац, в котором напишите несколько строк о себе;
 - горизонтальную линию;
 - заголовок второго уровня "Учёба";
 - абзац, в котором напишите название факультета, направление подготовки, группу;
 - горизонтальную линию;
 - надпись "Контактная информация", оформите её как гиперссылку на документ *kontakt.html*.

Сохраните файл с именем *index.html*.

7. Подключите к *HTML*-документу созданную таблицу стилей (*mycss.css*).
8. Используя внедрённую таблицу стилей, измените положение и повторяемость фонового рисунка, расстояние между буквами в заголовке второго уровня, интервал между строками в абзаце, выравнивание текста.
9. Создайте в **Блокноте** новый *HTML*-документ, в нём:
 - создайте заголовок первого уровня "Контактная информация";
 - создайте список, содержащий *e-mail*, адрес, телефон;
 - вставьте рисунок *risunok*, сделайте чтобы текст обтекал его с левой стороны;
 - вставьте рисунок *risunok1*, сделайте чтобы текст обтекал его с правой стороны;
 - подключите созданную таблицу стилей;
 - используя внедрённую таблицу стилей измените цвет текста в абзаце.

Сохраните файл в папку с именем *kontakt.html*.

Каскадный подход

Обтекание картинки текстом обычно используется для компактного размещения материала и связывания между собой иллюстраций и текста. Само обтекание создаётся с помощью стилевого свойства **float**, добавляемого к селектору **IMG**. Значение **left** выравнивает изображение по левому краю, **right** — по правому. Обтекание при этом происходит по другим, свободным сторонам. Приведен пример создания класса в каскадной технологии. Использование свойства **float** заставляет текст плотно прилегать к изображению. Поэтому в примере введёно универсальное свойство **margin**, которое добавляет отступы между картинкой и текстом. Это свойство одновременно задаёт отступ сверху, справа, снизу и слева от фотографий.

```
<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8">
<title>Обтекание</title>
<style>
.leftimg {
float:left; /* Выравнивание по левому краю */
margin: 7px 7px 7px 0; /* Отступы вокруг картинки */
}
.rightimg {
float: right; /* Выравнивание по правому краю */
margin: 7px 0 7px 7px; /* Отступы вокруг картинки */
```

```

}
</style>
</head>
<body>
<h2>Доклад лейтенанта Бокатуева</h2>
<p>
Вчера во время проведения разведоперации наша группа подверглась нападению неизвестного
противника в камуфляжной форме Алиенов. В результате эффективной обороны и стремительной
контратаки многочисленная группа боевиков была смята и отброшена. Среди личного состава потерь нет.
Бойцы разведгруппы проявили недюжие навыки владения оружием. Особо отличился в бою взводный
Кудряшев&nbsp;М.А., грамотно использовавший человеческие ресурсы своего взвода. В результате
операции были захвачены элементы веземной культуры, которые переданы аналитической группе.</p>
<h2>Пресс-релиз аналитической группы</h2>
<p>
В наших секретных лабораториях в рамках проект&laquo;Пандора&raquo; разрабатывалось психотропное
оружие. В результате неудачного эксперимента большинство ученых, работавших над прибором,
подверглись воздействию психотропного излучения, и они, находясь в состоянии аффекта, растащили
прототип по деталям. Возможно, наши ученые до сих пор находятся в состоянии аффекта.</p>
</body>
</html>

```

Практикум 12. Введение В JavaScript

Упражнение 1.

```

<HTML>
<HEAD>
<SCRIPT>
alert ("Добро пожаловать !") ;
</SCRIPT>
</HEAD>
</HTML>

```

*Сохраните файл под именем 1.html
Проверьте его работу.*

Подсказка по операторам:

Метод alert() – это один из инструментов языка JavaScript. При его выполнении на экране появляется окно сообщения, в котором отображаются заданный текст, кнопка ОК и пиктограмма с восклицательным знаком. Такие предупреждения используются для привлечения внимания пользователя. Приведенная программа выполняется немедленно после загрузки включающего ее документа.

- **JavaScript –зависит от регистра.** Имена JavaScript и Javascript - разные имена!! Все ключевые слова используют только нижний регистр.
- Требования к именам переменных такие же, как в Паскале.
- Операторы разделяются точкой с запятой, которую можно опустить, если оператор заканчивается символом новой строки (Enter).
- Комментарии:

```

// однострочный комментарий,
/*
..многострочный комментарий
*/

```

Упражнение 2

```

<HTML>
<HEAD>
</HEAD>
<body>
<SCRIPT>
var x=8
if (x>=2 && x<=6) y=0; else y=1;
alert(y)

```

```
</SCRIPT>
</body>
</HTML>
```

Сохраните файл под именем 2.html

В JavaScript существует 3 функции (метода), позволяющие пользователю выводить диалоговые окна:

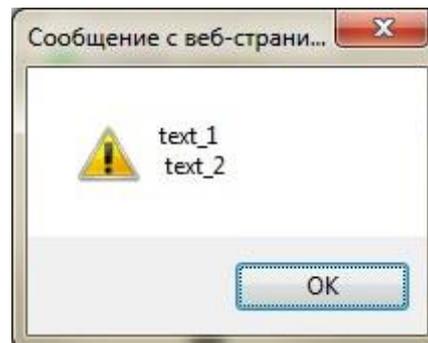
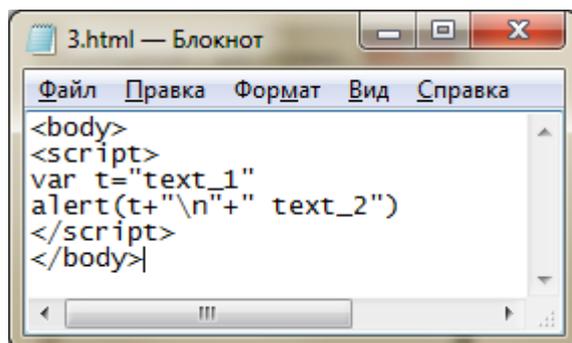
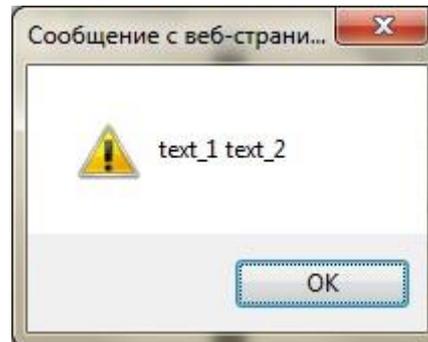
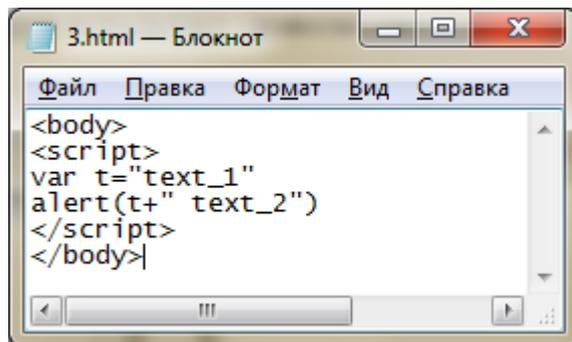
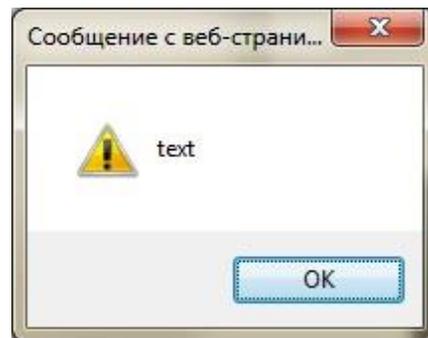
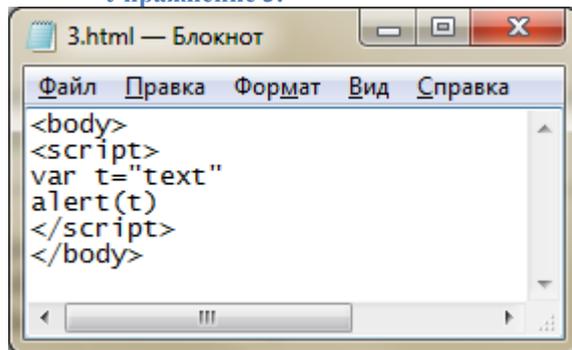
- alert
- confirm
- prompt

alert ("строка")

Метод alert используется для вывода простейшего диалогового окна, содержащего текст сообщения и единственную кнопку "Ok". Программа выводит сообщение и ожидает нажатия кнопки. После нажатия на кнопку, программа начинает выполняться дальше.

Текст сообщения может сцепляться с любой текстовой переменной с помощью знака «+». Чтобы текст выводился в несколько строк используют символы «\n»

Упражнение 3.



Сохраните файл под именем 3.html

Задание:

Создайте гипертекстовый документ, который при помощи скрипта выводит в отдельном окне вашу фамилию и номер группы (на разных строчках)/ Сохраните файл под именем 3-1.html

Confirm (“строка”)

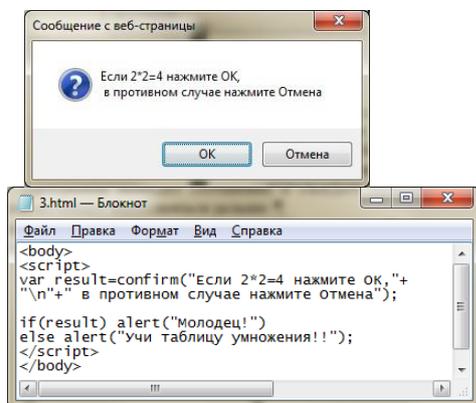
Метод confirm используется в тех случаях, когда пользователь должен сделать выбор.

Метод confirm позволяет пользователю вывести диалоговое окно, содержащее текст вопроса и кнопки "ОК" и "Отмена".

Функция confirm возвращает логическое значение в зависимости от нажатой пользователем кнопки:

- "ОК" соответствует значению true,
- "Отмена" - значению false.

Как правило, результат работы функции присваивают логической переменной, для дальнейшего анализа, как это показано в примере.

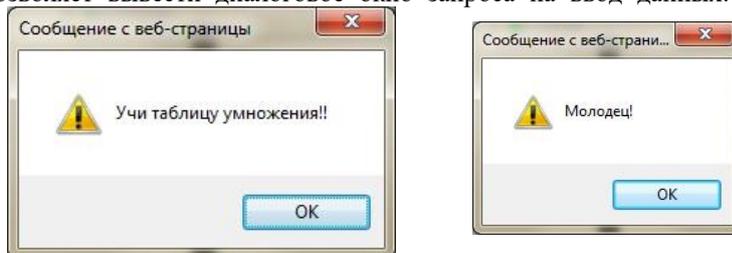


prompt (“строка1”, “строка2”)

Метод prompt используется в тех случаях, когда пользователю нужно ввести значение в переменную.

В окно выводится сообщение «строка1», в поле ввода помещается умалчиваемое значение «строка2».

Этот метод позволяет вывести диалоговое окно запроса на ввод данных. Результат работы функции

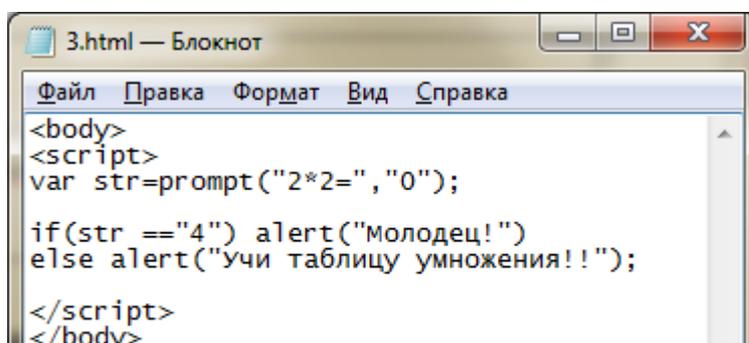


присваивают переменной строкового типа.

Если введенные данные нужно использовать в арифметических выражениях, необходимо выполнить преобразование введенной строки к числовому типу. Это можно сделать при помощи следующих функций:

- parseInt("строка") - преобразует строку в целое число;
- parseFloat("строка") - преобразует строку в число с плавающей точкой.

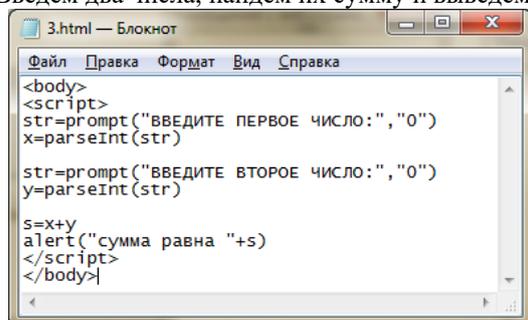
Упражнение 4



Сохраните файл под именем 4.html

Упражнение 5.

Введем два числа, найдем их сумму и выведем ее на экран.

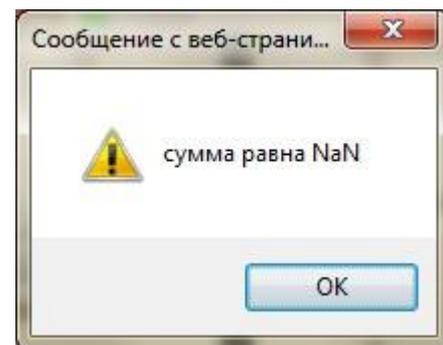
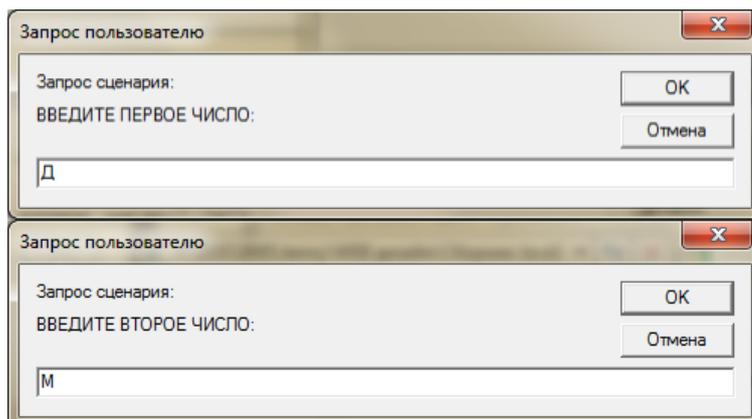


```
<body>
<script>
str=prompt("ВВЕДИТЕ ПЕРВОЕ ЧИСЛО:", "0")
x=parseInt(str)

str=prompt("ВВЕДИТЕ ВТОРОЕ ЧИСЛО:", "0")
y=parseInt(str)

s=x+y
alert("сумма равна "+s)
</script>
</body>
```

В данном скрипте переменные не описываются, что допускается. Переменная str будет иметь строковый тип, так как результат функции prompt должен быть строкового типа. Функция parseInt преобразовывает переменную str строкового типа в переменную x числового типа. Переменная s в операторе присваивания имеет числовой тип, так как переменные x и y имеют числовой тип. Переменная s в функции alert будет преобразована в строковый тип, так как параметр этой функции должен быть строкой.



Буквы не могут быть преобразованы в числа. Поэтому переменные x, y и s не будут иметь значений. Когда переменная не имеет значения, то выводится NaN.

Сохраните файл под именем 5.html

Задание:

1. Создайте скрипт вычисления площади треугольника (Запрашиваете основание и высоту, площадь равна половине произведения основания на высоту: $s=a*h*0.5$);

Сохраните файл под именем 5-1.html

2. Создайте мини-тест из 3-х вопросов. Каждый ответ увеличивает (или не увеличивает) значение переменной. В конце теста выводится результат в зависимости от значения переменной. Сохраните файл под именем 5-2.html

JavaScript это объектно-ориентированный язык. Основной единицей в объектно-ориентированном языке является объект, который объединяет в себе данные (свойства) и средства обработки этих данных (методы). Если говорить образно, то объекты – это «существительные», свойства объекта – это «прилагательные», а методы объекта – это «глаголы». Значения свойств объектов можно изменять.

Про JavaScript говорят, что в нем все объект. А именно: объектами являются окно, в котором открывается документ, сам документ, все элементы документа и даже свойства этих элементов. Есть также специальные встроенные объекты. Для упорядочивания огромного количества объектов создатели

браузеров придумали объектную модель документа. Эта модель является структурой организации объектов на странице.

Объект document соответствует всему HTML-документу.

Изучим один метод этого объекта, позволяющий динамически формировать документ. Метод document.write("строка html-кода") - выводит строку в окно документа.

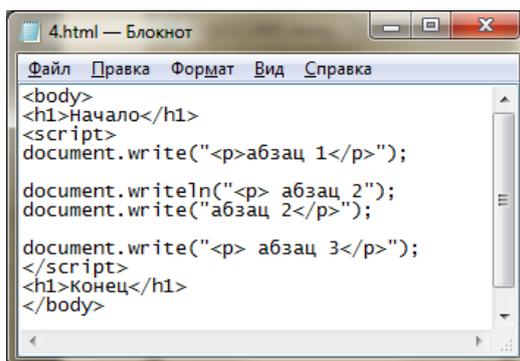
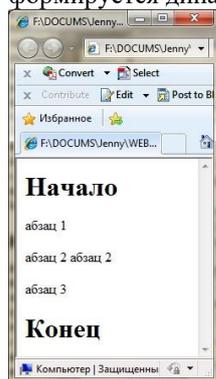
Метод document.writeln("строка html-кода") - выводит строку в окно документа, в конце выводится символ "пробел".

Метод, применяемый к объекту, пишется после имени объекта через точку. Содержимое строки должно быть в кавычках или это может быть объединение (сумма) нескольких строк или строковых переменных.

Строка должна содержать элементы разметки страницы (теги и их содержимое). Метод исполняется в процессе загрузки документа.

Упражнение 6.

В данном примере заголовки (Начало и Конец) находятся в документе, а текст абзацев формируется динамически и затем выводится в документ.



Сохраните файл под именем 6.html

```
<HTML>
<HEAD>
</HEAD>
<body>
<SCRIPT>
var str=prompt("2+2=", "0");
document.write("<p><b>Результат</b></p>");
if (str=="4")
document.write("<p>Отлично</p>");
else
document.write("<p>Плохо</p>");
alert(str);
</SCRIPT>
</body>
</HTML>
```

Арифметические операции

```
var a = 5;
var b = 12;
document.write(a + b);
```

Пример:

```
<HTML>
<HEAD>
</HEAD>
<body>
<SCRIPT>
var str=prompt("Введите произвольное число для возведения в квадрат и куб", "0");
document.write("<p><b>Результат</b></p>");
y=parseInt(str)
document.write(y * y);
```

```
document.write("<br>");
document.write(y * y * y);
</SCRIPT>
</body>
</HTML>
```

Задание:

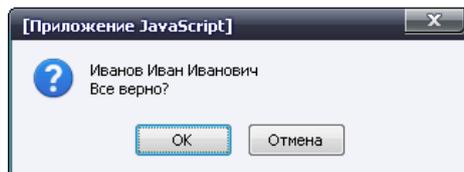
Пользователю предлагается выбрать картинку по ее номеру. В зависимости от выбора в окне браузера появляется изображение. Сохраните файл под именем 6-1.html

Практикум

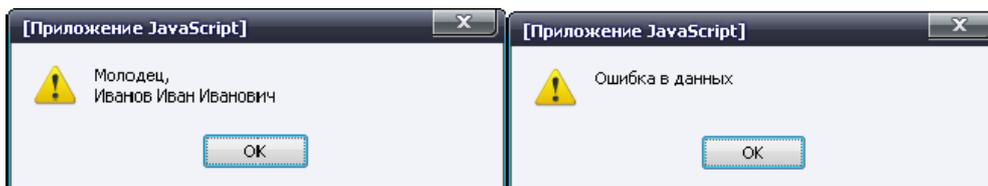
Задание 1

Введите свою фамилию, имя и отчество.

Запросите подтверждение.



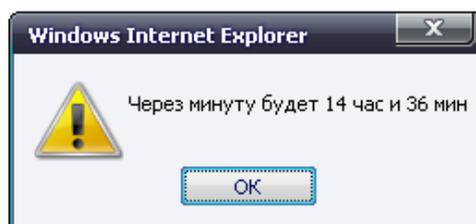
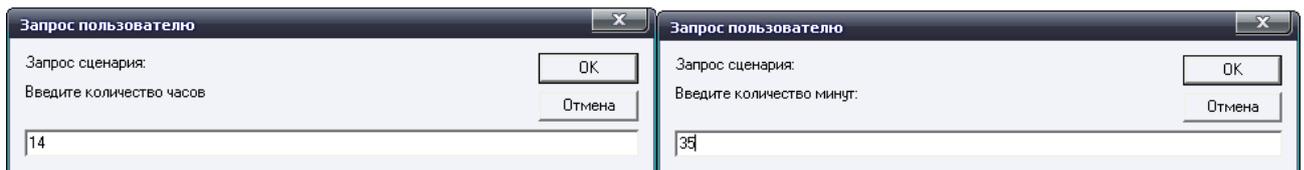
Если все верно, то вывести приветствие, если нет, вывести сообщение об ошибке.



Задание 2

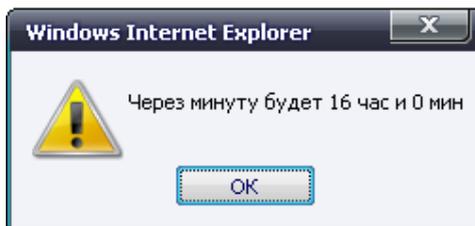
Введите время в часах и минутах. Определите время, которое будет через минуту. Используйте вложенные операторы IF. Возможны три случая:

1.



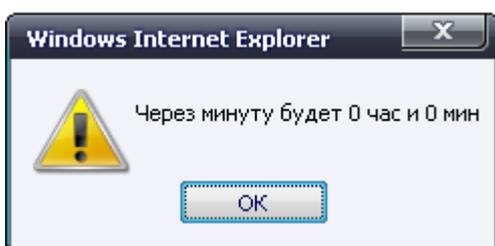
2.

| | |
|---|---|
| Запрос пользователю Запрос сценария: <input type="button" value="OK"/> Введите количество часов: <input type="button" value="Отмена"/> <input type="text" value="15"/> | Запрос пользователю Запрос сценария: <input type="button" value="OK"/> Введите количество минут: <input type="button" value="Отмена"/> <input type="text" value="59"/> |
|---|---|



3.

| | |
|---|---|
| Запрос пользователю Запрос сценария: <input type="button" value="OK"/> Введите количество часов: <input type="button" value="Отмена"/> <input type="text" value="23"/> | Запрос пользователю Запрос сценария: <input type="button" value="OK"/> Введите количество минут: <input type="button" value="Отмена"/> <input type="text" value="59"/> |
|---|---|



Задание 3. Разложите целое число введенное пользователем на простые множители.

Практикум 13. Вычисления в веб-документах

Часть 1-я «Формы»

Пример 1 - простая форма для ввода:

```

<HTML>
<HEAD>
<TITLE>Простая форма</TITLE>
</HEAD>
<BODY>
<FORM name="form1">
Улица: <INPUT NAME= "street" TYPE="TEXT"> <BR>
Город: <INPUT NAME= "city" TYPE="TEXT" SIZE= "20" MAXLENGTH= "20"> <BR>
Индекс: <INPUT NAME= "zip" TYPE="TEXT" SIZE= "5" MAXLENGTH= "5" VALUE= "99999"> <BR>
</FORM></BODY></HTML>

```

BUTTON - произвольная кнопка, ее действия назначаются вами, т.е. сама она делать ничего не умеет. Ее параметры:

- type="button" - тип произвольной кнопки,
- name - имя кнопки,
- value - надпись на кнопке.
- onclick - указывает, что делать при щелчке по кнопке.

Вообще, у этого типа кнопок есть и другие события (например, двойной щелчок), но здесь мы не будем их рассматривать.

RESET - кнопка, позволяющая восстановить все значения по умолчанию в форме. Ее параметры:

- type="reset" - тип кнопки очищения,

- name - имя кнопки,
- value - надпись на кнопке.

Пример 2 - простая форма для ввода с кнопкой:

```
<HTML>
<HEAD>
<TITLE>Простая форма</TITLE>
</HEAD>
<BODY>
<FORM name="form1">
Улица: <INPUT NAME= "street" TYPE="TEXT"> <BR>
Город: <INPUT NAME= "city" TYPE="TEXT" SIZE= "20" MAXLENGTH= "20"> <BR>
Индекс: <INPUT NAME= "zip" TYPE="TEXT" SIZE= "5" MAXLENGTH= "5" VALUE= "99999"> <BR>
<INPUT type="button" value=Отправить><BR>
<INPUT type="reset"><BR>
</FORM></BODY></HTML>
```

Существует возможность получения значение поля в форме, для использования внутри документа. Например, чтобы получить значение текстового поля street, входящего в состав нашей формы, можно воспользоваться следующей строкой кода:

```
Document.form1.street.value
```

Пример 3 Вычисление налога

Создадим документ, в котором после ввода некоторой суммы вычисляется налог, составляющий 13% от введенной суммы. На странице первое текстовое поле используется для ввода дохода, по щелчку по кнопке “Вычислить” в поле налог помещается вычисленная сумма. Вид документа приведен на рис.

Диалог с пользователем: вычисление налога



Интерактивные документы создаются с помощью форм. Форма на рис. содержит четыре элемента: два текстовых поля (для ввода значения и для результата) и две кнопки. При щелчке мышью по кнопке Вычислить мы хотим получить значение в поле налог.

Действие пользователя (например, щелчок кнопкой мыши) вызывает событие, которое производится в основном с элементами форм HTML. Обычно перехват и обработка события задается в параметрах элементов форм. Имя параметра обработки события начинается с приставки on, за которой следует имя самого события. Например, параметр обработки события Click будет выглядеть onClick. Значением параметра обработки события могут быть операторы языка JavaScript..

Пример 3

```
<HTML>
<HEAD>
<TITLE>Вычисление налога без использования сценария</TITLE>
</HEAD>
<BODY>
<h4>Вычисление размера налога</h4>
<FORM name="form1">
<p>Введите размер дохода и нажмите кнопку «Вычислить»</p>
Доход: <INPUT type="text" name="num" size=10><br>
```

```
<INPUT type="button" value=Вычислить onClick="document.form1.res.value =0.13*
document.form1.num.value" ><br>
Налог: <INPUT type="text" name="res" size=10><br>
<INPUT type="reset">
</FORM></BODY></HTML>
```

Задание. Добавьте поле стаж и премия. Вычисления премии: если стаж больше 3-х лет то премия равна 50% от дохода, иначе 10%.

Сохраните документ под именем f1.html

Задания:

1. Создайте документ, в котором после ввода трехзначного числа выводятся в информация о количестве сотен, десятков и единиц.
2. Создайте документ, в котором после ввода стороны квадрата вычисляется периметр и площадь.
3. Создайте документ, решающий квадратное уравнение.

Сохраните документы под именем z1.html, z2.html, z3.html.

Подсказка по командам:

Возведение в степень - `Math.pow(число, степень)`

Корень - `Math.sqrt(число)`

Прим.:

`Math.pow(2, 8) // 256`

`Math.sqrt(9) // 3`

Часть 2-я «Функции»

Пример 4

Стоимость заказа

Создадим документ, в котором располагается форма для ввода стоимости единицы наименования, при нажатии на кнопку вычисляется сумма всего заказа.

Функция `sumb()` используется без параметров. Введенные в форме значения сохраняются в локальных переменных `a1, a2, a3, a4`. Локальная переменная `s` служит для определения суммы, вычисленное значение записывается в соответствующее поле формы. Стандартная функция `Number` преобразует строковое значение в число. Выдаваемое функцией значение определяется оператором `return`, в рассматриваемом случае это значение переменной `s`.

Пример 4

```
<HTML>
<HEAD>
<TITLE>Возвращаемое значение функции</TITLE>
<script>
function sum()
{ var a1= Number(document.form1.num1.value)
  var a2= Number(document.form1.num2.value)
  var a3= Number(document.form1.num3.value)
  var a4= Number(document.form1.num4.value)
  var s= a1+a2+a3+a4
  return s
}
</script>
</HEAD>
<BODY>
<h4>Сумма заказа</h4>
<FORM name="form1">
<p>Введите цены и нажмите на кнопку «Вычислить»</p>
Наименование 1: <INPUT type="text" name="num1" size=4><br>
Наименование 2: <INPUT type="text" name="num2" size=4><br>
Наименование 3: <INPUT type="text" name="num3" size=4><br>
Наименование 4: <INPUT type="text" name="num4" size=4><br>
```

```

<INPUT type="button" value=Вычислить onClick="document.form1.res.value =sum()" ><br>
Стоимость заказа: <INPUT type="text" name="res" size=4><br>
<INPUT type="reset">
</FORM></BODY></HTML>

```

В результате выполнения вызова функции sum() вычисляется значение, которое записывается в текстовое поле результата с помощью оператора присваивания

```
document.form1.res.value= sum()
```

Сам оператор присваивания является значением параметра обработки события. Левая часть оператора присваивания соответствует свойству текстового поля, которое требуется изменить в результате выполнения сценария. Вызов функции выступает в роли выражения, значение которого используется в дальнейших вычислениях, в правой части оператора присваивания.

Пример 5

Вызов функции в качестве оператора

Рассмотрим случай, когда вызов функции выступает в роли оператора. Будем решать задачу определения суммы баллов по результатам пяти экзаменов. Запись вычисленного результата в соответствующее поле формы будет осуществляться в самой функции. Доступ к полю, в которое требуется поместить вычисленное значение, осуществляется следующим образом

```
d.form1.res.value= a1+a2+a3+a4
```

```

<HTML>
<HEAD>
<TITLE>Формирование результата в поле формы в теле функции</TITLE>
<script>
function sumball1()
{ var d= document
  var a1= Number(d.form1.num1.value)
  var a2= Number(d.form1.num2.value)
  var a3= Number(d.form1.num3.value)
  var a4= Number(d.form1.num4.value)
  d.form1.res.value= a1+a2+a3+a4
}
</script>
</HEAD>
<BODY>
<h4>Сумма баллов по результатам сессии</h4>
<FORM name="form1">
<p>Введите оценки и нажмите на кнопку <b>Вычислить</b></p>
Оценка 1: <INPUT type="text" name="num1" size=2><br>
Оценка 2: <INPUT type="text" name="num2" size=2><br>
Оценка 3: <INPUT type="text" name="num3" size=2><br>
Оценка 4: <INPUT type="text" name="num4" size=2><br>
<INPUT type="button" value=Вычислить onClick="sumball1()" ><br>
Сумма баллов: <INPUT type="text" name="res" size=4><br>
<INPUT type="reset">
</FORM></BODY></HTML>

```

Сценарий начинает работу после щелчка по кнопке с именем Вычислить, что соответствует конструкции onClick="sumball1()". Вызов функции выступает в роли оператора, в отличие от предыдущего примера, в котором вызов функции выступал в роли выражения.

Функция выступает в роли оператора и в том случае, когда при решении задачи требуется вычислить сразу несколько значений, как в следующем примере.

Задания (Вариант получите у преподавателя)

1. В документе задаются пять оценок, полученных в сессию. Напишите сценарий определения среднего балла по результатам сдачи сессии.
2. В документе задается стоимость единицы товара и количество изделий. Напишите сценарий, определяющий стоимость всей партии товаров.
3. В документе задается тариф, определяющий стоимость одной минуты переговоров, и количество минут. Напишите сценарий, определяющий, какая сумма должна быть выплачена за весь разговор.

4. В документе задается сумма гонорара за месяц. Напишите сценарий, определяющий сумму, выдаваемую на руки после удержания налога в 13%.
5. В документе задается стоимость одного часа занятий и количество часов, отводимых на курс. Напишите сценарий, определяющий стоимость всего курса.
6. В документе вводится доход сотрудника по месяцам в течение трех месяцев. Требуется определить сумму дохода за три месяца, налог с полученной суммы и сумму, выдаваемую на руки.
7. В документе вводится длина сторона квадрата. Требуется определить периметр и площадь квадрата.
8. В документе вводится длины двух сторон прямоугольника. Требуется написать сценарий, определяющий периметр и площадь прямоугольника.

Практикум 14.

Создайте сайт в одном из конструкторов сайтов.

5. Образовательные технологии, применяемые при освоении дисциплины

Предусматривается широкое использование в учебном процессе активных и интерактивных форм: организация дискуссий и обсуждений спорных вопросов, использование метода мозгового штурма и метода проектов, а также технология электронного портфолио.

При обучении лиц с ограниченными возможностями и инвалидов используются подходы, способствующие созданию безбарьерной образовательной среды: технологии дифференциации и индивидуализации обучения, сопровождение тьюторами в образовательном пространстве; увеличивается время на самостоятельное освоение материала.

Удельный вес занятий, проводимых в интерактивных формах, определяется главной целью программы, особенностью контингента обучающихся и содержанием конкретных дисциплин, и в целом в учебном процессе они должны составлять не менее 50% аудиторных занятий.

В рамках практической подготовки по данной дисциплине используются проектные задания, выполнение которых направлено на формирование таких профессиональных действий как способность использовать математический аппарат, методы программирования и современные информационно-коммуникационные технологии для решения практических задач получения, хранения, обработки и передачи информации

Примеры проектных заданий приведены в фондах оценочных средств.

6. Учебно-методическое обеспечение самостоятельной работы студентов. Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины.

Фонд оценочных средств дисциплины включает в себя: тестовые задания, задания контрольных работ, контрольные вопросы, задания для самостоятельных работ, задания для написания рефератов.

В рамках самостоятельной работы студенты изучают дополнительную литературу, интернет ресурсы по тематике курса.

Для реализации принципа индивидуального подхода на занятиях студентам предлагаются темы индивидуальных докладов и рефератов,

написание которых практикуется в учебном процессе в целях приобретения студентом необходимой профессиональной подготовки, развития навыков самостоятельного научного поиска; изучения литературы по выбранной теме; анализа различных источников и точек зрения, обобщения материала, выделения главного, формулирования выводов и т.п. С помощью рефератов и докладов студент глубже постигает наиболее сложные проблемы курса; учится лаконично излагать свои мысли, правильно оформлять работу, докладывать результаты своего труда. Содержание реферата и доклада должно соответствовать теме и ее плану. Процесс написания реферата и доклада включает в себя: 1) выбор темы; 2). подбор литературы и иных источников, их изучение; 3) составление плана; 4) введение (краткое введение, в котором обосновывается актуальность темы); 5) основной текст; 6) заключение; 7) список использованной литературы.

Студенты выполняют задания самостоятельно, пользуясь интернет-ресурсами, дополнительной литературой.

Задания для самостоятельной работы

Методические указания.

Задания студенты выполняют во внеурочное время, самостоятельно. Результаты предоставляются преподавателю в электронном виде.

Задания для самостоятельной работы

Задание 1.

Рефераты по темам:

1. Локальные и глобальные сети.
2. Информационные сети.
3. Wi-Fi – технология беспроводной связи.
4. Высокоскоростные сети.
5. Безопасность сетей на базе TCP/IP.
6. Обзор современных решений LAN на основе Wi-Fi.
7. Развитие технологий соединения компьютеров в локальные сети.
8. Программное обеспечение локальных сетей.

Задание 2.

Презентации по темам:

1. История формирования всемирной сети Internet.
2. Защита информации в глобальной сети.

Задание 3.

Реферат по теме:

1. Электронная почта.

Задание 4.

Рефераты по темам:

1. WWW технологии.
2. Интернет: административное устройство и структура глобальной сети.
3. Анализ поисковых систем Интернет.
4. Популярные услуги Интернет.
5. Глобальные гипертекстовые структуры: WWW.
6. Новые виды сервиса Internet – ICQ, IP-телефония, видеоконференция.
7. Телеконференции UseNet.
8. Образовательные ресурсы сети Internet.
9. Досуговые ресурсы сети Internet.
10. Компьютер как средство общения.
11. Интерактивные сервисы Интернет.

Задание 5.

Рефераты по темам:

1. Создание WEB-страниц.
2. Интерактивные элементы Web-страниц и скрипты.
3. Графические форматы при оформлении Web-страниц.
4. Средства разработки Web-страниц.
5. Элементы Web-дизайна.
6. Возможности организации мультимедиа-видеоконференций.

Задание 6. Определить наличие драйвера в ПК. Изучить способы установки драйверов. Установить драйвер по коду устройства из сети Интернет.

Задание 7. Подготовить 2 ПК к построению сети. Подготовить пачт-корд с разъемами RJ-45. Соединить два ПК. Настроить Сетевое подключение. Проверить передачу данных в сети.

Задание 8. Выполнить соединение трех ПК. Настроить и проверить сеть.

Задание 9. Соединить компьютеры посредством cross-over кабеля в сеть.

Задание 10. Подключить 4 ПК к коммутатору. Настроить и проверить сеть.

Задание 11. Обеспечить доступ к сети Интернет со всех рабочих станций.

Задание 12. Выполнение операций копирования и печати через локальную сеть.

Задание 13. Найти, как называется самое большое пресноводное озеро в мире.

Задание 14. Найти биографию министра образования Российской Федерации Филиппова В.М. с помощью поисковой системы Google.ru.

Задание 15. Найти сайты физико-математических школ с помощью тематического поискового каталога.

Задание 16. Найти Положение Министерства образования Российской Федерации о порядке аттестации педагогических и руководящих работников муниципальных и образовательных учреждений.

Задание 17. Подготовить иллюстрации к докладу о методике проведения уроков в школе.

Задание 18. Найти и сохранить на локальном диске один из рассказов Ивана Безродного.

Задание 19. Познакомиться с основными возможностями и элементами интерфейса клиентской почтовой программы Microsoft Outlook Express, с основными приемами доставки и сохранения почтовых сообщений. Создать и отправить почтовое сообщение. Переслать почтовое сообщение. Сохранить документ, полученный электронной почтой.

Задание 20. Зарегистрироваться на бесплатном почтовом сервере и познакомиться с основными возможностями и элементами интерфейса Web-mail. Создать и отправить по электронной почте одно почтовое сообщение; написать ответ на полученное письмо; создать сообщение и вложить в него файл любого формата; сохранить вложенный в почтовое сообщение файл на локальном диске; переслать преподавателю полученное сообщение с вложением.

Задание 21. Создать простейший гипертекстовый документ средствами текстового редактора Блокнот, используя теги форматирования шрифта и абзаца.

Задание 22. Создать в блокноте файл Tab1.html, внутри которого поместить таблицы в соответствии с предложенными образцами.

Задание 23 Создать простейшую HTML-страницу с помощью языка программирования JavaScript.

Задание 24. Овладеть приемами разработки мультимедиа реферата с применением текстового процессора. Создать мультимедиа реферат по заданной теме.

Проектные задания

1. Расчет подсетей IPv4

Цель работы

- Изучить адресацию в IP сетях.
- Научиться рассчитывать адреса сетей и подсетей.
- Научиться определять маску и адреса устройств для подсети.

В соответствии с вариантом по заданным IPv4 адресу и маске подсети определить следующие параметры:

- Адреса сетей А и Б.
- Широковещательные адреса сетей А и Б.

- Максимальное количество узлов в сетях А и Б.
- Диапазон доступных адресов узлов в сетях А и Б.
- Количество возможных подсетей Б в сети А.

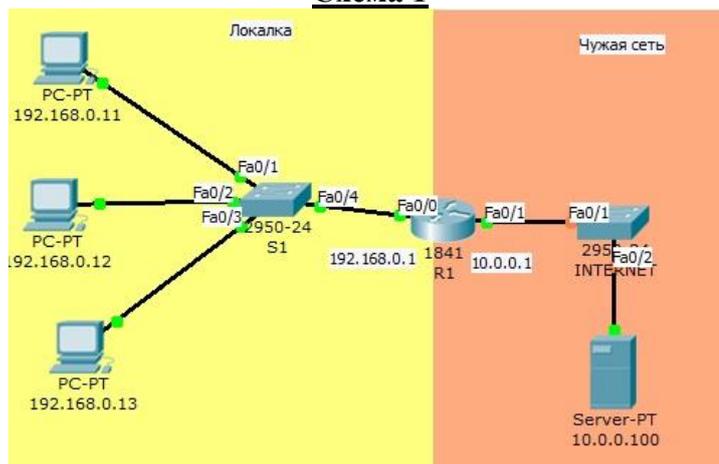
| Номер варианта | IP адрес | Маска сети А | Маска сети Б |
|----------------|-----------------|---------------|-----------------|
| 1 | 128.107.0.55 | 255.255.0.0 | 255.255.255.0 |
| 2 | 192.135.250.180 | 255.255.255.0 | 255.255.255.248 |
| 3 | 10.101.99.228 | 255.0.0.0 | 255.255.128.0 |
| 4 | 156.56.3.64 | 255.192.0.0 | 255.255.0.0 |
| 5 | 81.16.190.64 | 255.255.128.0 | 255.255.255.0 |
| 6 | 91.19.35.13 | 255.255.224.0 | 255.255.255.224 |
| 7 | 190.15.157.6 | 255.0.0.0 | 255.255.192.0 |
| 8 | 65.16.16.182 | 255.255.0.0 | 255.255.224.0 |
| 9 | 125.18.19.16 | 255.255.240.0 | 255.255.255.0 |
| 10 | 14.196.168.26 | 255.255.248.0 | 255.255.255.248 |

2. «Настройка NAT в Cisco Packet Tracer»

Преобразование сетевых адресов (NAT) — это процесс, при котором сетевое устройство, например маршрутизатор Cisco, назначает публичный адрес узловым устройствам в пределах частной сети. NAT используют для того, чтобы сократить количество публичных IP-адресов, используемых организацией, поскольку количество доступных публичных IPv4-адресов ограничено.

Соберите сеть согласно схеме 1

Схема 1



Две сети, 192.168.0.0/24 и 10.0.0.0/8. Одна названа локальной, вторая — чужой (например, сеть Интернет провайдера).

В локальной сети есть коммутатор, к которому подключено 3 узла. Их IP-адреса обозначены на схеме.

Задача: научиться делать трансляцию сетевых адресов тремя способами.

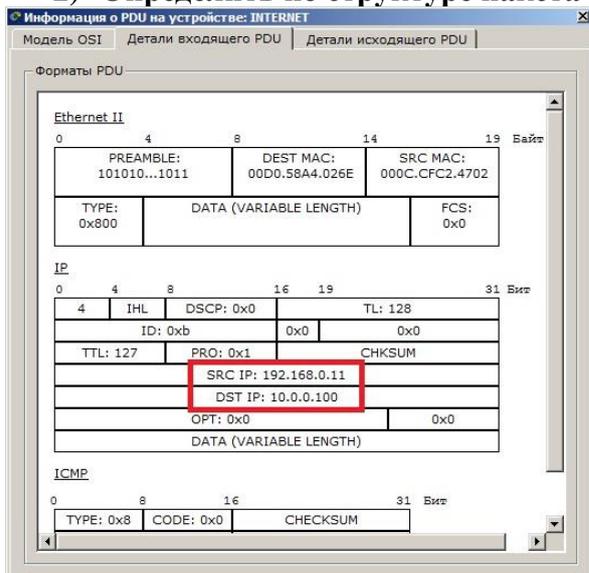
Для проверки работоспособности NAT – у вас должно быть построено три сети (своя на каждый из трех способов).

Задание 1

Организовать трансляцию адресов по следующей схеме:

| Source | New |
|--------------|-----------|
| 192.168.0.11 | 10.0.0.11 |
| 192.168.0.12 | 10.0.0.12 |
| 192.168.0.13 | 10.0.0.11 |

- 1) Отправить пакет на адрес 10.0.0.100.
- 2) Определить по структуре пакета адрес источника и получателя.



3. Настроить NAT.

Статическая трансляция сетевых адресов (Static NAT)

Настройка NAT на маршрутизаторах Cisco под управлением IOS включает в себя следующие шаги

1. Назначить внутренний (Inside) и внешний (Outside) интерфейсы. Внутренним интерфейсом обычно выступает тот, к которому подключена локальная сеть. Внешним — к которому подключена внешняя сеть, например сеть Интернет провайдера.
 2. Определить для кого (каких IP-адресов) стоит делать трансляцию.
 3. Выбрать какой вид трансляции использовать
 4. Осуществить проверку трансляций
- Существует три вида трансляции Static NAT, Dynamic NAT, Overloading.

Команда: ip nat inside source static АдресОтправителя МаскируемыйАдрес.

Пример:

```
Router(config)#ip nat inside source static 192.168.0.11 10.0.0.11
```

Аналогично настроить для адресов 192.168.0.12 и 192.168.0.13 (в соответствии с таблицей)

| Source | New |
|--------|-----|
|--------|-----|

| | |
|--------------|-----------|
| 192.168.0.11 | 10.0.0.11 |
| 192.168.0.12 | 10.0.0.12 |
| 192.168.0.13 | 10.0.0.11 |

На нужном сетевом интерфейсе указать, где находится внешняя сторона NAT:
 Router(config-if)#ip nat outside

Набор команд для схемы 1:

```
Router>en
Router#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#ip nat inside source static 192.168.0.11 10.0.0.11
Router(config)#ip nat inside source static 192.168.0.12 10.0.0.12
Router(config)#int fa0/0
Router(config-if)#ip nat inside
Router(config-if)#
Router(config-if)#int fa0/1
Router(config-if)#ip nat outside
Router(config-if)#exit
Router(config)#exit
Router#show ip nat translations
Pro Inside global Inside local Outside local Outside global
--- 10.0.0.11 192.168.0.11 --- ---
--- 10.0.0.12 192.168.0.12 --- ---

Router#
```

1. Отправьте пакеты в другую сеть.
2. Определить по заголовку пакета, был ли изменен IP-адрес отправителя.

Задание 2. Динамическая трансляция сетевых адресов.

Настройка Dynamic NAT

permim: число от 1 до 99 обозначает № списка доступа и задается администратором.
 source-wildcard – список доступа, по которому разрешается передавать пакеты сети (причём используется инверсная маска) – можно написать Any- ключевое слово, означает, что список доступа будет разрешать пакеты с любым адресом отправителя.
 router(config)#access-list permit [source-wildcard]

При задании пула адресов необходимо указать первый и последний адреса из входящей в пул последовательности адресов. Если в пуле 1 адрес (как в нашем случае) необходимо указать его 2 раза.

```
router(config)#ip nat pool name start-ip end-ip { netmask netmask | prefix-length prefix-length }
```

применить этот пул на правило
 router(config)#ip nat inside source list pool

На нужном сетевом интерфейсе указать, где находится внутренняя и внешняя сторона
 router(config)#interface fa0/4
 router(config-if)#ip nat inside

```
router(config-if)#exit
router(config)#interface s0
router(config-if)#ip nat outside
```

В схеме 1 – измените адрес роутера во внешней сети на 192.168.0.2. Поменяйте ссылки на новый шлюз у трех компьютеров внутренней сети.

Перед конфигурацией проверьте пересылку пакетов от узла до роутера. Если ошибка – проверьте ссылку на шлюзы (на маршрутизатор) у компьютеров в сети. Попробуйте снова изменить адрес маршрутизатора.

Набор команд для схемы 1:

```
Router>en
Router#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#access-list 1 permit 192.168.0.0 0.0.0.255
Router(config)#ip nat pool mypool 10.0.0.30 10.0.0.39 netmask 255.0.0.0
Router(config)#ip nat inside source list 1 pool mypool
Router(config)#int Fa0/0
Router(config-if)#ip nat inside
Router(config-if)#int Fa0/1
Router(config-if)#ip nat outside
Router(config-if)#exit
Router(config)#
```

**Отправить пакет из внешней сети во внутреннюю
Определить по заголовку пакета, был ли изменен IP-адрес отправителя.**

Задание 3.

Port Address Translation(PAT) configuration in Packet Tracer

PAT (Port Address Translation) — технология трансляции адресов с использованием портов. Данная технология решает проблему **доставки возвратных пакетов**. Так как количество белых IP **ограничено** нам необходимо экономить эти адреса. Помня об этом, была создана технология PAT. Она позволяет **локальным хостам** использовать **частные IP-адреса** и установить один зарегистрированный адрес на маршрутизатор доступа. В технологии преобразования адресов PAT используется особенность работы протокола TCP: с точки зрения сервера абсолютно все равно, осуществляются соединения с тремя разными хостами с разными адресами или соединения устанавливаются с одним хостом на один IP-адрес, но с разными портами. Следовательно, чтобы подключить к Интернету множество хостов небольшого офиса с помощью **одного** только зарегистрированного **публичного IP адреса**, служба PAT транслирует частные адреса локальных хостов в один имеющийся зарегистрированный. Чтобы правильно пересылать пакеты обратной коммуникации локальным хостам, маршрутизатор хранит у себя таблицу IP адресов и номеров портов для протоколов TCP и UDP.

Набор команд для схемы 1

192.168.0.3 – маршрутизатор во внутреннюю сеть
10.0.0.2 – маршрутизатор во внешнюю сеть

```
Router(config)#interface FastEthernet0/0
```

```

Router(config-if)#ip address 192.168.0.3 255.255.255.0
Router(config-if)#ip nat inside
Router(config-if)#no shutdown
Router(config-if)#exit
Router(config)#int Fa0/1
Router(config-if)#ip address 10.0.0.2 255.0.0.0
Router(config-if)#ip nat outside
Router(config-if)#no shutdown
Router(config-if)#exit
Router(config)#access-list 2 permit 192.168.0.0 0.0.0.255
Router(config)#ip nat inside source list 2 interface Fa0/1 overload
Router(config)#end
Router#
%SYS-5-CONFIG_I: Configured from console by console

```

```

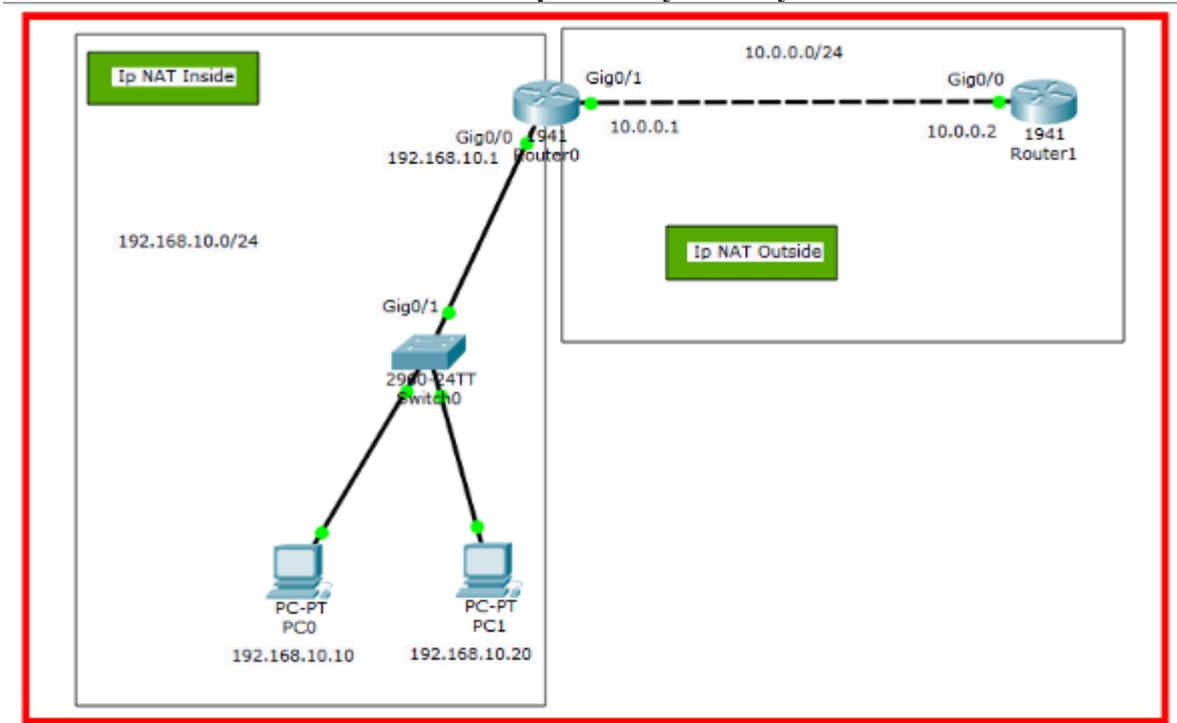
Router#wr
Building configuration...
[OK]
Router#

```

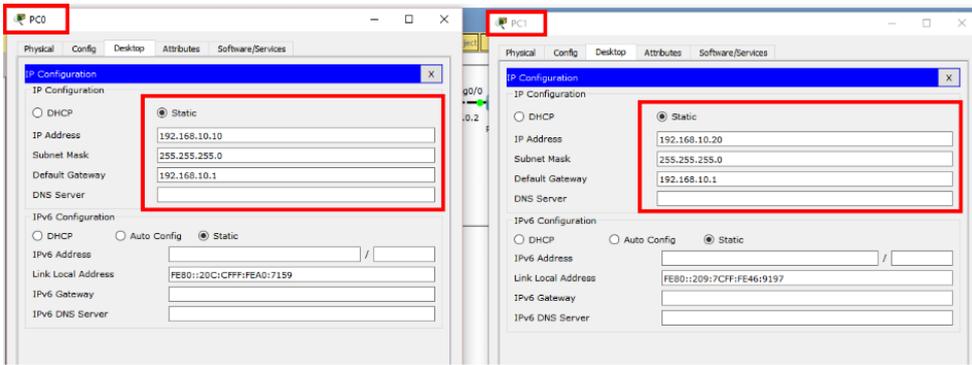
Самостоятельно:

1. На оценку «3» (50% баллов) соберите схему 2 и проведите настройку по образцу
2. На оценки «4» и «5» выполните задание по варианту

Схема 2
Соберите новую схему



1. Настройка компьютеров

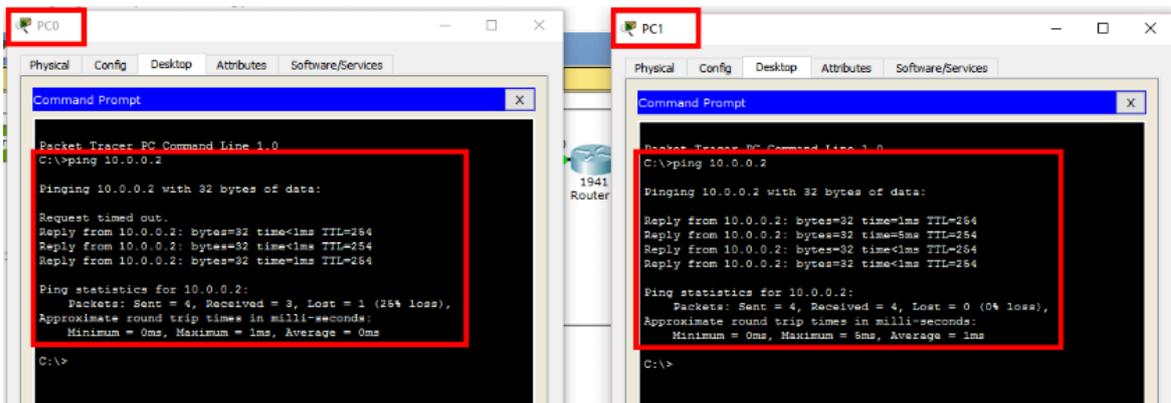


1. Настройки маршрутизатора

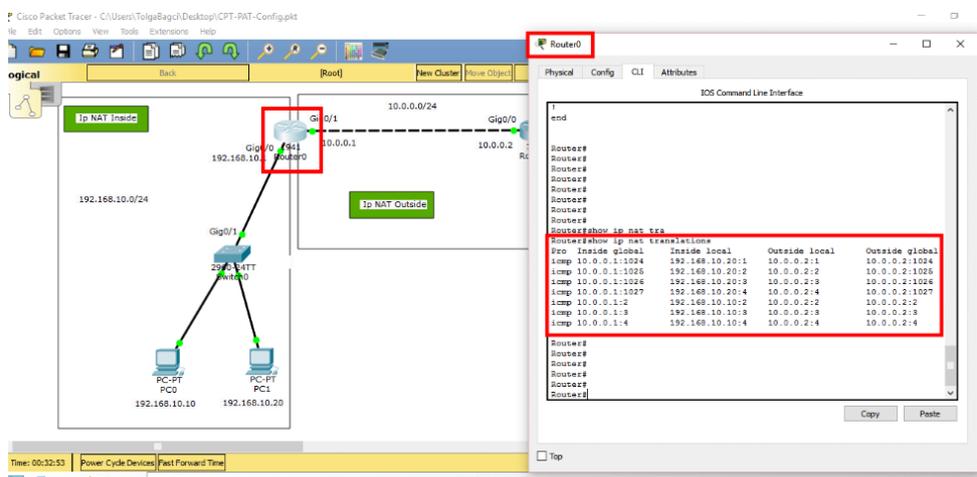
```

Router# conf t
Router(config)# interface gigabitethernet 0/0
Router(config-if)# ip address 192.168.10.1 255.255.255.0
Router(config-if)# ip nat inside
Router(config-if)# no shutdown
Router(config-if)# exit
Router(config)# interface gigabitethernet 0/1
Router(config-if)# ip address 10.0.0.1 255.255.255.0
Router(config-if)# ip nat outside
Router(config-if)# no shutdown
Router(config-if)# exit
Router(config)# access-list 1 permit 192.168.10.0 0.0.0.255
Router(config)# ip nat inside source list 1 interface gigabitethernet0/1 overload
Router(config)# end
Router# wr
  
```

3. Проверка пингов с внешней сетью



2. Просмотр таблицы соответствия адресов



```

Router0#show ip nat translations
Pro Inside global Inside local Outside local Outside global
icmp 10.0.0.1:1024 192.168.10.20:1 10.0.0.2:1 10.0.0.2:1024
icmp 10.0.0.1:1025 192.168.10.20:2 10.0.0.2:2 10.0.0.2:1025
icmp 10.0.0.1:1026 192.168.10.20:3 10.0.0.2:3 10.0.0.2:1026
icmp 10.0.0.1:1027 192.168.10.20:4 10.0.0.2:4 10.0.0.2:1027
icmp 10.0.0.1:2 192.168.10.10:2 10.0.0.2:2 10.0.0.2:2
icmp 10.0.0.1:3 192.168.10.10:3 10.0.0.2:3 10.0.0.2:3
icmp 10.0.0.1:4 192.168.10.10:4 10.0.0.2:4 10.0.0.2:4

```

3. Просмотр дополнительной информации

```

Router0#show ip nat statistics
Total translations: 0 (0 static, 0 dynamic, 0 extended)
Outside Interfaces: GigabitEthernet0/1
Inside Interfaces: GigabitEthernet0/0
Hits: 7 Misses: 8
Expired translations: 8
Dynamic mappings:

```

```

Router0#show running-config
Building configuration...
Current configuration : 772 bytes
!
version 15.1
no service timestamps log datetime msec
no service timestamps debug datetime msec
no service password-encryption
!
hostname Router
!
ip cef
no ipv6 cef
!
license udi pid CISCO1941/K9 sn FTX1524V40L
!
spanning-tree mode pvst
!
interface GigabitEthernet0/0
ip address 192.168.10.1 255.255.255.0
ip nat inside
duplex auto
speed auto
!
interface GigabitEthernet0/1

```

Варианты:

| Варианты | Внутренняя сеть | Внешняя сеть |
|----------|-----------------|--------------|
| 1 | 192.168.0.0 /24 | 15.2.0.0 /24 |
| 2 | 192.168.0.0 /25 | 10.0.0.0 /16 |

| | | |
|----|------------------|------------------|
| 3 | 192.168.0.0 /26 | 11.0.0.0 /8 |
| 4 | 192.168.1.0 /24 | 10.0.1.0 /24 |
| 5 | 192.168.1.0 /25 | 10.0.2.0 /24 |
| 6 | 192.168.1.0 /26 | 10.0.3.0 /24 |
| 7 | 192.168.2.0 /24 | 10.0.4.0 /24 |
| 8 | 192.168.2.0 /25 | 10.1.0.0 /16 |
| 9 | 192.168.2.0 /26 | 10.2.0.0 /16 |
| 10 | 192.168.3.0 /24 | 10.3.0.0 /16 |
| 11 | 192.168.3.0 /25 | 10.4.0.0 /16 |
| 12 | 192.168.3.0 /26 | 10.0.5.0 /25 |
| 13 | 172.16.0.0 /24 | 10.0.5.0 /26 |
| 14 | 172.16.0.0 /25 | 10.0.4.0 /25 |
| 15 | 172.16.0.0 /26 | 10.0.4.0 /26 |
| 16 | 172.16.1.0 /24 | 10.0.3.0 /25 |
| 17 | 172.16.1.0 /25 | 10.0.3.0 /26 |
| 18 | 172.16.1.0 /26 | 10.0.2.0 /25 |
| 19 | 172.16.2.0 /24 | 10.0.2.0 /26 |
| 20 | 172.16.2.0 /25 | 10.10.10.0 /24 |
| 21 | 172.16.2.0 /26 | 10.10.11.0 /24 |
| 22 | 172.16.3.0 /24 | 10.10.12.0 /24 |
| 23 | 172.16.3.0 /25 | 10.10.13.0 /24 |
| 24 | 172.16.3.0 /26 | 10.10.14.0 /24 |
| 25 | 172.16.0.128 /25 | 10.10.15.0 /24 |
| 26 | 172.16.0.192 /26 | 10.10.10.128 /25 |
| 27 | 172.16.0.0 /23 | 10.10.10.192 /26 |
| 28 | 172.16.0.0 /22 | 10.10.0.0 /16 |
| 29 | 192.168.0.0 /23 | 10.10.0.0 /24 |
| 30 | 192.168.0.0 /22 | 10.10.192.0 /17 |

3. Настройка IPv6

IPv4 имеет 4,3 миллиарда адресов, что может показаться невероятным. Однако потребовалось всего два десятилетия, чтобы она достигла своего истощения. На помощь пришел IPv6 в виде 128-битных адресов. Packet Tracer поддерживает широкий спектр функций IPv6.

Задания:

1. Ознакомьтесь с теоретической частью
2. Ответьте на контрольные вопросы
3. Выполните практикум (представьте отчет со снимками экрана и файлы проектов)

Теоретическая часть

IPv6 - это новейшая версия IP-протокола, которую люди часто называют «Интернет-протоколом нового поколения». Он был разработан как ответ на многие недостатки IPv4, в первую очередь на проблему исчерпания адресов IPv4. IPv6 используется для тех же общих функций, что и IPv4, но с другой реализацией.

Давайте рассмотрим некоторые из **наиболее важных функций IPv6**.

- **Существует очень большое количество адресов IPv6**, поскольку адреса IPv6 128-битные, в отличие от адресов IPv4, которые являются 32-битными.

- **IPv6 имеет более простой заголовок** - потому что в заголовке IPv4 нет битов контрольной суммы. По этой причине маршрутизаторам не нужно вычислять контрольную сумму для каждого пакета.

- **В IPv6 есть автоконфигурация адресов без сохранения состояния** : - Хосты автоматически настраиваются с помощью адресов IPv6.
- **Нет необходимости в NAT**, поскольку каждое устройство в сети IPv6 имеет глобально уникальный IPv6-адрес.

Формат IPv6-адреса.

Адрес IPv6 имеет длину **128 бит** и использует **8 групп по 4 шестнадцатеричных** цифры, разделенных двоеточиями. Адрес IPv6 выглядит так:

FE80 : 0A3B : 0002 : 4B3C : 0F6D : 3D40 : FFC5 : 005A

Как сократить IPv6-адрес.

У нас есть 2 правила для этого:

Правило №1: Начальный ноль можно опустить.

Например, рассмотрим следующий IPv6-адрес:

FE80 : 0 A : 000 2 : 4B3C : 0 F6D : 3D40 : FFC5 : 00 5A

Если опустить все ведущие нули (синего цвета), адрес сократится до:

FE80: A3B: 2: 4B3C: F6D: 3D40: FFC5: 5A

Правило # 2: Последовательные группы нулей могут быть представлены в виде двух двоеточий (::)

Например, рассмотрим адрес ниже:

2002 : 0000 : 0000 : A29F : 7D12 : 5502 : 63AF : BD2C

Замените следующие друг за другом группы нулей (красного цвета) двойным двоеточием, и сокращенная версия будет выглядеть так:

2002 :: A29F : 7D12 : 5502

Что, если бы вам сказали сократить этот адрес:

2002 : 0000 : 0000 : 2B3C : 0000 : 0000 : 0000 : 55DA

Будет ли сокращенная версия выглядеть так -> **2002 :: 2B3C :: 55DA ? НЕТ !**

Вместо этого, сокращенный вариант должен быть либо 2002 : 0 : 0 : 2B3C :: 55DA или выглядеть как 2002 :: 2B3C : 0 : 0 : 0 : 55DA.

Почему это так? У вас **не может быть более одного вхождения двойного двоеточия** в сокращенной версии, потому что в противном случае вы бы не знали, сколько наборов нулей было пропущено для каждой части.

Типы IPv6-адресов.

В IPv4 есть 3 типа адресов: *одноадресный, многоадресный и широковещательный*. В IPv6 у нас больше нет широковещательной передачи. Адреса широковещательной рассылки были удалены и заменены адресами произвольной и многоадресной рассылки.

Итак, для IPv6 у нас есть **одноадресная**, **многоадресная** и **произвольная рассылка**.

Одноадресный адрес - определяет один интерфейс. Пакеты, адресованные на одноадресный адрес, доставляются на единственный интерфейс.

Многоадресный адрес - определяет определенную группу хостов. Как и в случае с IPv4, пакеты, отправленные на адрес многоадресной рассылки, доставляются на все интерфейсы, идентифицированные этим адресом многоадресной рассылки. Многоадресные IPv6-адреса всегда начинаются с **FF**.

Anycast-адрес - как и многоадресный адрес, идентифицирует несколько интерфейсов, но есть разница: пакеты, отправленные на произвольный адрес, доставляются на один адрес - адрес, который является ближайшим к источнику с точки зрения расстояния маршрутизации. Например, мы можем назначить один и тот же произвольный адрес серверам, предлагающим аналогичные услуги. Пакеты, отправленные на этот IP-адрес, будут перенаправлены на ближайший сервер. По этой причине вы можете называть произвольный адрес адресом **«один к одному из многих»**. Адреса Anycast используются для балансировки **нагрузок**.

Типы одноадресных адресов IPv6

Одноадресные IPv6-адреса бывают **трех** типов:

Global одноадресного - Это публично **маршрутизируемые** IPv6 адрес похожий на публичные адреса IPv4. Они начинаются с **2000 :: / 3**.

Link local - они похожи на адреса IPv4 в диапазоне APIPA. Эти адреса можно использовать только в том сегменте сети, к которому подключен хост. Маршрутизаторы не будут пересылать пакеты, предназначенные для локального адреса ссылки, на другие ссылки. Локальный адрес канала должен быть назначен каждому сетевому интерфейсу, на котором включен протокол IPv6. Эти адреса с префиксом **FE80 :: / 10**

Уникальные локальные - они имеют те же функции, что и частные адреса в IPv4: разрешить обмен данными во всем сетевом сегменте с возможностью маршрутизации в несколько локальных сетей. **Уникальные** локальные адреса начинаются с префикса **FD00 :: / 8**.

Расчет IPv6 EUI-64

Мы уже видели, что IPv6-адрес 128-битный. Стоит знать, что IPv6 индивидуальный адрес состоит из **идентификатора подсети** и **идентификатора интерфейса**.

Например, глобальный одноадресный адрес имеет 64-битный идентификатор подсети и 64-битный идентификатор интерфейса. Идентификатор подсети содержит **префикс сайта** и **идентификатор подсети** (подсети внутри сайта). ID интерфейса состоит из части MAC-адреса интерфейса.

Давайте посмотрим, как получить идентификатор интерфейса из MAC-адреса устройства, например ПК.

MAC-адрес обычно имеет длину **48 бит**. А теперь мы хотим сгенерировать **64-битный идентификатор интерфейса**, используя этот MAC. Как мы делаем это? Как вы можете догадаться, мы добьемся этого, добавив к MAC-адресу 16 бит, чтобы получить 64-битный идентификатор. 16 бит эквивалентны 4 шестнадцатеричным цифрам. Поскольку IPv6-адреса обычно выражаются в шестнадцатеричном формате, мы обычно вставляем шестнадцатеричное число **FFFE** в середину MAC-адреса, чтобы получить идентификатор интерфейса.

Итак, чтобы получить идентификатор интерфейса с помощью метода EUI-64:

1. Разделите MAC-адрес на **две половины** (каждая половина будет иметь 6 шестнадцатеричных цифр).

2. Вставьте **FFFE** между двумя половинами, чтобы создать **идентификатор интерфейса**.

3. Инвертируйте **седьмой бит** идентификатора интерфейса.

Например, если MAC-адрес сетевой карты ПК **00: 00: AA: BB: FF: 55**

Разделение MAC на 2 половины и вставка **FFFE** в середину дает:

0000AA FFFE BBFF55

Теперь перевернем седьмой бит этого идентификатора. Для этого мы сначала запишем его в двоичном формате (помните, что каждая шестнадцатеричная цифра эквивалентна 4 битам)

0000 0000 0000 0000 1010 1010 1111 1111 1111 1110 1011 1011 1111 1111 1001 0101

Таким образом, перевернув 7-й бит, мы получим:

0000 00**1**0 0000 0000 1010 1010 1111 1111 1111 1110 1011 1011 1111 1111 1001 0101

Итак, теперь наш измененный идентификатор интерфейса - **0200: AAFF: FEBB: FF55**.

Пример настройки параметров IPv6 на интерфейсе маршрутизатора. Cisco packet tracer.

R1>en // Переходим в привилегированный режим EXEC

R1#conf t // Переходим в режим глобальной конфигурации

Enter configuration commands, one per line. End with CNTL/Z.

R1(config)#interface Serial1/0 // Выбираем интерфейс Serial1/0 для дальнейшей настройки

R1(config-if)#ipv6 address 2001:db8:acad:1::1/64 // Присваиваем адрес IPv6

R1(config-if)#no shutdown // Включаем интерфейс

R1(config-if)#exit

Важно : прежде чем вы сможете использовать IPv6-адресацию на маршрутизаторе, не забудьте сначала включить IPv6-маршрутизацию на маршрутизаторе с помощью команды IPv6 unicast-routing из его режима глобальной конфигурации.

Контрольные вопросы:

1. Необходимость использования IPv6.
2. Виды адресов IPv6.
3. Упростите адреса IPv6

2002:0000:3D12:25AF:2788:00AB:03AF:002C

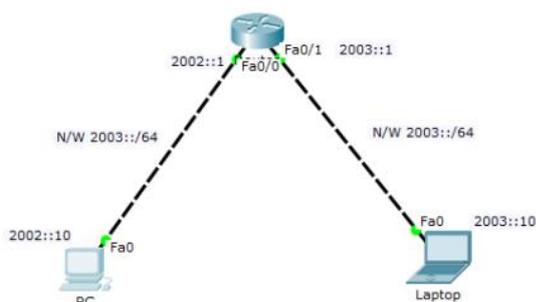
FE80:A290:0000:0000:4B2C:0000:0000:45DB

2002:0000:0000:2B3C:0000:0000:0000:00DA

4. На основе мас адреса 00-50-B6-5B-CA-6A создайте адрес IPv6.

Практикум 1:

Постройте топологию сети.



Настройте адреса IPv6 на интерфейсах маршрутизатора.

```
Router#config t
Router(config)#ipv6 unicast-routing
Router(config)#int fa0/0
Router(config-if)#ipv6 address 2002::1/64
Router(config-if)#no shut
Router(config-if)#
Router(config-if)#int fa0/1
Router(config-if)#ipv6 address 2003::1/64
Router(config-if)#no shut
```

Теперь настройте статические IPv6-адреса на ПК и ноутбуке.

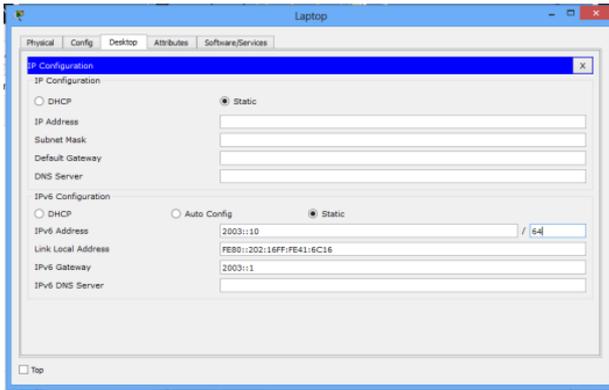
Ноутбук:

IPv6-адрес 2003 :: 10 Шлюз по умолчанию 2003 :: 1

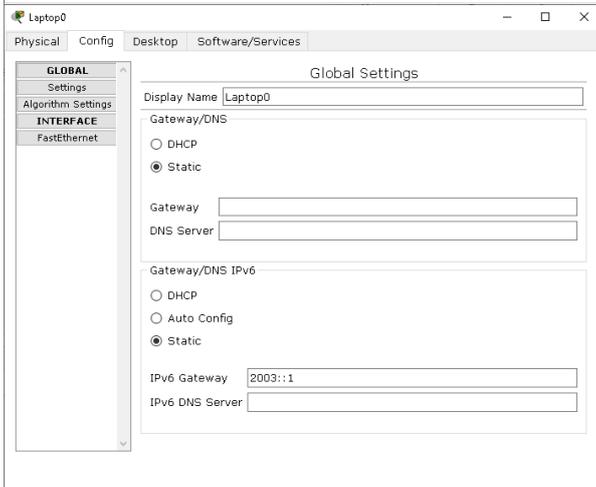
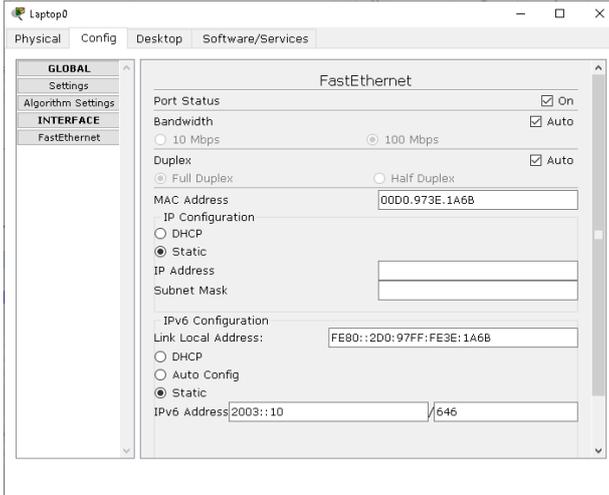
Link Local Address набирать не нужно, он автоматически сформируется по EUI-64.

Через Desktop

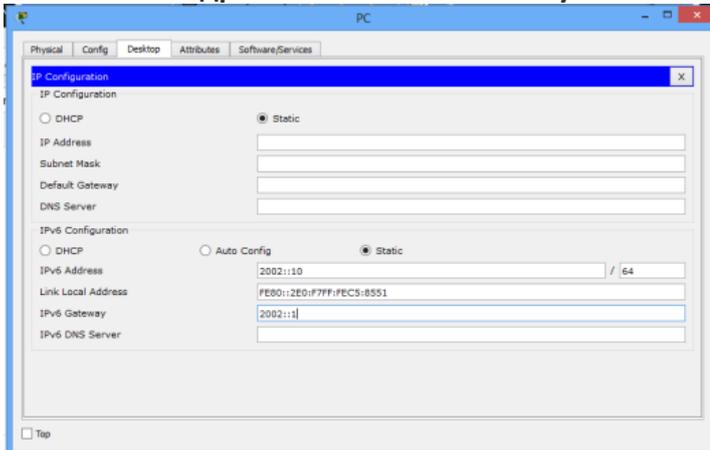
Начиная с версии Packet Trace Version 6, служебная программа **IP Configuration** на вкладке **Desktop** оконечных устройств имеет возможность вводить IPv6-адрес.



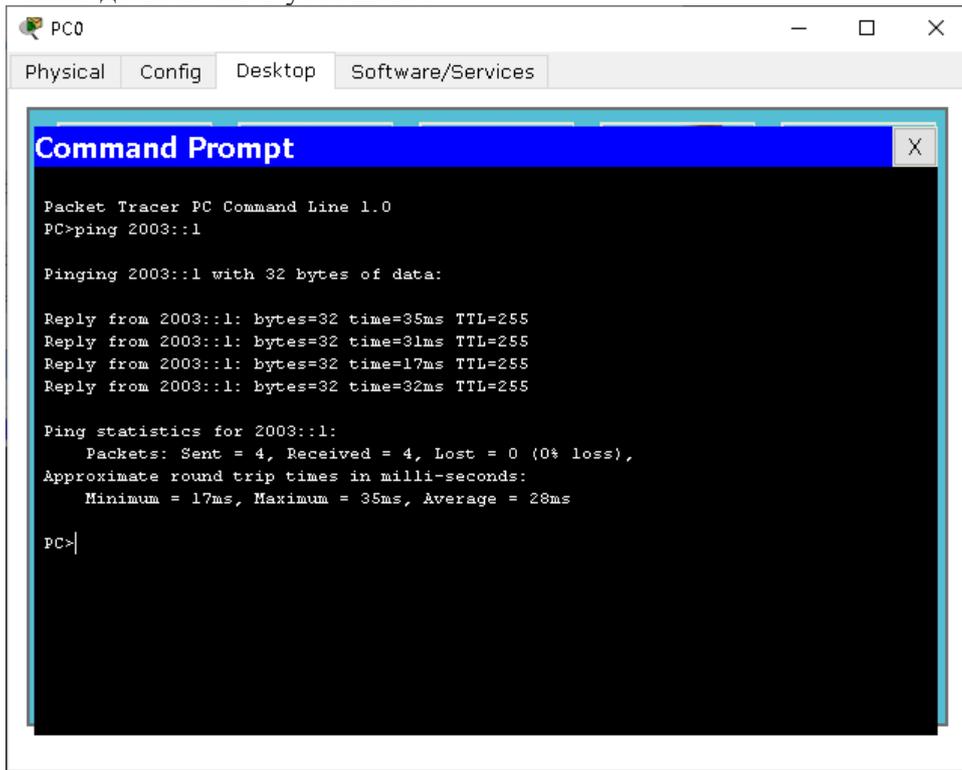
Или Настройки (для более ранних версий)



ПК : IPv6-адрес 2002 :: 10 Шлюз по умолчанию 2002 :: 1



Проверьте конфигурацию IPv6 с помощью команды **ping**.
Сначала выполните эхо-запрос портативного компьютера с ПК, используя его IPv6-адрес. Пинг должен быть успешным.



```
PC0
Physical Config Desktop Software/Services
Command Prompt
Packet Tracer PC Command Line 1.0
PC>ping 2003::1

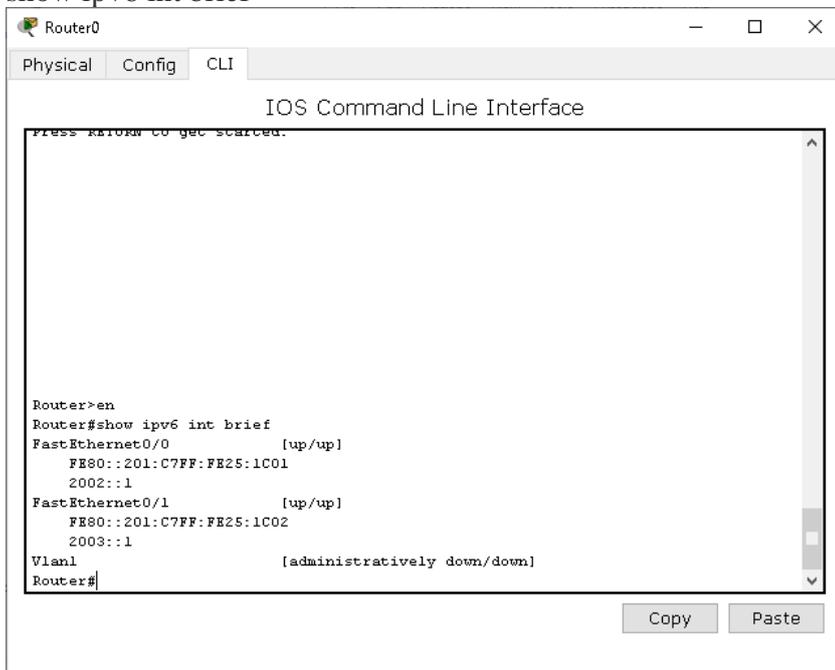
Pinging 2003::1 with 32 bytes of data:

Reply from 2003::1: bytes=32 time=35ms TTL=255
Reply from 2003::1: bytes=32 time=31ms TTL=255
Reply from 2003::1: bytes=32 time=17ms TTL=255
Reply from 2003::1: bytes=32 time=32ms TTL=255

Ping statistics for 2003::1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 17ms, Maximum = 35ms, Average = 28ms

PC>
```

Затем мы проверим конфигурацию IPv6-адреса на маршрутизаторе.
show ipv6 int brief



```
Router0
Physical Config CLI
IOS Command Line Interface
Press RETURN to get started.

Router>en
Router#show ipv6 int brief
FastEthernet0/0          [up/up]
  FE80::201:C7FF:FE25:1C01
  2002::1
FastEthernet0/1          [up/up]
  FE80::201:C7FF:FE25:1C02
  2003::1
Vlan1                    [administratively down/down]
Router#
```

Мы можем убедиться, что:

1. **Локальный IPv6-** адрес канала (начиная с **FE80**) был автоматически настроен для каждого интерфейса.
2. Вы также можете увидеть **статический адрес, который** мы только что настроили - **2002 :: 1** для int fa0 / 0 и **2003 :: 1** для fa0 / 1.

Практикум 2

Локальный адрес ссылки был автоматически настроен на каждом интерфейсе маршрутизатора. Это происходит, когда маршрутизатор объединяет **идентификатор**

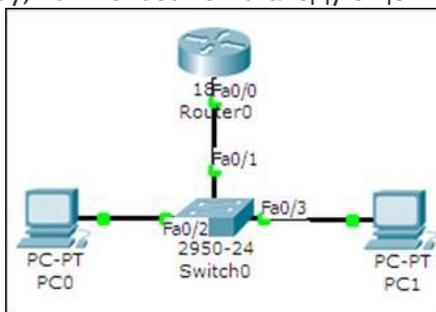
префикса для локального адреса канала (который всегда начинается с **FE80**) с его **измененным идентификатором интерфейса** (полученным с помощью метода EUI-64), чтобы сформировать локальный для канала IPv6-адрес.

Аналогичным образом ПК автоматически настроится с использованием локального адреса ссылки.

Стоит отметить: **автоконфигурация** выполняется не только для локальных адресов, но и для других одноадресных IPv6-адресов (глобальных одноадресных и уникальных локальных).

В приведенном выше примере ПК сначала получает идентификатор префикса уникального локального адреса из объявления маршрутизатора (RA). Здесь ПК получает идентификатор префикса интерфейса fa0 /0, который равен 2002 :: / 64. Затем ПК выполнит EUI-64, используя свой MAC-адрес для получения идентификатора интерфейса. В сочетании префикс и идентификатор интерфейса образуют уникальный локальный адрес ПК. Поскольку DHCP не использовался и уникальный локальный адрес не был статически настроен на ПК, мы говорим, что ПК автоматически настроил себя с уникальным локальным адресом.

Создайте новую топологию, состоящую из двух компьютеров и маршрутизатора, подключенного к коммутатору, как показано на следующем снимке экрана:



Автоконфигурация требует наименьшего количества настроек, но затрудняет запоминание адресов IPv6. Этот метод использует MAC-адрес устройства для создания IPv6-адреса с префиксом FE80 ::. Выполните следующие шаги, чтобы назначить адреса IPv6 с помощью автоконфигурации:

Начните с настройки роутера. Войдите в режим настройки интерфейса и включите IPv6 на интерфейсе.

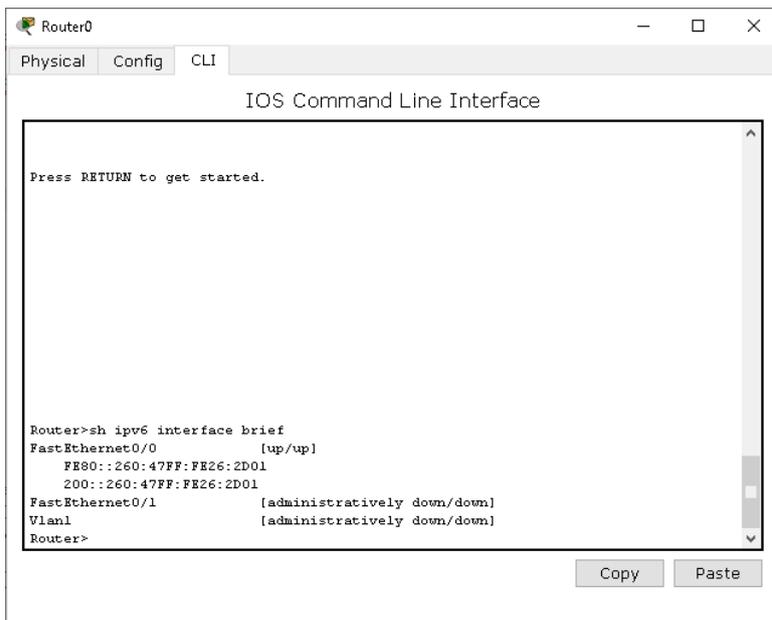
1. **R0 (config) # ipv6 unicast-routing**
2. **R0 (config) #interface FastEthernet0 / 0**
3. **R0 (config-if) # ipv6 enable**

Затем мы настроим для этого интерфейса локальный адрес ссылки и глобальный одноадресный адрес. Мы будем использовать eui-64, чтобы уменьшить конфигурацию.

4. **R0(config-if)#ipv6 address autoconfig**
5. **R0(config-if)#ipv6 add 2000::/64 eui-64**
6. **R0(config-if)#no shutdown**

Убедитесь, что интерфейс включен и имеет два адреса IPv6.

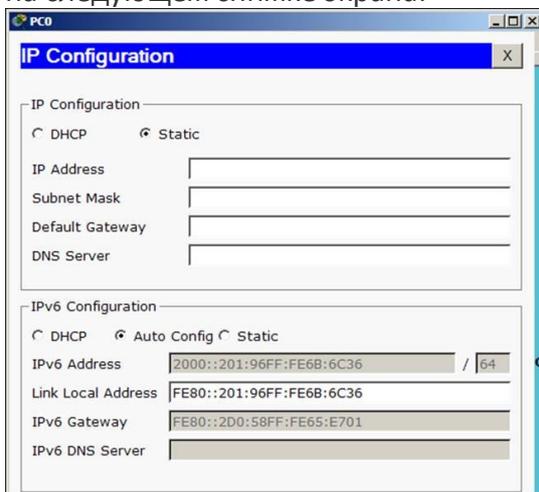
7. **R0>sh ipv6 interface brief**



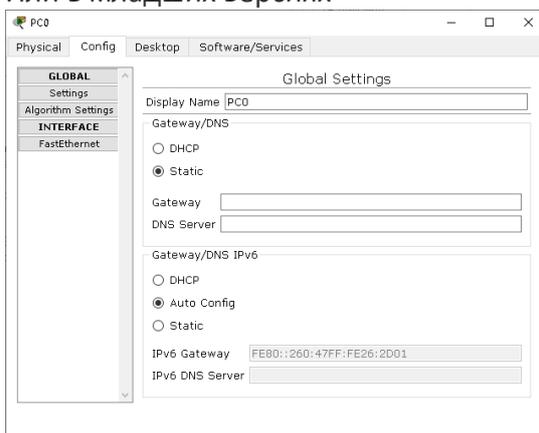
Эти IPv6-адреса могут отличаться, когда вы их опробуете, поскольку они основаны на MAC-адресе. Включите маршрутизацию, чтобы этот маршрутизатор можно было идентифицировать как шлюз по умолчанию.

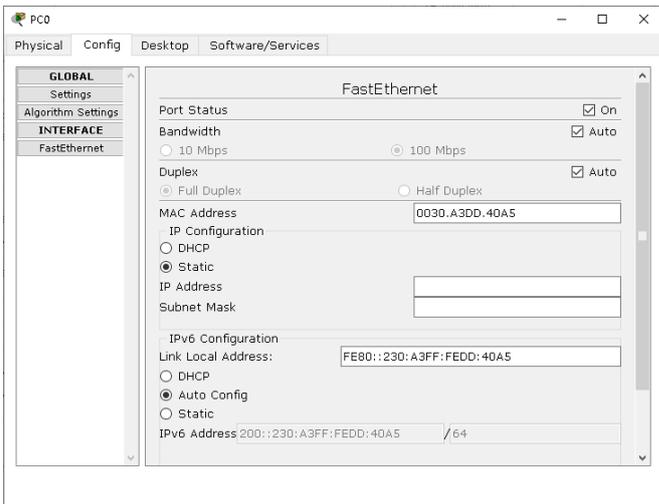
15. R0 (config) # ipv6 unicast-routing (одноадресная маршрутизация)

На этом настройка роутера завершена, перейдем к ПК. Перейдите на вкладку «Рабочий стол» ПК, откройте «Конфигурация IP» и в разделе «Конфигурация IPv6» выберите «Автоконфигурация». Шлюз и IP-адрес ПК будут назначены автоматически, как показано на следующем снимке экрана:



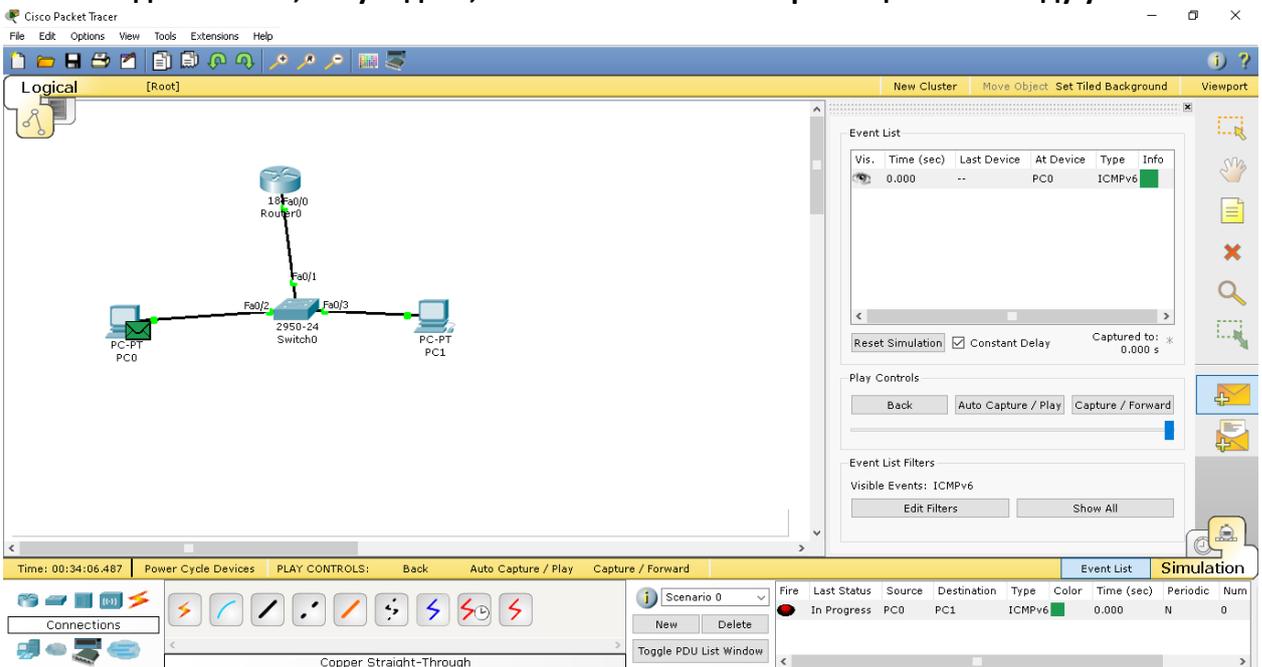
Или в младших версиях





Задания:

1. Используйте простой инструмент PDU для проверки возможности подключения; вы увидите, как пакеты ICMPv6 перемещаются между узлами.



2. Чтобы просмотреть IPv6-адрес из командной строки ПК, используйте команду `ipv6config`.

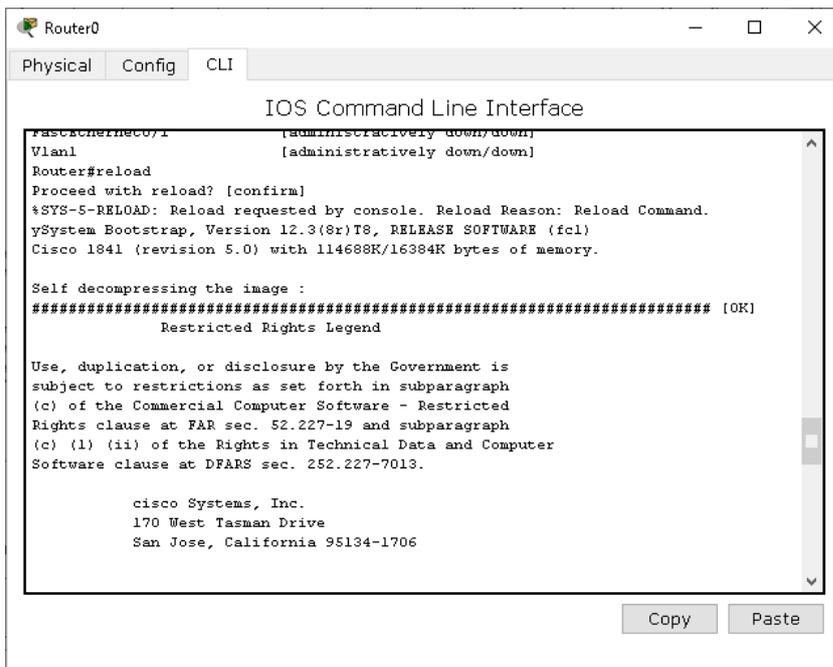
Практикум 3. Маршрутизация

Адреса IPv6 назначим статически на всех устройствах. Мы будем использовать ту же топологию. Мы выполним следующие шаги для статической настройки IPv6-адресов:

Начните с настройки статического IPv6-адреса на маршрутизаторе.

Выполните перезагрузку роутера (чтоб очистить предыдущие настройки)

`reload`



R0(config)#interface fastethernet0/0

R0(config-if)#ipv6 enable

R0(config-if)#ipv6 address 2000::1/64

R0(config-if)#no shutdown

Настройте адреса 2000::N+1 и 2000::N+2 на компьютерах (где N – номер вашего варианта).

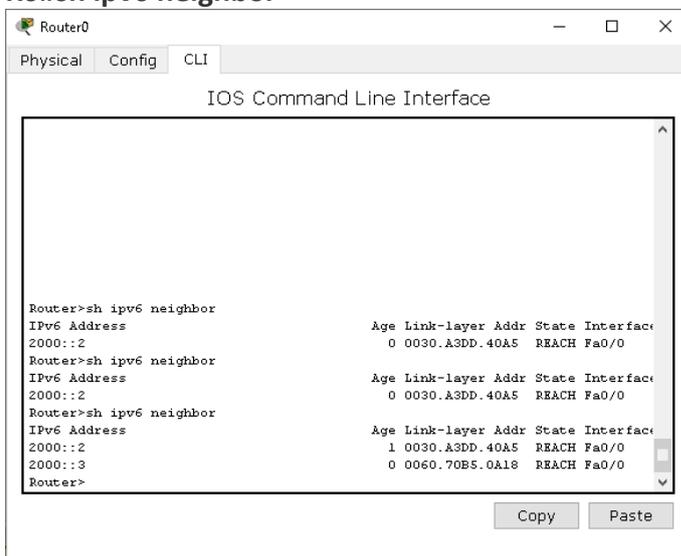
Не забудьте каждому компьютеру указать шлюз по умолчанию (адрес роутера).

Используйте простой инструмент PDU для проверки возможности подключения.

Сделайте запросы от персональных компьютеров к роутеру.

Вы можете взглянуть на таблицу соседей IPv6. Это похоже на таблицу ARP IPv4.

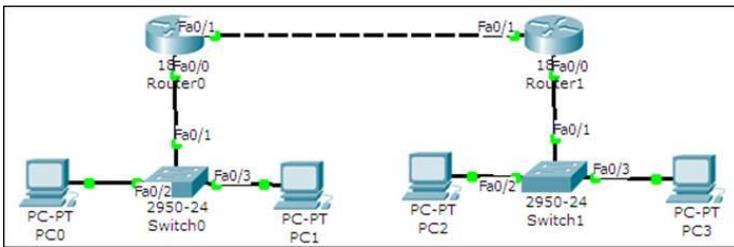
R0#sh ipv6 neighbor



Теперь, когда мы настроили IPv6-адреса в одной сети, давайте настроим их в большем количестве сетей и включим маршрутизацию между ними.

Статическая маршрутизация

Изменив ту же топологию, которую мы использовали ранее, давайте добавим маршрутизатор, коммутатор и два ПК для создания отдельной сети, как показано на следующем снимке экрана:



Вот таблица с описанием топологии (N – номер вашего варианта):

| Устройство | Интерфейс | Адрес |
|------------|-------------------|----------------|
| R0 | FastEthernet0 / 0 | 2000:1::1/64 |
| | FastEthernet0 / 1 | 2001::10/64 |
| ПК0 | FastEthernet | 2000:1::N+1/64 |
| ПК1 | FastEthernet | 2000:1::N+2/64 |
| R1 | FastEthernet0 / 0 | 2000:2::1/64 |
| | FastEthernet0 / 1 | 2001::20/64 |
| ПК2 | FastEthernet | 2000:2::N+1/64 |
| ПК3 | FastEthernet | 2000:2::N+2/64 |

После того, как необходимые IP-адреса и шлюзы были назначены, откройте вкладку **CLI** для маршрутизатора **R0** и начните настройку маршрутизации, выполнив следующие команды:

R0(config)#ipv6 unicast-routing

R0(config)#ipv6 route 2000:2::/64 2001::20

Принцип последней команды: если приходит пакет в сеть, то куда его отправить.

```

Router0
  Physical Config CLI
  IOS Command Line Interface
  Router(config-if)#
  Router(config-if)#exit
  Router(config)#interface FastEthernet0/1
  Router(config-if)#exit
  Router(config)#int fa0/0
  Router(config-if)#ipv6 enable
  Router(config-if)#ipv6 address 2000:1::1/64
  Router(config-if)#no shutdown
  Router(config-if)#exit
  Router(config)#ipv6 unicast-routing
  Router(config)#int fa0/1
  Router(config-if)#ipv6 enable
  Router(config-if)#ipv6 address 2001::10/64
  Router(config-if)#no shutdown

  %LINK-5-CHANGED: Interface FastEthernet0/1, changed state to up
  Router(config-if)#
  %LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, changed state to up

  Router(config-if)#exit
  Router(config)#ipv6 unicast-routing
  Router(config)#ipv6 route 2000:2::/64 2001::20
  
```

Затем откройте вкладку **CLI** для **R1** и настройте на нем маршрутизацию.

R2(config)#ipv6 unicast-routing

R2(config)#ipv6 route 2000:1::/64 2001::10

```
Router1
Physical Config CLI
IOS Command Line Interface

Router#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#ipv6 unicast-routing
Router(config)#int fa0/1
Router(config-if)#ipv6 enable
Router(config-if)#ipv6 address 2001:20::64
% Incomplete command.
Router(config-if)#ipv6 address 2001:20::64
Router(config-if)#no shutdown

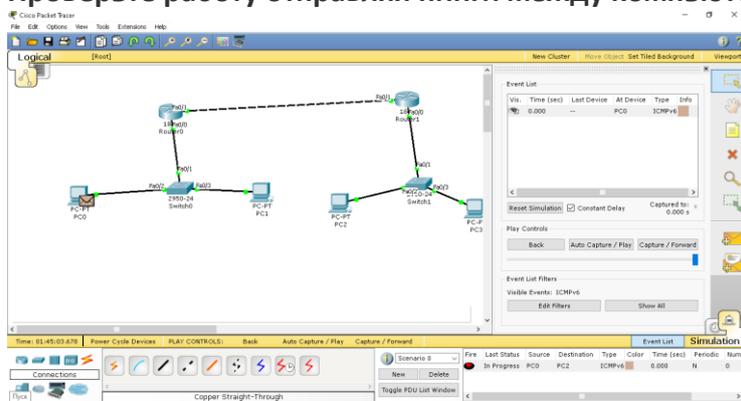
%LINK-5-CHANGED: Interface FastEthernet0/1, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, changed state to up

Router(config-if)#int fa0/0
Router(config-if)#ipv6 enable
Router(config-if)#ipv6 address 2000:2::1/64
Router(config-if)#no shutdown

%LINK-5-CHANGED: Interface FastEthernet0/0, changed state to up
%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/0, changed state to up

Router(config-if)#exit
Router(config)#ipv6 unicast-routing
Router(config)#ipv6 route 2000:1::/64 2001::10
Router(config)#
```

Проверьте работу отправляя пинги между компьютерами разных подсетей



Используйте команду `tracert` на ПК, чтобы увидеть путь, по которому проходит пакет.

Отправьте от ПК0 к ПК3

Например

```
PC>tracert 2000:2::3
```

3. Списки доступа ACL

Списки доступа (access-lists) используются в целом ряде случаев и являются механизмом задания условий, которые роутер проверяет перед выполнением каких-либо действий. Маршрутизатор проверяет каждый пакет и на основании вышеперечисленных критериев, указанных в ACL определяет, что нужно сделать с пакетом, пропустить или отбросить. Типичными критериями являются адреса отправителя и получателя пакета, тип протокола. Каждый критерий в списке доступа записывается отдельной строкой. Список доступа в целом представляет собой набор строк с критериями, имеющих один и тот же номер (или имя). Порядок задания критериев в списке существенен. Проверка пакета на соответствие списку производится последовательным применением критериев из данного списка (в том порядке, в котором они были введены). Пакет, который не соответствует ни одному из введенных критериев будет отвергнут. Для каждого протокола на интерфейс может быть назначен только один список доступа.

Без ACL - по умолчанию при создании конечной точки ей все разрешено.

Разрешить - при добавлении одного или нескольких диапазонов "разрешения" все остальные диапазоны по умолчанию запрещаются. Только пакеты из разрешенного диапазона IP-адресов смогут достичь конечной точки виртуальной машины.

Запретить - при добавлении одного или нескольких диапазонов "запретить" все другие диапазоны трафика по умолчанию разрешаются.

Сочетание разрешения и запрета - можно использовать сочетание правил "разрешить" и "запретить", чтобы указать вложенный разрешенный или запрещенный диапазон IP-адресов.

Рассмотрим два примера стандартных списков:

access-list 1 permit host 10.0.0.10 - разрешаем прохождение трафика от узла 10.0.0.10.

access-list 2 deny 10.0.1.0 0.0.0.255 - запрещаем прохождение пакетов из подсети 10.0.1.0/24.

Создание стандартного списка доступа

Списки доступа бывают нескольких видов: стандартные, расширенные, динамические и другие. В стандартных ACL есть возможность задать только IP адрес источника пакетов для их запретов или разрешений.

На [рисунке 1](#) показаны две подсети: 192.168.0.0 и 10.0.0.0.

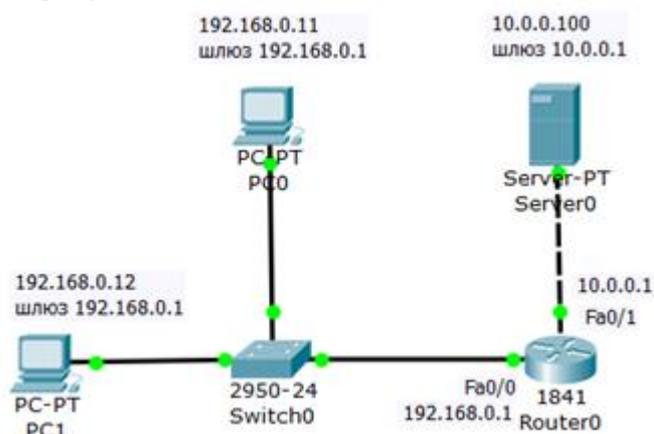


Рис. 1. Схема сети

Постановка задачи

Требуется разрешить доступ на сервер PC1 с адресом 192.168.0.12, а PC0 с адресом 192.168.0.11 – запретить ([рис. 2](#)).



Рис. 2. Постановка задачи

Соберем данную схему и настроим ее. Настройку PC0 и PC1 выполните самостоятельно.

Настройка R0 (если вы не настроили через интерфейс)

Интерфейс 0/0 маршрутизатора 1841 настроим на адрес 192.168.0.1 и включим следующими командами:

```
Router>en
Router#conf t
Router (config)#int fa0/0
Router (config-if)#ip addr 192.168.0.1 255.255.255.0
Router (config-if)#no shut
Router (config-if)#exit
```

Второй интерфейс маршрутизатора (порт 0/1) настроим на адресом 10.0.0.1 и так же включим:

```
Router (config)#intfa0/1
Router (config-if)#ip addr 10.0.0.1 255.255.255.0
Router (config-if)#no shut
```

Настройка сервера

Настройки сервера приведены на [рис. 3](#).

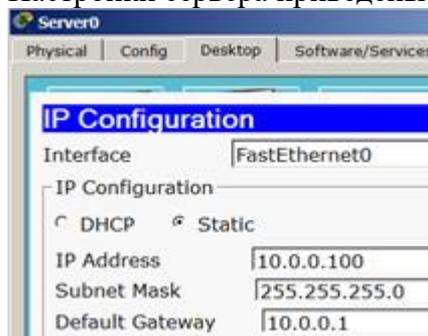


Рис. 3. Конфигурирование S0

Диагностика сети

Проверяем связь ПК из разных сетей ([рис. 4](#)).

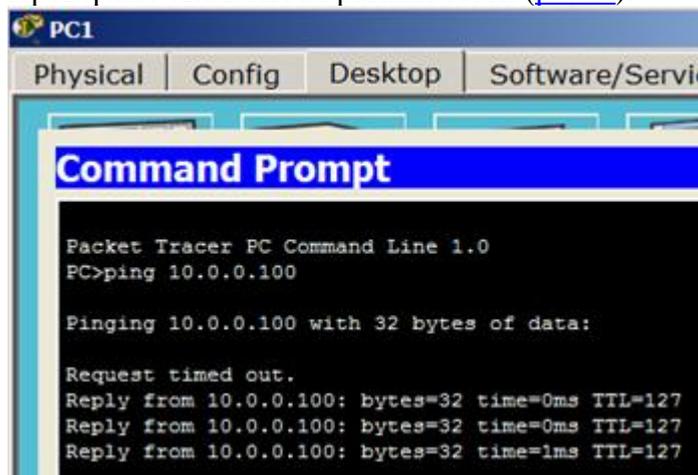


Рис. 4. ПК из разных сетей могут общаться

Приступаем к решению задачи

Правило запрета и разрешения доступа будем составлять с использованием стандартных списков доступа (ACL). Пока не задан список доступа на интерфейсе всё разрешено (**permit**). Но, стоит создать список, сразу действует механизм "Всё, что не разрешено, то запрещено". Поэтому нет необходимости что-то запрещать (**deny**) –

указываем что разрешено, а "остальным – запретить" подразумевается автоматически. По условиям задачи нам нужно на R0 пропустить пакеты с узла 192.168.0.12 на сервер ([рис. 5](#)).

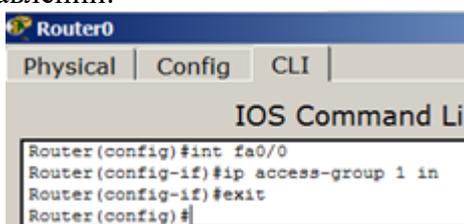


```
Router0
Physical | Config | CLI |
IOS Command Line Interface
Router#
Router#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#access-list 1 permit host 192.168.0.12
Router(config)#exit
```

Рис. 5. Создаем на R0 разрешающий ACL

Замечание: набор команд в режиме конфигурирования маршрутизатора.

Применяется данное правило на интерфейс в зависимости от направления (PC1 расположен со стороны порта Fa0/0) – [рис. 6](#). Эта настройка означает, что список доступа (правило с номером 1) будет действовать на интерфейсе fa0/0 на входящем (in) от PC1 направлении.



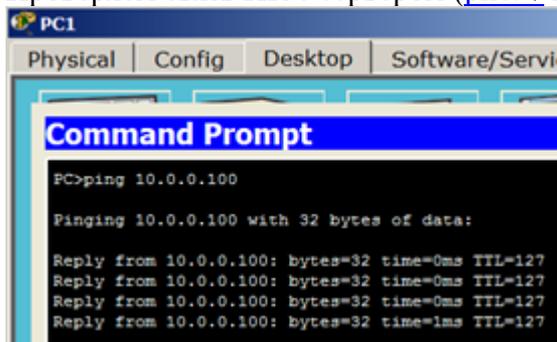
```
Router0
Physical | Config | CLI |
IOS Command Li
Router(config)#int fa0/0
Router(config-if)#ip access-group 1 in
Router(config-if)#exit
Router(config)#
```

Рис. 6. Применяем правило к порту Fa0/0

Примечание

Входящий трафик (in) — этот тот, который приходит на интерфейс извне. Исходящий (out) — тот, который отправляется с интерфейса вовне. Список доступа вы можете применить либо на входящий трафик, тогда неудобные пакеты не будут даже попадать на маршрутизатор и соответственно, дальше в сеть, либо на исходящий, тогда пакеты приходят на маршрутизатор, обрабатываются им, доходят до целевого интерфейса и только на нём обрабатываются. Как правило, списки применяют на входящий трафик (in).

Проверяем связь ПК с сервером ([рис. 7](#) и [рис. 8](#)).



```
PC1
Physical | Config | Desktop | Software/Servi
Command Prompt
PC>ping 10.0.0.100
Pinging 10.0.0.100 with 32 bytes of data:
Reply from 10.0.0.100: bytes=32 time=0ms TTL=127
Reply from 10.0.0.100: bytes=32 time=0ms TTL=127
Reply from 10.0.0.100: bytes=32 time=0ms TTL=127
Reply from 10.0.0.100: bytes=32 time=1ms TTL=127
```

Рис. 7. Для PC1 сервер доступен

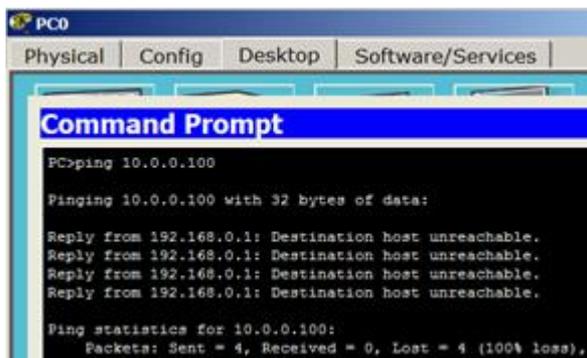


Рис. 8. Для PC0 сервер не доступен
Давайте посмотрим ACL ([рис. 9](#)).

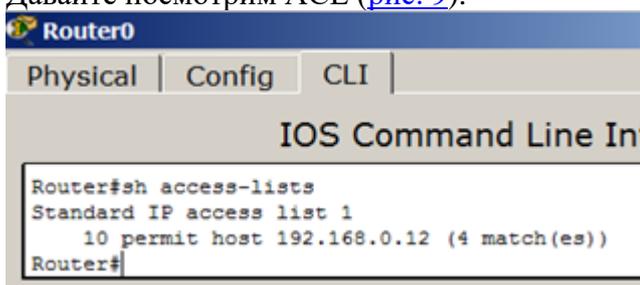
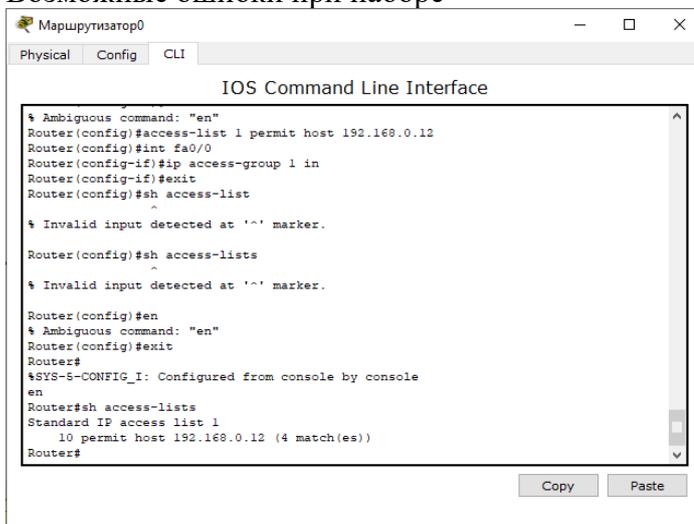


Рис. 9.9. Узел 192.168.0.12 разрешен
Возможные ошибки при наборе



Команда набирается в режиме основной работы с маршрутизатором (не конфигурирование). Выйдите из конфигурирования: exit. Ввод команд – en.

Задание.

Нужно добавить новый узел, например, PC2 с адресом 192.168.0.13 в раздел "разрешённых". Пишем команду **Router (config)#access-list 1 permit host 192.168.0.13**. Теперь адреса 192.168.0.12 и 192.168.0.13 могут общаться с сервером, в 192.168.0.11 – нет. А для отмены какого-либо правила – повторяем его с приставкой "no". Тогда это правило исключается из конфигурации. Например (см. рис. 6), если выполнить команду **Router (config-if)#no ip access-group 1 in**, то ACL будет отменен и снова все ПК могут пинговать сервер.

Расширенные списки доступа ACL

Стандартные права не так гибки, как хотелось бы. В отличие от стандартных списков, расширенные списки фильтруют трафик более "тонко". При создании расширенных списков в правилах доступа можно включать фильтрацию трафика по протоколам и портам. Для указания портов в правиле доступа указываются следующие обозначения ():

Таблица. Обозначение портов в ACL

| обозначение | действие |
|-------------|-----------------------------------|
| lt n | Все номера портов, меньшие n. |
| gt n | Все номера портов, большие n. |
| eq n | Порт n |
| neq n | Все порты, за исключением n. |
| range n m | Все порты от n до m включительно. |

Расширенные списки доступа ACL

Соберите схему сети, показанную на [рис. 10](#).

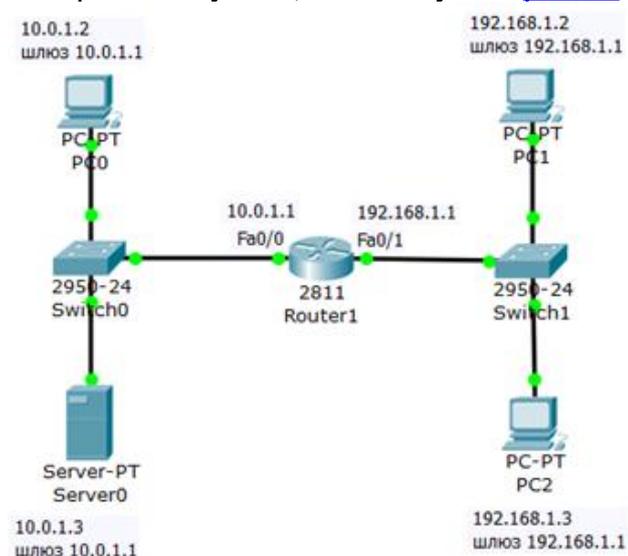


Рис. 10. Схема сети

Задача: разрешить доступ к FTP (пересылка файлов) серверу 10.0.1.3 для узла 192.168.1.2 и запретить для узла 192.168.1.3.

Создаем расширенные списки доступа и запрещаем FTP трафик

Постановка задачи графически изображена на [рис. 11](#).

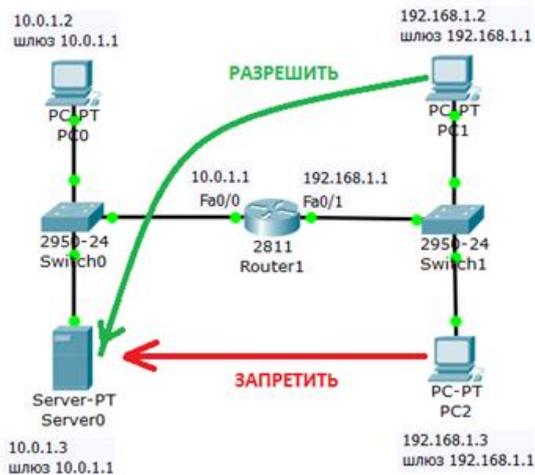


Рис. 9.11. Стрелками показана цель нашей работы

Изначально на сервере 10.0.1.3 FTP сервис поднят по умолчанию со значениями имя пользователя Cisco, пароль Cisco. Убедимся, что узел S0 доступен и FTP работает, для этого заходим на PC1 и связываемся с сервером ([рис. 12](#)). Выполняем какие-либо команды, например, DIR – чтение директории.

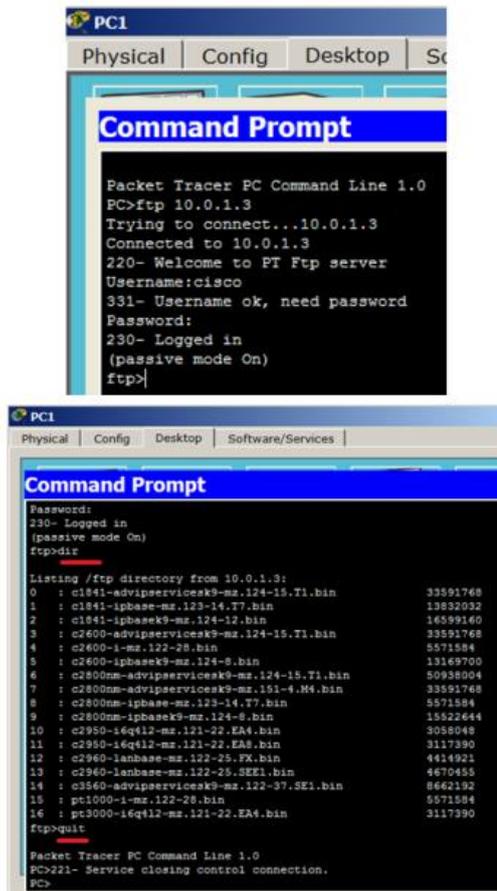


Рис. 9.12. FTP сервер доступен

Примечание. При наборе пароля на экране ничего не отображается.

Теперь создадим список правил с номером 101 в котором укажем 2 разрешающих и по 2 запрещающих правила для портов сервера 21 и 20 (Эти порты служат для FTP - передачи команд и данных) – [рис. 13](#).

```
Router>en
Router#conf t
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#ip access-list extended 101
Router(config-ext-nacl)#permit tcp 192.168.1.2 0.0.0.0 10.0.1.3 0.0.0.0 eq 21
Router(config-ext-nacl)#permit tcp 192.168.1.2 0.0.0.0 10.0.1.3 0.0.0.0 eq 20
Router(config-ext-nacl)#deny tcp 192.168.1.3 0.0.0.0 10.0.1.3 0.0.0.0 eq 21
Router(config-ext-nacl)#deny tcp 192.168.1.3 0.0.0.0 10.0.1.3 0.0.0.0 eq 20
Router(config-ext-nacl)#deny tcp 192.168.1.3 0.0.0.0 10.0.1.3 0.0.0.0 eq 20
Router(config-ext-nacl)#
```

Рис. 13. Составляем расширенные списки доступа

Совет

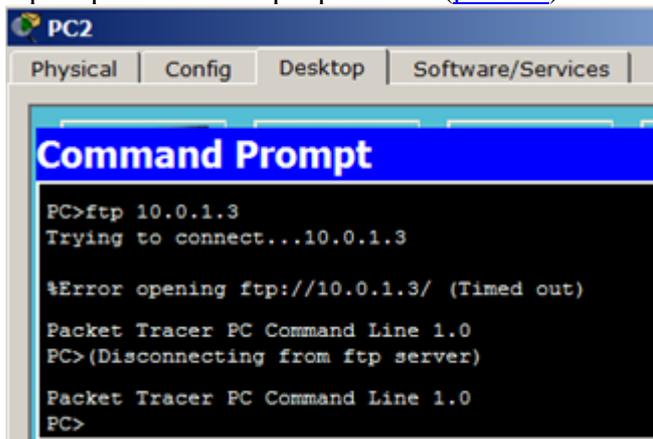
Набирайте команды аккуратно и внимательно: даже один лишний пробел может привести к ошибке при выполнении команды.

А теперь применяем наш список с номером 101 на вход (in) Fa0/1 потому, что трафик входит на этот порт роутера со стороны сети 192.168.1.0 (рис. 14).

```
Router(config-ext-nacl)#int fa0/1
Router(config-if)#ip access-group 101 in
Router(config-if)#exit
Router(config)#exit
Router#
%SYS-5-CONFIG_I: Configured from console by console

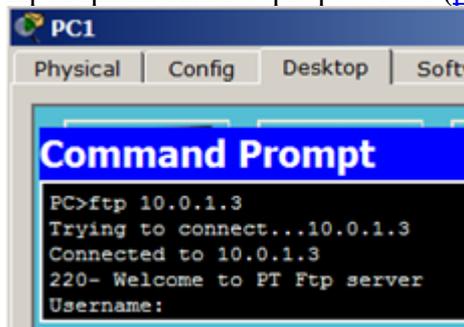
Router#wr mem
Building configuration...
[OK]
Router#
```

Рис. 14. Применяем правило с номером 101 к порту 0/1 роутера
Проверяем связь сервера с PC2 (рис. 15).



```
PC2
Physical | Config | Desktop | Software/Services
Command Prompt
PC>ftp 10.0.1.3
Trying to connect...10.0.1.3
%Error opening ftp://10.0.1.3/ (Timed out)
Packet Tracer PC Command Line 1.0
PC>(Disconnecting from ftp server)
Packet Tracer PC Command Line 1.0
PC>
```

Рис. 15. Для PC2 FTP сервер не доступен
Проверяем связь сервера с PC1 (рис. 9.16).



```
PC1
Physical | Config | Desktop | Soft
Command Prompt
PC>ftp 10.0.1.3
Trying to connect...10.0.1.3
Connected to 10.0.1.3
220- Welcome to PT Ftp server
Username:
```

Рис. 16. Для PC1 FTP сервер доступен

Автоматизированное тестирование

Методические указания.

Максимальное время выполнения 30 минут.

Критерии оценивания.

Оценка от 0 до 10 баллов.

Тестовые задания

Тест по теме «Компьютерные сети»

Вариант 1

Вопрос 1. Единая система компьютерных сетей, объединенных на базе общего протокола TCP/IP.

Ответ 1. Интранет

Ответ 2. WWW

Ответ 3. Интернет

Ответ 4. Локальная сеть

Вопрос 2. К классификации сетей «По территориальной распространенности» не относится:

Ответ 1. Региональная

Ответ 2. Глобальная

Ответ 3. Городская

Ответ 4. Локальная

Вопрос 3. Максимальная скорость передачи информации в компьютерной локальной сети (на основе кабеля витая пара) может достигать:

Ответ 1. 100 Кбайт/с

Ответ 2. 56,6 Кбит/с

Ответ 3. 100 Мбит/с

Ответ 4. 100 Кбит/с

Вопрос 4. Что НЕ нужно для подключения к кабельному Интернету?

Ответ 1. Модем

Ответ 2. Кабель

Ответ 3. Сетевой адаптер

Ответ 4. Провайдер

Вопрос 5. Классификация сетей «По типу среды передачи» включает:

Ответ 1. Кабельные

Ответ 2. По радиоканалам

Ответ 3. В инфракрасном диапазоне

Ответ 4. Все ответы верны

Вопрос 6. Общая схема соединения компьютеров в локальные сети называется...

Ответ 1. Типология

Ответ 2. Топология

Ответ 3. Схема

Ответ 4. Нет правильного ответа

Вопрос 7. Какая из приведенных схем соединения компьютеров основывается на технологии «клиент-сервер»?

Ответ 1. Шина

Ответ 2. Кольцо

Ответ 3. Звезда

Ответ 4. Нет правильного ответа

Вопрос 8. Какая из приведенных схем соединения компьютеров представляет собой последовательное соединение?

Ответ 1. Шина

Ответ 2. Кольцо

Ответ 3. Звезда

Ответ 4. Нет правильного ответа

Вопрос 9. Какой кабель обеспечивает высокоскоростную передачу данных?

Ответ 1. Коаксиальный

Ответ 2. Витая пара

Ответ 3. Оптоволокно

Ответ 4. Нет правильного ответа

Вопрос 10. Скорость передачи данных равна 600 Мбит/мин. Это составляет ... Мбит/с

Ответ 1. 100

Ответ 2. 10

Ответ 3. 600

Ответ 4. 3600

Вопрос 11. Для открытия web-страницы используется протокол...

Ответ 1. POP3

Ответ 2. HTTP

Ответ 3. SMTP

Ответ 4. FTP

Вопрос 12. С помощью какого протокола можно обеспечить защиту сообщений от прочтения или подделки при их передаче по каналам Интернета?

Ответ 1. SMTP

Ответ 2. SSL

Ответ 3. HTTP

Ответ 4. FTP

Вопрос 13. Что является именем сервера, где размещен почтовый ящик в адресе Sv-Ks@mail.ru:

Ответ 1. @

Ответ 2. .ru

Ответ 3. Sv-Ks

Ответ 4. mail.ru

Вопрос 14. Выберите корректный адрес электронной почты:

Ответ 1. ivan_petrov@mail.ru

Ответ 2. ivan petrov@mail.ru

Ответ 3. ivan_petrov@mail

Ответ 4. ivanpetrov.mail.ru

Вопрос 15. Задан адрес электронной почты в сети Интернет: user_name@mtu-net.ru. Каково имя домена верхнего уровня?

Ответ 1. user_name

Ответ 2. ru

Ответ 3. mtu-net.ru

Ответ 4. mtu-net

Вопрос 16. Компьютер, подключенный к сети Internet, обязательно имеет

Ответ 1. WEB-страницу

Ответ 2. URL-адрес

Ответ 3. доменное имя

Ответ 4. IP-адрес

Вопрос 17. Обращение к какому серверу позволяет почтовым серверам определять имена и IP-адреса друг друга?

Ответ 1. К контроллеру домена

Ответ 2. К DNS-серверу

Ответ 3. К WINS-серверу

Ответ 4. К FTP-серверу

Вопрос 18. Какой домен верхнего уровня означает «военная организация»?

Ответ 1. com

Ответ 2. gov

Ответ 3. mil

Ответ 4. edu

Вопрос 19. Определите номер компьютера в сети по IP 137.128.255.106

Ответ 1. 137

Ответ 2. 128.255.106

Ответ 3. 255.106

Ответ 4. 137.128.255.106

Вопрос 20. Модем, передающий информацию со скоростью 28 800 бит/с, за 1 секунду сможет передать:

Ответ 1. Видео файл (3,6 Мбайт)

Ответ 2. Две страницы текста (3600 байт)

Ответ 3. Аудио файл (360 Кбайт)

Ответ 4. Рисунок (36 Кбайт)

Вариант 2

Вопрос 1. Глобальная сеть – это ...

Ответ 1. Система, связанных между собой компьютеров.

Ответ 2. Система, связанных между собой локальных сетей.

Ответ 3. Система, связанных между собой локальных телекоммуникационных сетей.

Ответ 4. Система, связанных между собой локальных сетей и компьютеров отдельных пользователей.

Вопрос 2. Чтобы соединить два компьютера по телефонным линиям связи необходимо иметь:

Ответ 1. Модем

Ответ 2. Два модема

Ответ 3. Телефон, модем, специальное программное обеспечение

Ответ 4. По модему на каждом компьютере и специальное программное обеспечение

Вопрос 3. E-mail – это:

Ответ 1. Поисковая программа

Ответ 2. Название почтового сервера

Ответ 3. Почтовая программа

Ответ 4. Обмен письмами в компьютерных сетях (электронная почта)

Вопрос 4. Протокол HTTP служит для:

Ответ 1. Передачи гипертекста

Ответ 2. Передачи файлов

Ответ 3. Управления передачи сообщениями

Ответ 4. Запуска программы с удаленного компьютера

Вопрос 5. Какие компоненты вычислительной сети необходимы для организации одноранговой локальной сети?

Ответ 1. Модем, компьютер-сервер

Ответ 2. Сетевая плата, сетевое программное обеспечение

Ответ 3. Компьютер-сервер, рабочие станции

Ответ 4. Линии связи, сетевая плата, сетевое программное обеспечение

Вопрос 6. Для просмотра WEB-страниц предназначены:

Ответ 1. Поисковые серверы

Ответ 2. Браузеры

Ответ 3. Телеконференции

Ответ 4. Провайдеры

Вопрос 7. Какая из приведенных схем соединения компьютеров представляет собой замкнутую цепочку?

Ответ 1. Шина

Ответ 2. Кольцо

Ответ 3. Звезда

Ответ 4. Нет правильного ответа

Вопрос 8. Какой кабель обеспечивает скорость передачи данных до 10 Мбит/с?

Ответ 1. Коаксиальный

Ответ 2. Витая пара

Ответ 3. Оптоволокно

Ответ 4. Нет правильного ответа

Вопрос 9. Для передачи файлов по сети используется протокол...

Ответ 1. POP3

Ответ 2. HTTP

Ответ 3. SMTP

Ответ 4. FTP

Вопрос 10. Выберите корректный адрес электронной почты:

Ответ 1. ivanpetrov@mail

Ответ 2. ivan_petrov.mail.ru

Ответ 3. ivan petrov.mail.ru

Ответ 4. ivan_petrov@mail.ru

Вопрос 11. Скорость передачи данных равна 6000 Мбит/мин. Это составляет ... Мбит/с.

Ответ 1. 10

Ответ 2. 100

Ответ 3. 3600

Ответ 4. 36000

Вопрос 12. Задан адрес электронной почты в сети Интернет: fortuna@list.ru. Каково имя почтового сервера?

Ответ 1. fortuna@list.ru

Ответ 2. fortuna

Ответ 3. list.ru

Ответ 4. list

Вопрос 13. Компьютер, подключенный к сети Internet, обязательно имеет ...

Ответ 1. URL-адрес;

Ответ 2. IP-адрес

Ответ 3. WEB-страницу

Ответ 4. Доменное имя

Вопрос 14. Выберите корректный IP-адрес компьютера в сети

Ответ 1. 108.214.198.112

Ответ 2. 18.274.198.0

Ответ 3. 1278.214.198

Ответ 4. 10,0,0,1225

Вопрос 15. Топология компьютерной сети, в которой все компьютеры сети присоединены к центральному узлу, называется ...

Ответ 1. Шина

Ответ 2. Кольцо

Ответ 3. Звезда

Ответ 4. Нет правильного ответа

Вопрос 16. Какой домен верхнего уровня означает «образовательный сайт»?

Ответ 1. mil

Ответ 2. gov

Ответ 3. com

Ответ 4. edu

Вопрос 17. Определите домен 2 уровня www.klyaksa.inform.net.

Ответ 1. klyaksa

Ответ 2. inform

Ответ 3. www

Ответ 4. net

Вопрос 18. Определите номер компьютера в сети по IP 215.128.255.106.

Ответ 1. 215.128.255.106

Ответ 2. 128.255.106

Ответ 3. 255.106

Ответ 4. 106

Вопрос 19. Протокол – это ...

Ответ 1. Способность компьютера посылать файлы через каналы передачи информации.

Ответ 2. Устройство для работы локальной сети.

Ответ 3. Стандарт передачи данных через компьютерную сеть.

Ответ 4. Стандарт отправки сообщений через электронную почту.

Вопрос 20. Сколько времени будет проходить передача файла размером 128 Кбайт по сети, скорость которой составляет 128 Кбит/с?

Ответ 1. 8 с.

Ответ 2. 1 с.

Ответ 3. 1 мин.

Ответ 4. 10 с.

Другие виды учебной деятельности

Выполнение контрольной работы – от 0 до 10 баллов.

Задания для контрольных работ

Контрольная работа № 1 «Протоколы компьютерных сетей» (от 0 до 5 баллов)

Вариант 1

1. Определите класс сети в котором определён узел
128.10.0.11
221.17.16.67
2. Найдите: номер сети, маску, номер узла, количество узлов в сети, широковещательный адрес:
178.20.64.14 /20
3. Разделите на 4 подсети
200.14.96.0 /23
4. Определите адрес кольцевого элемента в IPv4
5. Упростите: 2001:0DB8:AA10:0001:0000:0000:0000:00FB
6. Определите таблицу маршрутизации узла 192.169.11.3 /24
подключенного к роутеру 192.169.11.4 /24 (видящего дополнительную сеть 192.169.12.0 /24)

Вариант 2

1. Определите класс сети в котором определён узел

- 17.11.123.30
150.1.6.167
2. Найдите: номер сети, маску, номер узла, количество узлов в сети, широковещательный адрес:
204.25.225.15 /19
 3. Разделите на 3 подсети
200.14.96.0 /25
 4. Определите адрес кольцевого элемента в IPv6
 5. Упростите: FF02:0000:0000:0000:0001:FF00:0300
 6. Определите таблицу маршрутизации узла 192.161.15.3 /21 подключенного к роутеру 192.161.15.3 /24 (имеющий выход в Интернет)

Вариант 3

1. Определите класс сети в котором определён узел
 - а. 170.15.13.37
 - б. 200.17.90.200
2. Найдите: номер сети, маску, номер узла, количество узлов в сети, широковещательный адрес:
77.250.241.15 /21
3. Разделите на 5 подсетей
200.14.128.0 /18
4. Определите количество разрядов в адресе IPv4 и IPv5
5. Упростите: FF80:0000:0000:0000:0123:1234:ABCD:EF12
6. Определите таблицу маршрутизации узла 192.17.12.6 /25 подключенного к роутеру 192.17.12.3 /25 (имеющий выход в Интернет)

Контрольная работа № 2 по теме «Организация и проведение поиска информации в сети Интернет. Электронная почта» (от 0 до 5 баллов)

Вариант № 1.

1. Найдите ответ на поставленные вопросы. Подготовьте в Word-е файл отчета, куда будете вносить ссылки и адреса найденных Вами сведений и файлов (отчет заполняется по ходу работы). Указывайте, какие ключевые слова Вы вводили (см. табл.). Подготовьте папки, куда будете складывать найденную Вами информацию. Например, папку Отчет по поиску.

- Сколько работ Карла Брюллова представлено на выставке в Третьяковской галерее? Найдите биографию и репродукции первых его работ.
- Аббревиатура названия какой уфимской организации совпадает с названием столицы Башкортостана?

- Какому племени принадлежал город Помпеи до его покорения римлянами?
- Что являлось основными ингредиентами кока-колы в 1886 году?
- Какой хоккеист девять раз становился чемпионом мира, дважды – олимпийским чемпионом, но ни разу не был чемпионом СССР?
- Кто был министром иностранных дел в России с 1648 по 1653 гг.?
- Как называется специальный ящик для хранения сигар?
- Кто придумал способ попадания Санта Клауса в дом через дымоход?
- Сколько американских солдат участвовали в боях на Гренаде в 1983 году?
- Назовите издателя и разработчика игры «Братья пилоты».
- Какую должность занимал человек, объявивший предсмертную волю Сальвадора Дали относительно места его захоронения?
- Какой женский портрет написал Пикассо в 1905 году? Найдите репродукцию этой картины и биографию художника.
- Каких четырёх композиторов приводит в пример Чайковский, утверждая, что Моцарт сочетает в себе их лучшие качества?
- Как расшифровывается V.S.O.P на коньячной этикетке?
- Найдите место и дату рождения математика Н.И.Лобачевского.
- Когда и где состоялись первые зарубежные гастроли группы Beatles?
- Как звали монарха, правившего Таиландом в 1917 году?
- Откуда пришло выражение «Козёл отпущения», и что оно означает?
- Назовите птицу, являющуюся рекорсменом среди птиц по размерам откладываемых яиц. Найдите ее изображение.
- Фамилия архитектора, построившего кафедральный собор в городе, где родилась теннисистка Елена Лиховцева?
- Когда и кем был совершен первый воздушный таран в Великой Отечественной войне?
- В каком году И. К. Айвазовский получил официальную должность художника?
- За какие две картины Айвазовский получил звание художника? Найдите репродукции картин и биографию художника.
- В честь какого писателя в 1998 году на Чукотке была учреждена прижизненная литературная премия?
- Когда появились первые города-побратимы? Как они назывались?
- Что означает слово «Lego» (название известной компании по производству игрушек)?
- Назовите клички родителей жеребца, в 60-х годах три раза завоевывавшего «Приз Европы».

- В какой башне содержался во время французского плена человек, заказавший Леонардо да Винчи одно из его изображений Мадонны? Назовите имя этого человека.
- В каком году первое место на конкурсе Евровидение поделили четыре участника?
- Соотношение между какими двумя физическими величинами устанавливалось в диссертации Пьера Кюри?
- Каково имя первой жены 29-го султана Брунея?
- В каком году Сахаров стал лауреатом нобелевской премии мира?
- Какой храм начал строиться в 1832 году, а был освящён только в 1883?

| Тема | Адрес | Автор, или кому принадлежат права | Краткое описание | С помощью, какой системы нашли, Запрос |
|-----------------------|---|---|---|--|
| Купание Красного коня | Энциклопедия 100top.ru http://www.100top.ru/encyclopedia/ | Информационное наполнение издательство «Вече» | История создания, судьба картины | |
| | Лошади http://loshadi.ru/cgi-bin/maine.cgi?page=author&link=00104N:OO11VR | | Марка с репродукцией картины на сайте о лошадях | |

2. Отчет по поиску пришлите по электронной почте преподавателю.

Вариант № 2.

1. Найдите ответ на поставленные вопросы. Подготовьте в Word-файл отчета, куда будете вносить ссылки и адреса найденных Вами сведений и файлов (отчет заполняется по ходу работы). Указывайте, какие ключевые слова Вы вводили (см. табл.). Подготовьте папки, куда будете складывать найденную Вами информацию. Например, папку Отчет по поиску.

- В каком году была написана картина Айвазовского «Море. Коктебельская бухта»? Где она хранится в настоящее время? Найдите репродукцию этой картины и биографию художника.
- Какой король правил Францией во время похода Жанны Д'Арк на Орлеан?
- Как в программировании называют ошибку, которая исчезает или меняет свои свойства при попытке её обнаружения?

- Как называется современное корейское письмо? Когда и кем оно было введено в употребление?
- Сколько литров воды вмещается в мешочек под клювом пеликана?
- Где располагалось правительство Японии в конце XII века?
- Что в переводе на русский язык означает выражение, которым эскимосы называют остров Гренландия?
- В каком году и какую школу окончила Алла Пугачева?
- Как звали автора сценария и автора музыки одного из фильмов Ф. Копполы, которые впоследствии стали супругами?
- Назовите родину колорадского жука.
- Назовите животное, обитающее в Андах и своим внешним видом напоминающее кроликов с длинным, как у белок, хвостом. Найдите изображение этого животного.
- Когда (месяц, год) Ван Гог создал единственную картину, проданную им при жизни? Найдите репродукцию этой картины и биографию художника.
- Назовите отца и сына, вместе получивших Нобелевскую премию. В каком году?
- Какую длину по ватерлинии (с точностью до миллиметра) имел парусник, на котором было совершено первое одиночное плавание вокруг света?
- В каком году Алла Пугачева получила Гран-при фестиваля «Золотой Орфей» с песней «Арлекино»?
- Какая мелодия стала основой для Гимна СССР (1943-1991), и нынешнего государственного Гимна России (с 2001)?
- Что означает термин «энтропия» с точки зрения теории информатики?
- Назовите настоящую фамилию Кира Булычева?
- От названия какого бельгийского города происходит русское выражение «малиновый звон»? Чем знаменит этот город?
- Назовите две реки, на берегах которых произошло сражение, проигранное Юрием Долгоруким своему племяннику.
- Какой почтовый адрес у Белого Дома США?
- Какие игрушки лучше всего продавались в сети магазинов «Toys R Us» в 1990 году?
- Какова фамилия художника, картины которого были украдены из галереи Челябинска летом 2001 года? Найдите название этих картин и их репродукции. Найдите биографию художника.
- В каком году Чарлз Лютвидж Доджсон (Доджсон) стал Льюисом Кэрролом?
- Что происходит с памятником Козе в г. Урюпинске каждый вечер?
- Автором каких строений в Санкт-Петербурге является Александр Павлович Брюллов?

- Сколько времени составляет полный оборот самой медленной стрелки в мире на одном из часов Зиммера?
- Как называются самые большие съедобные плоды, произрастающие на деревьях? Какова их длина и вес и где они произрастают?
- Как расшифровывается название крупной автомобильной компании, открывшей свою первую фабрику в 1900 году?
- Какой художник оформлял самую глубокую станцию метро в Москве?
- Где можно совершить самую длинную поездку на трамвае в мире? Какова протяженность маршрута?
- Почему Александр Бэлл, изобретатель телефона, никогда не говорил по нему со своей матерью?
- Какие животные помогают рыбакам Западного Ириана плести сети?

| Тема | Адрес | Автор, или кому принадлежат права | Краткое описание | С помощью, какой системы нашли, Запрос |
|-----------------------|---|---|---|--|
| Купание Красного коня | Энциклопедия 100top.ru http://www.100top.ru/encyclopedia/ | Информационное наполнение издательство «Вече» | История создания, судьба картины | |
| | Лошади http://loshadi.ru/cgi-bin/maine.cgi?page=author&link=00104N:OO11VR | | Марка с репродукцией картины на сайте о лошадях | |

3. Отчет по поиску пришлите по электронной почте преподавателю.

Промежуточная аттестация

Методические указания.

На экзамене студент отвечает на один вопрос. При необходимости преподаватель задает дополнительные вопросы.

Критерии оценивания.

Ответ оценивается от 0 до 30 баллов.

Вопросы для проведения промежуточной аттестации (экзамен)

1. Охарактеризовать основы построения и функционирования локальных вычислительных сетей.
2. Основные принципы и приемы работы с сетевыми операционными системами.

3. Архитектура локальных сетей.
4. Основы построения глобальных вычислительных сетей.
5. Классификацию компьютерных сетей.
6. История создания и развития Интернет.
7. Основные протоколы передачи информации в локальных сетях.
8. Основные протоколы передачи информации в глобальных сетях.
9. Основные принципы поиска информации в Интернет.
10. Принципы работы с гипертекстовыми ресурсами Интернет.
11. Основные протоколы передачи информации в Интернет.
12. Основы работы с электронной почтой.
13. Передача данных с помощью электронной почты.
14. Основные команды языка HTML.
15. Принципы написания WWW страниц на языке HTML.
16. Написание программных продуктов на языке JavaScript.
17. Основные команды языка JavaScript.
18. Основы технологии мультимедиа.
19. Применение средств мультимедиа в учебном процессе.
20. Мультимедийные ресурсы Интернет.
21. Работа с протоколом передачи данных FTP и удаленного доступа Telnet.

7. Данные для учета успеваемости студентов в БАРС

Таблица 1.1 Таблица максимальных баллов по видам учебной деятельности.

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|---------|--------|----------------------|----------------------|------------------------|---------------------------------|----------------------------------|--------------------------|-------|
| Семестр | Лекции | Лабораторные занятия | Практические занятия | Самостоятельная работа | Автоматизированное тестирование | Другие виды учебной деятельности | Промежуточная аттестация | Итого |
| 3 | 10 | 0 | 20 | 20 | 10 | 10 | 30 | 100 |

Программа оценивания учебной деятельности студента 3 семестр

Лекции: посещаемость, активность; за один семестр – от 0 до 10 баллов.

Лабораторные занятия: Не предусмотрены.

Практические занятия: Контроль выполнения практических заданий в течение одного семестра – от 0 до 20 баллов.

Самостоятельная работа: Контроль выполнения заданий для самостоятельной работы, рефератов в течение семестра – от 0 до 20 баллов.

Автоматизированное тестирование: максимально можно набрать 10 баллов. Автоматизированное тестирование осуществляется системой

автоматически и баллы заносятся автоматически в соответствующую колонку таблицы после прохождения студентом on-line теста.

Другие виды учебной деятельности: Выполнение контрольных работ – от 0 до 10 баллов.

Промежуточная аттестация:

При определении разброса баллов при аттестации преподаватель может воспользоваться следующим примером ранжирования:

- 25-30 баллов – ответ на «отлично»
- 19-24 балла – ответ на «хорошо»
- 11-18 баллов – ответ на «удовлетворительно»
- 0-10 баллов – неудовлетворительный ответ.

Таким образом, максимально возможная сумма баллов за все виды учебной деятельности студента за третий семестр по дисциплине «Компьютерные сети и интернет технологии» составляет 100 баллов.

Таблица пересчета полученной студентом суммы баллов по дисциплине «Компьютерные сети и интернет технологии» в оценку (экзамен):

| | |
|---------------|------------------------|
| 90-100 баллов | «отлично» |
| 76-89 баллов | «хорошо» |
| 61-75 баллов | «удовлетворительно» |
| 0-60 баллов | «не удовлетворительно» |

8. Учебно-методическое и информационное обеспечение дисциплины «Компьютерные сети и интернет технологии».

а) литература:

1. Компьютерные сети [Электронный ресурс] : учебник / В.Г. Карташевский [и др.]. — Электрон. текстовые данные. — Самара: Поволжский государственный университет телекоммуникаций и информатики, 2016. — 267 с. — 2227-8397. — Режим доступа: <http://www.iprbookshop.ru/71846.html>
2. Лиманова Н.И. Архитектура вычислительных систем и компьютерных сетей [Электронный ресурс] : учебное пособие / Н.И. Лиманова. — Электрон. текстовые данные. — Самара: Поволжский государственный университет телекоммуникаций и информатики, 2017. — 197 с. — 2227-8397. — Режим доступа: <http://www.iprbookshop.ru/75368.html>
3. *Оливер Ибе* Компьютерные сети и службы удаленного доступа [Электронный ресурс] : учебное пособие / Ибе Оливер. — Электрон.

текстовые данные. — Саратов: Профобразование, 2017. — 333 с. — 978-5-4488-0054-2. — Режим доступа: <http://www.iprbookshop.ru/63577.html>

4. *Проскуряков, А. В.* Компьютерные сети. Основы построения компьютерных сетей и телекоммуникаций : учебное пособие / А. В. Проскуряков. — Ростов-на-Дону, Таганрог : Издательство Южного федерального университета, 2018. — 201 с. — ISBN 978-5-9275-2792-2. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <https://www.iprbookshop.ru/87719.html> (дата обращения: 15.12.2021). — Режим доступа: для авторизир. пользователей

б) программное обеспечение и Интернет-ресурсы

1. Библиотека научной и студенческой информации.
<http://www.bibliofond.ru/>
2. Методическая копилка учителя информатики.
<http://www.metod-kopilka.ru/>
3. Видео уроки в сети интернет.
<http://www.videouroki.net/>

Лицензионное программное обеспечение:

Office Professional Plus 2007 (44107825)

Бесплатное программное обеспечение

Cisco Packet Tracer: <https://www.netacad.com/ru/courses/packet-tracer>

Xlight - Windows FTP and SFTP Server:

<https://www.xlightftpd.com/download.htm>

Visual Studio Community 2017: <https://msdn.microsoft.com/ru-ru/dn878009.aspx>

9. Материально-техническое обеспечение дисциплины «Компьютерные сети и интернет технологии»

Для проведения практических занятий требуются компьютерные классы с программным обеспечением (Cisco Packet Tracer, Microsoft Office), рассчитанные на обучение группы студентов из 10–15 человек, удовлетворяющие санитарно-гигиеническим требованиям, работающие под управлением операционной системы Windows с подключением к Internet.

Для проведения групповых лекционных занятий необходим проектор, подключенный к компьютеру, и экран. Требования к программному обеспечению:

- Операционная система Windows;
- Microsoft Office Power Point.
- Cisco Packet Tracer

Реализация практической подготовки в рамках учебных занятий запланирована на базе кафедры информационных систем и технологий в обучении.

Программа составлена в соответствии с требованиями ФГОС ВО с учетом Примерной ООП ВО по направлению 44.03.01 – Педагогическое образование и профилю подготовки «Информатика».

Автор

к. п. н., доцент

В.А. Векслер

Программа одобрена на заседании кафедры информационных систем и технологий в обучении от 31 августа 2021 года, протокол № 1.