Федеральное государственное автономное
образовательное учреждение
высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт педагогики, психологии и социологии

Кафедра информационных технологий обучения и непрерывного образования

УТВЕРЖДАЮ
Заведующий кафедрой
_____ О. Г. Смолянинова
подпись
« _____» _____ 2018г.

БАКАЛАВРСКАЯ РАБОТА

44.03.01 Педагогическое образование

Разработка моделирующей программы для изучения темы «Локальные компьютерные сети»

Руководитель _____ доцент, канд.пед.наук Е. В. Киргизова
подпись, дата

Выпускник _____ К. В. Александров
подпись, дата

Красноярск 2018

РЕФЕРАТ

Выпускная квалификационная работа по теме «Разработка моделирующей программы для изучения темы «Локальные компьютерные сети»» содержит 68 страниц текстового документа, 19 иллюстраций, 1 таблицу, 41 использованный источник.

ИНФОРМАТИКА, ЛОКАЛЬНАЯ СЕТЬ, КОМПЬЮТЕРНАЯ СЕТЬ, ИНФОРМАЦИЯ, ОБУЧЕНИЕ, ПРОГРАММА, МОДЕЛЬ, ТОПОЛОГИЯ.

Объектом исследования является процесс изучения темы «Локальные компьютерные сети» на уроках информатики.

Предметом исследования являются особенности применения моделирующей программы при изучении темы «Локальные компьютерные сети» на уроках информатики.

Цель исследования: разработать моделирующую программу в среде Delphi для изучения темы «Локальные компьютерные сети» в школьном курсе информатики.

Для достижения цели были поставлены следующие задачи:

1. Проанализировать учебную и методическую литературу по теме исследования, описать её теоретические аспекты.

2. Провести анализ школьного курса информатики и раскрыть современное состояние и методические аспекты изучения темы «Локальные компьютерные сети» в курсе информатики.

3. Разработать моделирующую программу в среде Delphi для изучения темы «Локальные компьютерные сети» на уроках информатики.

4. Разработать конспект урока информатики по теме «Топология локальных компьютерных сетей» для демонстрации использования возможностей моделирующей программы.

При выполнении работы использованы следующие методы: анализа, синтеза, сравнения, формализации и классификации.

В результате проведения исследования выявлено, что тема «Локальные компьютерные сети» является одной из ключевых в развитии информационных технологий, а школьные учебники информатики

располагают малым количеством информации по этой теме.

Практическая значимость исследования состоит в разработке моделирующей программы для изучения темы «Локальные компьютерные сети» и разработке урока информатики, который должен продемонстрировать актуальность использования данной программы для организации образовательного процесса.

СОДЕРЖАНИЕ

Введение..... 4

1 Теоретические и методические аспекты изучения темы «Локальные компьютерные сети» в курсе информатики..... 6

1.1 Локальные компьютерные сети на современном этапе развития..... 6

1.2 Особенности изучения темы «Локальные компьютерные сети» в школьном курсе «Информатика и ИКТ»..... 15

2 Разработка моделирующей программы для изучения темы «Локальные компьютерные сети»..... 22

2.1 Этапы разработки моделирующей программы для изучения темы «Локальные компьютерные сети» 22

2.2 Методическое планирование урока по теме «Топология локальных компьютерных сетей» с демонстрацией использования возможностей моделирующей программы..... 50

Заключение..... 58

Список использованных источников..... 59

ВВЕДЕНИЕ

С распространением электронно-вычислительных машин нетрудно предсказать рост в потребности передачи данных. На сегодняшний день в мире существует более 130 миллионов компьютеров и более 80% из них объединены в различные информационно-вычислительные сети от малых локальных сетей в офисах до глобальных сетей типа Internet. Всемирная тенденция к объединению компьютеров в сети обусловлена рядом важных причин, таких как ускорение передачи сообщений, безопасность информации, обмен информацией между компьютерами разных фирм и производителей, работающих под разным программным обеспечением.

С развитием локальных сетей появится потребность в соответствующих специалистах. Поэтому ещё со школы необходимо просвещать в данной теме подрастающее поколение для развития базовых навыков работы с вычислительными сетями, и, возможно, подготовки будущих специалистов в этой отрасли.

Однако, несмотря на всю перспективность и актуальность данной темы, в основной и старшей школе на эту тему выделяется всего 2 часа. Этого времени не хватает для изучения даже основных знаний по данной теме: виды локальных сетей, модель сети, коммуникационное оборудование сети, виды серверов, программное обеспечение, топологии сетей, достоинства и недостатки каждой из них, и многое другое. Поэтому выпускники школы не имеют представление о профессиях, связанных с компьютерными сетями, безопасности данных в сети, защитой личной информации. Необходимо организовать деятельность учащихся на уроке так, чтобы они максимально эффективно использовали время, отведенное на данную тему.

Чтобы вызвать интерес к учению, применяется больше наглядности, так как теория быстро утомляет учащихся. Для этого разрабатываются различные программы и системы, моделирующие определенные процессы. С помощью них каждый ученик сможет создать модель чего-либо,

поэкспериментировать, и, вероятно, это сможет его заинтересовать.

Но, изучив методику обучения информатики и содержание школьных учебников, можно сказать, что учебный план располагает довольно малым количеством информации по данной теме и времени, отведенного на нее.

Чтобы это компенсировать, была разработана программа для оптимального изучения темы «Локальные компьютерные сети».

Объектом исследования является процесс изучения темы «Локальные компьютерные сети» на уроках информатики.

Предметом исследования являются особенности применения моделирующей программы при изучении темы «Локальные компьютерные сети» на уроках информатики.

Цель исследования: разработать моделирующую программу в среде Delphi для изучения темы «Локальные компьютерные сети» в школьном курсе информатики.

Для достижения цели были поставлены следующие задачи:

1. Проанализировать учебную и методическую литературу по теме исследования, описать её теоретические аспекты.

2. Провести анализ школьного курса информатики и раскрыть современное состояние и методические аспекты изучения темы «Локальные компьютерные сети» в курсе информатики.

3. Разработать моделирующую программу в среде Delphi для изучения темы «Локальные компьютерные сети» на уроках информатики.

4. Разработать конспект урока информатики по теме «Топология локальных компьютерных сетей» для демонстрации использования возможностей моделирующей программы.

При выполнении исследования использованы следующие методы: анализа, синтеза, сравнения, формализации и классификации.

В первой главе в результате проведения исследования выявлено, что тема «Локальные компьютерные сети» является одной из ключевых в развитии информационных технологий, и необходимо дальнейшее развитие в

этом направлении, а также подготовке соответствующих специалистов.

Рассмотрены три учебно-методических комплекта по информатике разных авторов, в результате чего выяснено, что школьные учебники информатики располагают малым количеством информации по этой теме, в которой не отражаются основные разделы темы «Локальные компьютерные сети».

Практическая значимость исследования состоит в разработке моделирующей программы для изучения темы «Локальные компьютерные сети» и разработке урока информатики, который должен продемонстрировать актуальность использования данной программы для организации образовательного процесса.

Выпускная квалификационная работа состоит из двух глав, заключения, списка использованных источников и содержит 64 страницы.

1 СОВРЕМЕННОЕ СОСТОЯНИЕ ИЗУЧЕНИЯ ТЕМЫ «ЛОКАЛЬНЫЕ КОМПЬЮТЕРНЫЕ СЕТИ» В КУРСЕ ИНФОРМАТИКИ

1.1 Локальные компьютерные сети на современном этапе развития

Эволюция компьютеров привела к тому, что большая комплексная машина стала необходимостью в каждом доме и офисе [22].

При работе на персональном компьютере без использования компьютерных сетей пользователи могут обмениваться информацией только путём копирования её на внешние носители информации. Но такой способ передачи информации не всегда может использоваться и может занимать достаточно долгое время.

Формирование компьютерных сетей связано с потребностью общего использования данных пользователями, работающими на удаленных друг от друга компьютерах. Сети дают возможность быстрого обмена информацией, совместного использования сетевых устройств, таких как принтеры, прочие периферийные устройства и даже одновременно работать с документами.

С целью объединения находящихся на небольшом расстоянии компьютеров используют ЛВС – локальную вычислительную сеть. Такие сети нашли широкое применение на самых различных мероприятиях и в учебных заведениях [11].

Ключевые элементы ЛВС – это:

1. Серверы – аппаратно-программные комплексы, выполняющие функции управления распределением сетевых ресурсов общего доступа.

2. Рабочие станции – компьютеры, осуществившие доступ к сетевым ресурсам, предоставляемым сервером.

3. Физическая среда передачи информации (сетевой кабель) – к ней относятся коаксиальные и оптоволоконные кабели, витые пары проводов, кроме того беспроводные каналы связи (инфракрасное излучение, лазеры, радиопередача).

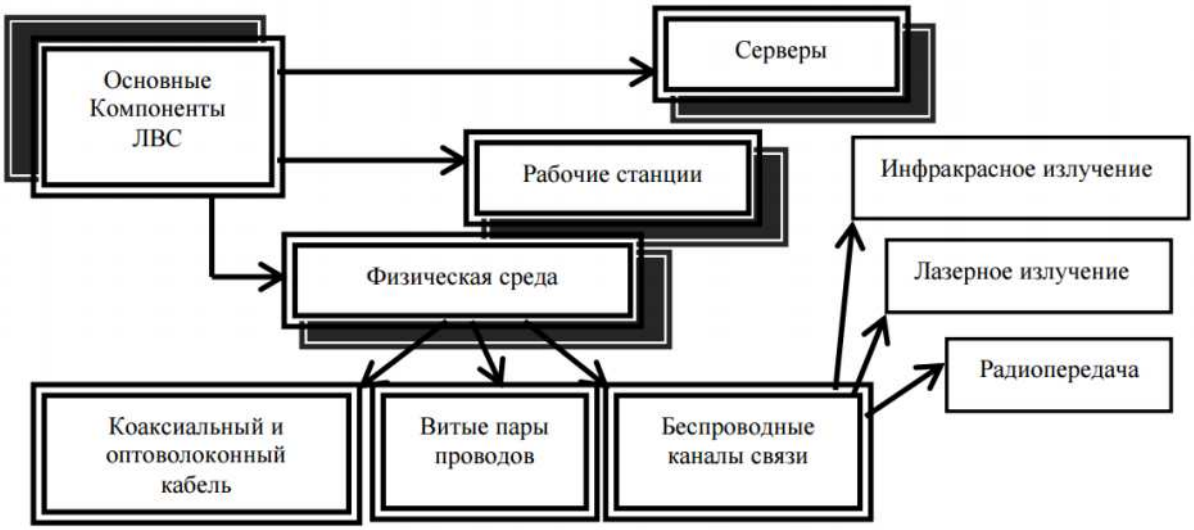


Рисунок 1 - Компоненты ЛВС

ЛВС обеспечивает:

1. Высокоскоростную многоуровневую коммутацию;
2. Контроль и разграничение доступа к сетевым ресурсам;
3. Доступ к локальным сетевым устройствам (принтеры, сканеры…);
4. Доступ к сети Интернет.

ЛВС должна быть надежной. Одно из главных преимуществ автономного ПК состоит в том, что влияние его поломки или сбоя ограничено. При объединении ПК в ЛВС система должна сохранить работоспособность при выходе из строя отдельного ПК.

На данный момент развитие сетевых технологий позволяет реализовать информационные сети самых различных конфигураций и вычислительных мощностей. Это связано с различными способами передачи информационных потоков: по витой паре, по оптоволокну, по выделенному радиоканалу.

Также это связано с большим разнообразием активного коммутационного оборудования, которое применяется для локальных и глобальных связей. В то же время практическое использование ЛВС связано с рядом трудностей. Объединение в одну сеть нескольких сетей, расположенных зачастую территориально далеко друг от друга, приводит к возрастанию стоимости создания и сложности обслуживания сети, усложняется разработка сетевых приложений. Возникает необходимость

обеспечивать совместимость программного обеспечения и требуемый уровень безопасности данных, передаваемых по сети.

Анализ источников[32,36] показывает, что под архитектурой ЛВС понимается «совокупность четырех составляющих: топология, методы доступа, программное обеспечение и комплекс технических средств». Обзор источников показывает, что сегодня наиболее часто используются три топологии локальных сетей: звезда, кольцо, дерево и шина. В результате приведенного анализа можно установить, что топология определяет требования к оборудованию, тип используемого кабеля, допустимые и наиболее удобные методы управления обменом, надежность работы, возможности расширения сети. И хотя выбирать топологию пользователю сети приходится нечасто, знать об особенностях основных топологий, их достоинствах и недостатках надо.

На различных стадиях жизненного цикла ЛВС могут использоваться различные методы оценки ее эффективности и оптимизации. В процессе проектирования ЛВС с использованием современной методологии проектирования и технологических комплексов (САПР) могут применяться экспериментальные методы исследования, аналитическое и имитационное моделирование. На стадиях опытной и рабочей эксплуатации ЛВС основным методом оценки их качества следует считать экспериментальное исследование. Онопозволяет собрать статистическую информацию о действительном ходе вычислительного, процесса, использовании оборудования, степени удовлетворения требований пользователей системы и т.п. и затем по результатам ее обработки сделать заключение о качестве проектных решений, заложенных при создании системы, а также принять решение по модернизации системы (устранению "узких" мест). Однако не исключено и использование методов моделирования, с помощью которых можно оценить эффект от модернизации ЛВС, не изменяя рабочей конфигурации и организации работы системы.

Моделирование - один из наиболее распространенных методов

исследования. Модель ЛВС - это такое ее представление, которое состоит из определенного количества организованной информации о ней и построено с целью ее изучения. Другими словами, модель - физическая или абстрактная система, представляющая объект исследования.

Рассмотрим способы соединения двух компьютеров, которые используются, как правило, для создания временных сетей компьютер-компьютер с целью совместного использования файлов. Для обмена данными между двумя компьютерами (например, настольными и переносными ПК) можно использовать различные способы их соединения:

– прямое соединение компьютеров через последовательные и параллельные порты (COM, USB, LPT, IrDA, Bluetooth);

– удаленное соединение двух компьютеров через модемы;

– соединение двух компьютеров в локальную сеть, используя сетевые карты и проводные линии связи;

– соединения двух компьютеров в локальную сеть, используя встроенные беспроводные интерфейсы Wi-Fi.

При прямом и беспроводном соединении компьютеров используются следующие порты:

Порт COM. Наиболее простой и доступный способ соединения двух компьютеров - это прямое соединение их через асинхронный последовательный порт COM. В этом случае для соединения используется только нуль-модемный кабель. Этот способ самый доступный способ подключения, так как все персональные компьютеры оснащены портами COM.

USB порт. Способ соединение двух компьютеров через USB порты. USB это универсальный интерфейс (последовательный порт) для подключения 127 устройств. Для соединения двух компьютеров требуется только кабель USBLink.

LPT - порт. Два компьютера могут быть подключены через параллельные LPT - порты. Эти порты имеют восемь разрядов. В этом

способе для соединения двух компьютеров используется только нуль-модемный кабель.

IrDA - инфракрасные порты предназначены для беспроводного подключения карманных или блокнотных компьютеров к настольному компьютеру. Связь обеспечивается при условии прямой видимости, дальность передачи данных не более 1 м. Если в компьютере нет встроенного iRDA адаптера, то он может быть выполнен в виде дополнительного внешнего устройства (USB iRDA адаптера), подключаемого через USB-порт. Этот способ предназначен для беспроводного подключения карманных или блокнотных ПК к настольному компьютеру.

Bluetooth («блютус») - высокоскоростной микроволновый стандарт, позволяющий передавать данные на расстояниях до 10 метров. Если нет встроенного Bluetooth адаптера, то он может быть выполнен в виде дополнительного внешнего устройства (USB bluetooth адаптера), подключаемого через USB-порт. Этот способ предназначен для беспроводного подключения карманных или блокнотных компьютеров к настольному компьютеру.

Беспроводная связь. Стандартом беспроводной связи для локальных сетей является технология Wi-Fi. Wi-Fi обеспечивает соединение точка-точка (соединение Ad-Hoc), которое можно применить для подключения двух ПК. При выборе этого способа для соединения двух ПК требуется два адаптера Wi-Fi.

Если говорить о локальных сетях, то в любой сети на основном слое находится компьютер. Компьютеры подключаются к сети с помощью сетевой карты.

Ко второму слою относится коммуникационное оборудование, которое играет не менее важную роль, чем компьютеры. Коммуникационное оборудование сетей можно разделить на три группы:

– сетевые адаптеры (карты);

– сетевые кабели;

– промежуточное коммуникационное оборудование (трансиверы, повторители, концентраторы, коммутаторы, мосты, маршрутизаторы и шлюзы).

Третьим слоем, образующим программную платформу сети, являются операционные системы. В зависимости от того, какие концепции управления локальными и распределенными ресурсами положены в основу сетевой ОС, зависит эффективность работу всей сети.

Четвертый слой - это сетевые приложения. К сетевым приложениям относятся такие приложения как сетевые базы данных, почтовые приложения, системы автоматизации коллективной работы и т.д.

Рассмотрим более подробно аппаратные средства сетей - компьютеры. Архитектура компьютера включает в себя как структуру, отражающую аппаратный состав ПК, так и программно – математическое обеспечение. Все компьютеры сетей можно разделить на два класса: серверы и рабочие станции.

Сервер, как уже было сказано ранее, - это многопользовательский компьютер, выделенный для обработки запросов от всех рабочих станций. Это мощный компьютер или «мэйнфрейм», предоставляющий рабочим станциям доступ к системным ресурсам и распределяющий эти ресурсы. Сервер имеет сетевую операционную систему, под управлением, которой происходит совместная работа всей сети.

Основными требованиями, которые предъявляются к серверам, являются высокая производительность и надежность их работы. Серверы в больших сетях стали специализированными и, как правило, используются для управления сетевыми базами данных, организации электронной почты, управления многопользовательскими терминалами (принтерами, сканерами, плоттерами) и т.д.

Существует несколько типов серверов:

1. Файл-серверы. Управляют доступом пользователей к файлам и программам.

2. Принт-серверы. Управляют работой системных принтеров.

3. Серверы приложений. «Серверы приложений - это работающий в сети мощный компьютер, имеющий прикладную программу, с которой могут работать клиенты. Приложения по запросам пользователей выполняются непосредственно на сервере, а на рабочую станцию передаются лишь результаты запроса».

4. Почтовые серверы. «Данный сервер используется для организации электронной корреспонденции с электронными почтовыми ящиками».

5. Прокси-сервер. Это эффективное средство подключения локальных сетей к сети Интернет. «Прокси-сервер - компьютер, постоянно подключенный к сети Интернет, через который происходит общение пользователей локальной сети с сетью Интернетом»[36].

В ближайшее время значимость Интернета как единого информационного пространства будет постепенно уменьшаться [19]. Большую часть глобальных сервисов могут сменить локальные (региональные) сервисы. «Если еще несколько лет назад пользователь, несмотря на небольшую пропускную способность каналов связи, за сеанс посещал более десяти разных веб-сайтов, то на сегодняшний день, несмотря на значительный рост пропускной способности каналов связи, значительная доля пользователей ограничивается посещением лишь одного ресурса – социальной сети. Локализации пользователей также будет содействовать внедрению государственных алфавитов в доменное пространство сети Интернет. Последующая стратификация пользователей способна стать причиной того, что через определённое время в Интернете образуются слабосвязанные друг с другом сегменты. Вследствие произойдет вырождение глобальной сети Интернет в 11 (согласно количеству введенных языков) региональных сетей» [26].

Предположим изменение результатов решаемых с помощью локальных сетей задач, которые в данный момент можно решить в сети Интернет [8]:

– информационная – обеспечивает возможность поиска и получения н

научной, общественно-политической и другой информации. В локальных сетях можно формировать (или перемещать из других мест) необходимые локальные базы данных.

– коммуникационная – дает возможность реализовывать коммуникации с помощью цифровой телефонии, передачи/приема факсов, электронных писем, форумов и социальных сетей. Размещение всех таких ресурсов в локальной сети способно существенно увеличить скорость доступа к ним и обеспечить необходимый уровень защиты данных.

Размещение в локальной сети периферийных устройств (принтеров, сканеров и т.д.) открывает возможность совместного доступа к ним. Остается возможность реализации многопользовательского режима доступа к программным средствам (компьютерные игры, корпоративные проекты).

– совещательная – обеспечивает интерактивный режим обмена данными с применением телеконференций (MSN), систем интерактивного общения и др. Достоинства применения локальных сервисов подобны тем, что у коммуникационной функции:

– коммерческая – обеспечивает возможность использования услуг электронных магазинов, Web-витрин, аукционов (просмотр и выбор, заказ товаров на компьютере покупателя, оплата с помощью электронных денег) электронной коммерции в бизнесе и пр., в.т.ч. выполняя маркетинговую функцию – оперативно доставляя рекламу до массового потребителя. Применение в локальной сети этих сервисов, безусловно, уменьшит количество потенциальных покупателей, но, тем не менее, такого рода подход гарантирует доступ к определенной целевой аудитории.

– развлекательная – обеспечивает доступ к огромному объему развлекательной литературы, фильмов, музыкальных файлов, игр и др. Локальные ресурсы способны гарантировать соблюдение авторских прав и ограничение доступа детей к информации, не предназначенной для их возрастной категории.

– образовательная – даёт возможность получать образование удаленно,

используя разнообразные формы обучения. Грамотно организованное применение локальных ресурсов никак не отразится на качестве получаемых знаний, а близость ресурса заметно улучшит качество использования мультимедийных программ в связи с большей скоростью.

– исследовательская – обеспечивает возможность проведения рекламных и социологических исследований. Исследования на локальном ресурсе способствуют сегментированию целевых групп.

– кроме того, можно отметить так называемую компьютерную функцию – возможность приобрести новейшие программные ресурсы, инструкции и рекомендации по работе. Расположение в локальной сети, к примеру, набора некоторых необходимых драйверов способно существенно облегчить работу конечных пользователей.

Появление и в дальнейшем развитие локальных вычислительных сетей - это эволюция компьютерных технологий. Границы между вычислительной и коммуникационной техникой стираются, происходит конвергенция технологий. Их проблемы и опыт в разработках становятся общими.

При использовании ЛВС появляется ряд преимуществ, а также возможность совместно использовать данные или устройства, что означает сокращение расходов на оборудование. Сотрудники получают доступ к корпоративной информации. Использование ЛВС так же позволяет снизить потребность компаний в других формах передачи информации. Уже на сегодняшний день любая современная организация имеет свою построенную локальную сеть.

Исходя из этого, в настоящее время необходимо дальнейшее развитие локальных ресурсов. Развитие локальных сетей способствует как развитию глобальной сети в целом, так и локализации конкретных сервисов, что способствует улучшению качества их функционирования, гарантирует новый качественный уровень защиты конфиденциальной информации.

1.2 Особенности изучения темы «Локальные компьютерные сети» в школе

Для анализа были рассмотрены 3 школьных учебно-методических комплекса и изучены следующие учебники: «Л. Л. Босова. Информатика: Учебник для 9 класса», «Семакин И.Г. Информатика: Учебник для 8 класса» и «Угринович Н.Д. Информатика: Учебник для 8 класса».

Наиболее узко данная тема рассматривается в учебнике Семакина И.Г., она находится в главе «Передача данных в компьютерных сетях»в параграфе «Как устроена компьютерная сеть», занимает всего один пункт в параграфе, на который отводится половина урока, и в него включено следующее:

– определение локальных сетей;

– типы локальных сетей;

– понятие сервера и рабочей станции;

– применение локальных сетей.

В учебнике Босовой Л.Л. «Локальные компьютерные сети» расположены в главе «Коммуникационные технологии» и занимают один раздел, на который тоже отводится менее одного часа. Здесь, помимо информации, даваемой в предыдущем учебнике, содержатся:

– функция сетевого адаптера;

– виды соединений компьютеров;

– типы кабелей, скорость передачи данных по каждому из них.

Данная тема рассматривается более подробно в учебнике «Угринович Н.Д. Информатика: Учебник для 8 класса». Тема «Локальные компьютерные сети» содержится в главе «Коммуникационные технологии и разработка Web-сайтов» и изучается второй по счёту, отдельно от других тем этой главы, тем не менее на её изучение отводится всего один час, и один час на практическое задание.

Изучив материалы каждого из учебников, можно сделать следующие выводы:

Цели изучения темы в базовом курсе информатики: дать учащимся представление о назначении и структуре локальных сетей. Обучить их способам обмена файлами в локальной сети компьютерного класса.

Ядро темы – определение «локальные компьютерные сети» и технология построения этих сетей.

Взаимосвязь с другими темами: данная тема связана с разделами «Передача информации» и «Глобальная компьютерная сеть Интернет». Изучение любого вида сетей, в частности локальной компьютерной сети, базируется на знаниях по теме «Передача информации», так как именно в этой теме рассматривается понятие «пропускная способность» и общая схема передачи информации. Без знания этой темы сложно полноценно воспринимать информацию по теме «Глобальная компьютерная сеть Интернет».

Если компьютеры в школьном кабинете информатики объединены в локальную сеть, то это обстоятельство значительно упрощает изучение этой темы. Именно школьный компьютерный класс должен стать исходной точкой в разговоре о передаче данных в компьютерных сетях. Определив компьютерную сеть как систему компьютеров, связанных каналами передачи данных, педагог наглядно показывает работу системы на оборудовании компьютерного класса и говорит, что такая сеть называется локальной. Локальные компьютерные сети небольшие по масштабам и расположены в пределах одного помещения, здания, предприятия. Пользователей общей локальной сети принято называть рабочей группой, а компьютеры, за которыми они работают, - рабочими станциями[15].

Различают следующую физическую топологию локальных компьютерных сетей [40]:

– Полносвязная. Данная топология громоздкая и неэффективная, требует для любого ребра графа своей электрической линии связи, для каждой вершины графа (компьютера) – множества коммуникационных портов по количеству связей. Используется реже других, например, в

многомашинных комплексах, либо глобальных сетях при малом количестве ПК. Это единственная топология, не требующая транзитной передачи информации посредством других промежуточных узлов сети;

– Ячеистая (mesh) – образуется из полносвязной посредством удаления определённых ребер. В ней напрямую соединены только компьютеры, среди которых ожидается активный обмен информацией. Обмен среди остальных компьютеров выполняется через промежуточные узлы. Ячеистая топология допускает объединение большого количества ПК и свойственна, как правило, для глобальных сетей; «Общая шина» (магистраль, шлейфовое подключение, daisy-цепь – гирлянда) – весьма популярная топология локальных сетей. Компьютеры присоединяются к одному коаксиальному кабелю согласно схеме «монтажного ИЛИ». Передаваемая информация может распространяться в обе стороны. Её плюсы: невысокая цена и легкость разводки кабеля по помещениям, наименьшая протяженность кабеля; минусы: надежность (любой дефект кабеля или одного разъема целиком отключает всю сеть), невысокая производительность (только один компьютер способен передавать информацию в сеть в одно время);

– «Звезда». Каждый компьютер подключается отдельным кабелем к общему центральному устройству – концентратору, направляющему передаваемую одним компьютером информацию дальше: другому или всем другим компьютерам сети. Основное преимущество перед общей шиной – намного более высокая надежность. Какие-либо дефекты кабеля здесь касаются только одного компьютера, вывести из строя всю сеть может лишь неисправность самого концентратора. Концентратор способен также играть роль фильтра информации, поступающей от узлов в сеть, блокируя запрещенные администратором передачи. Минусы: высокая цена сетевого оборудования, включающая стоимость концентратора, ограниченное число компьютеров по числу (8, 16, 24, …) портов концентратора;

– «Иерархическая звезда» (звездообразное дерево) – развитие топологии «звезда», когда применяется несколько концентраторов,

объединенных друг с другом иерархически (каскадами). На сегодняшний день иерархическая звезда является наиболее популярным типом топологии связей в локальных и глобальных сетях;

– «Кольцо». Компьютеры соединены последовательно и образуют замкнутую цепь – однонаправленный замкнутый канал связи. Данные передаются по кольцу, и в случае если ПК распознает их как «свои», то принимает их. Отключение какой-либо станции влечет за собой разрыв цепи, для исключения чего разработаны специальные устройства. Станция-отправитель может контролировать доставку, дожидаясь по кольцу получения признака-квитанции;

– Смешанная. Она включает отдельные произвольно связанные фрагменты (подсети) выше рассмотренных типовых топологий.

В небольших локальных сетях все компьютеры, как правило, равноправны, т.е. пользователи самостоятельно решают, какие ресурсы своего компьютера (диски, папки или подключенные принтеры) сделать доступными для других пользователей сети. После этого пользователи, работающие за другими компьютерами сети, смогут использовать ресурсы другого компьютера как своими собственными. Главным минусом таких одноранговых локальных сетей считается низкая безопасность данных от несанкционированного доступа.

В целях обеспечения большей информационной безопасности один из компьютеров локальной сети может быть выделен в качестве сервера, на котором обычно хранится наиболее важная информация. Правила доступа к этой информации определяет один человек - администратор сети [33].

Использование локальных сетей отвечает двум главным требованиям:

1) обмену файлами среди пользователей сети;

2) использованию общедоступных ресурсов: большого пространства дисковой памяти, принтеров, централизованной системы данных, программного обеспечения.

Освоение сетевых информационных технологий предполагает знание

следующих разделов информатики: [15]

– архитектура и ключевые принципы работы персонального компьютера;

– шифрование дынных и представление их в ПК;

– классификация программного обеспечения;

– роль операционной системы в работе ПК, классификация операционных систем;

– компоненты объектно-ориентированного интерфейса пользователя;

– файловая система;

– основные принципы обработки текстовой и графической информации, мультимедиа;

– информационные системы;

– основы защиты информации.

Поэтому изучением сетевых технологий обычно завершается общий курс информатики и ИКТ.

Учебный план не даёт возможности долго изучать тему локальных сетей. Педагог, в первую очередь, должен дать представление учащимся об организации сети, функционирующей в компьютерном классе, и, если имеется, общешкольной сети, именно поэтому необходимо за это время организовать работу так, чтобы учащиеся изучили данную тему с разных сторон, заинтересовались ей, и, возможно, задумались о её самостоятельном изучении. В приведенных выше учебниках единственное практическое задание по данной теме – организация общего доступа к принтеру в сети. Не всегда имеется возможность выполнить такое задание в условиях класса, а если учитывать, что на домашних и школьных компьютерах зачастую установлены разные операционные системы, отчего будет меняться алгоритм организации общего доступа, то выполнение этой практической работы не принесёт ни пользы, ни интереса для школьников. Поэтому необходимо создать что-то универсальное, интересное, понятное, имеющее набор различных заданий, эффективное в изучении данной темы. Существует

несколько программ для моделирования компьютерных сетей, таких, как NetCracker,Net-Simulator, GraphicalNetworkSimulator 3, Cisco Packet Tracer, LanFlow, далее представлен интерфейс некоторых из них:

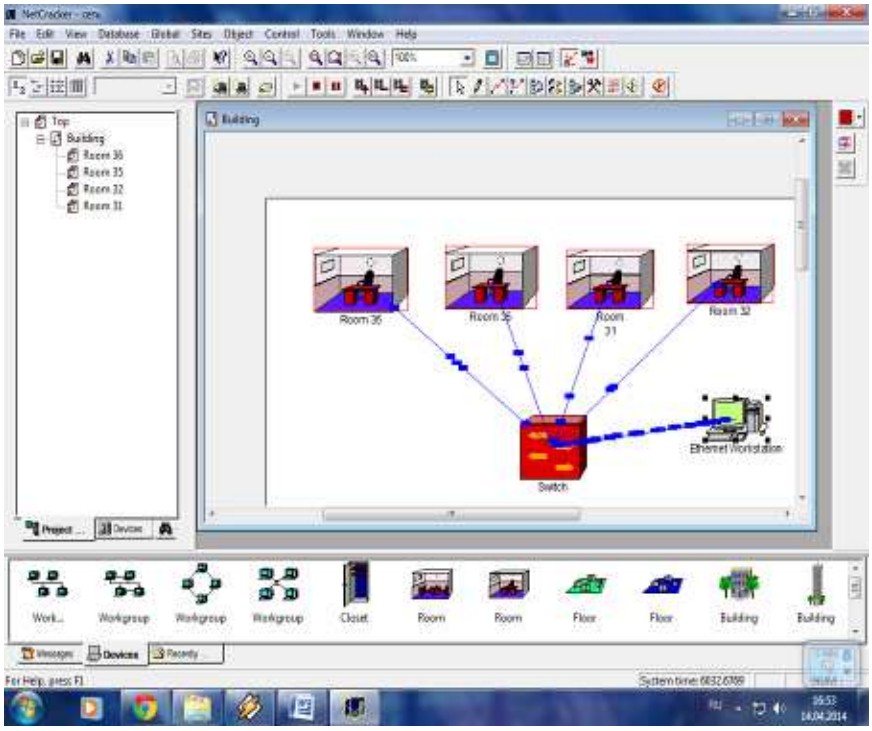


Рисунок 2 - Программа для моделирования компьютерных сетей NetCracker

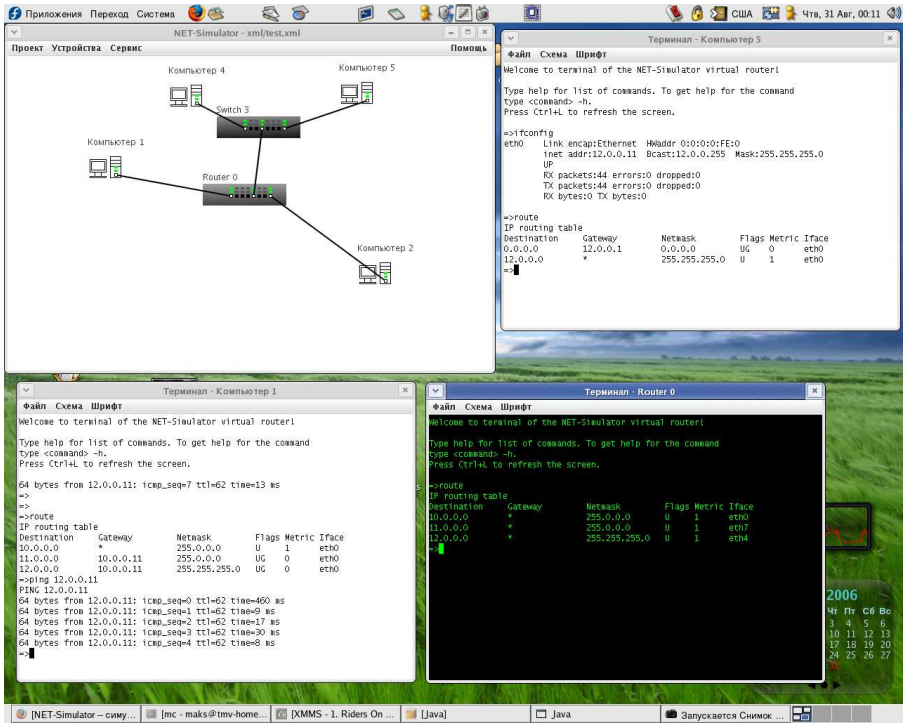


Рисунок 3 -Программа для моделирования компьютерных сетей Net-Simulator

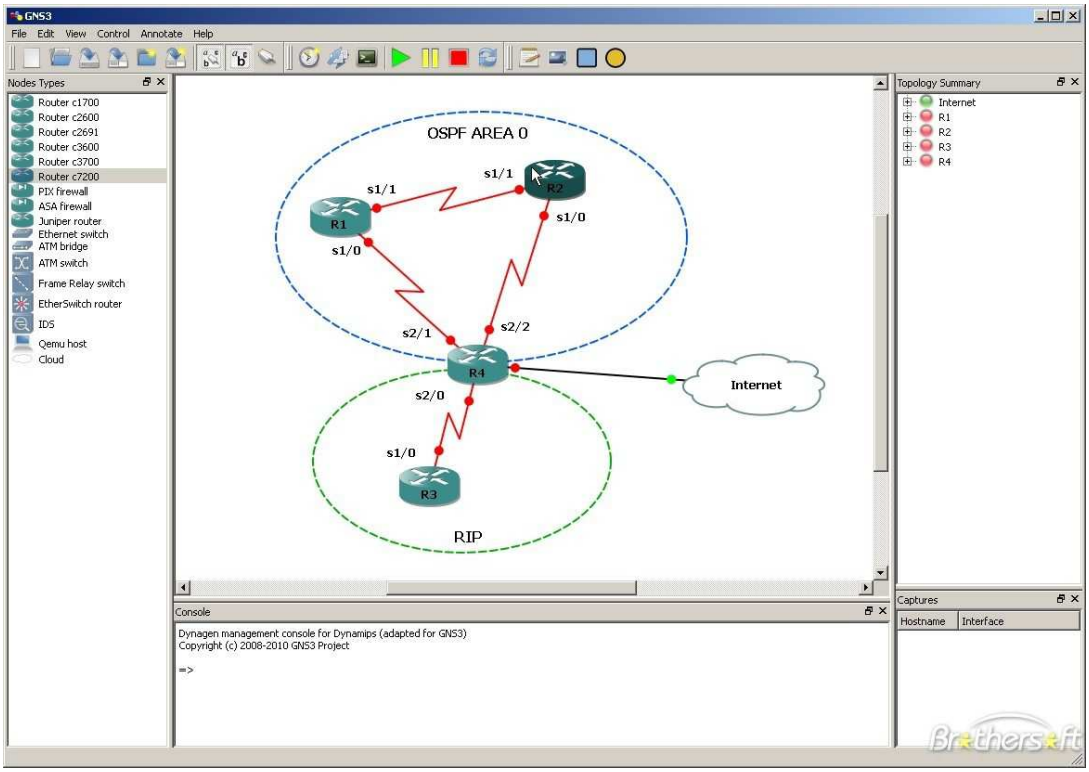


Рисунок 4 - Программа для моделирования компьютерных сетей GNS3

Главное преимущество разрабатываемой программы в том, что она является образовательной. В ней есть всё необходимое для изучения темы «Локальные компьютерные сети» в школе. В отличие от других программ, большинство из которых на английском языке, имеют сложный интерфейс, и рассчитаны на пользователей, уже знакомых с компьютерными сетями, она имеет довольно простой и удобный интерфейс, она полностью направлена на изучение темы, а не только на моделирование сетей, так как содержит теорию, в которую включены знания из школьных учебников и дополнительных источников, этап тестирования, который позволяет осуществить контроль знаний, и этап создания модели, который так же, как и вся программа, имеет удобный интерфейс. Сверху выводится конкретное задание, для каждого ученика своё, и учащийся сразу понимает что ему нужно делать.

Также программа рассчитана на быструю и легкую проверку заданий учителем после урока, так как все результаты работ с именем и фамилией учащегося, выполнившего задание, сохраняются.

2 РАЗРАБОТКА МОДЕЛИРУЮЩЕЙ ПРОГРАММЫ ДЛЯ ИЗУЧЕНИЯ ТЕМЫ «ЛОКАЛЬНЫЕ КОМПЬЮТЕРНЫЕ СЕТИ»

2.1 Этапы разработки программы

Перед началом разработки программы были поставлены задачи, которые должна выполнять данная программа:

1) расширение знаний в изучаемой теме (в данной программе представлено больше теоретического материала по теме «Локальные компьютерные сети», чем в учебно-методических комплексах);

2) повторение и закрепление знаний по данной теме (если учащийся недостаточно хорошо воспринял материал, данный учителем, он может ещё раз его закрепить, просмотрев теорию, которая дана в простом и доступном, но содержательном виде);

3) проведение учебных экспериментов с моделями изучаемых объектов;

4) контроль и оценивание уровня знаний.

Было рассмотрено множество вариантов интерфейса программы, типов соединения элементов в практическом задании, и мы пришли к выводу, что наиболее оптимальным будет использование элемента DrawGrid – таблицы изображений.

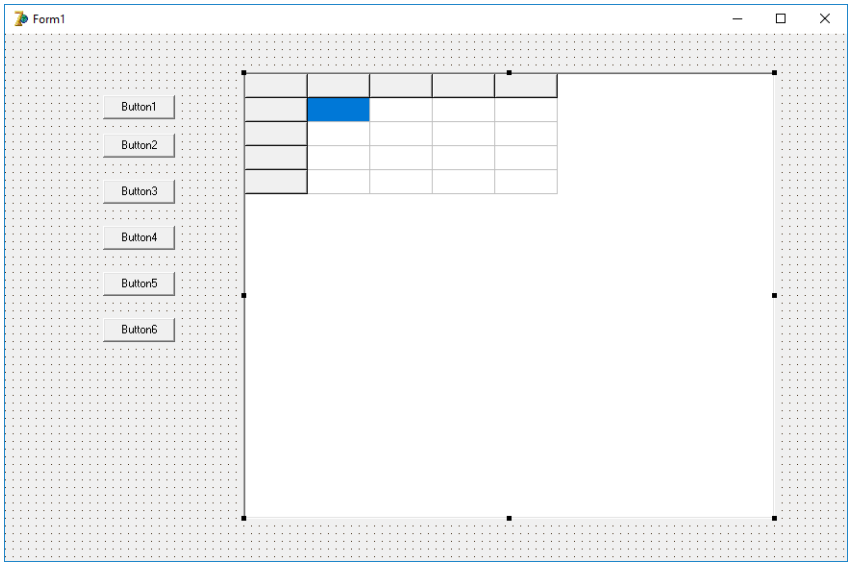


Рисунок 5 - Первый этап разработки программы

На первом этапе разработки был создан простой, интерфейс, без какой-либо связи между компонентами, добавлены 6 кнопок Button, элемент DrawGrid.

Затем началась работа над элементом DrawGrid. Сначала было необходимо сформировать равные клетки для помещения в них различных устройств. Также кнопкам было присвоено определенная надпись, для последующего перехода по ним к определенной теме.

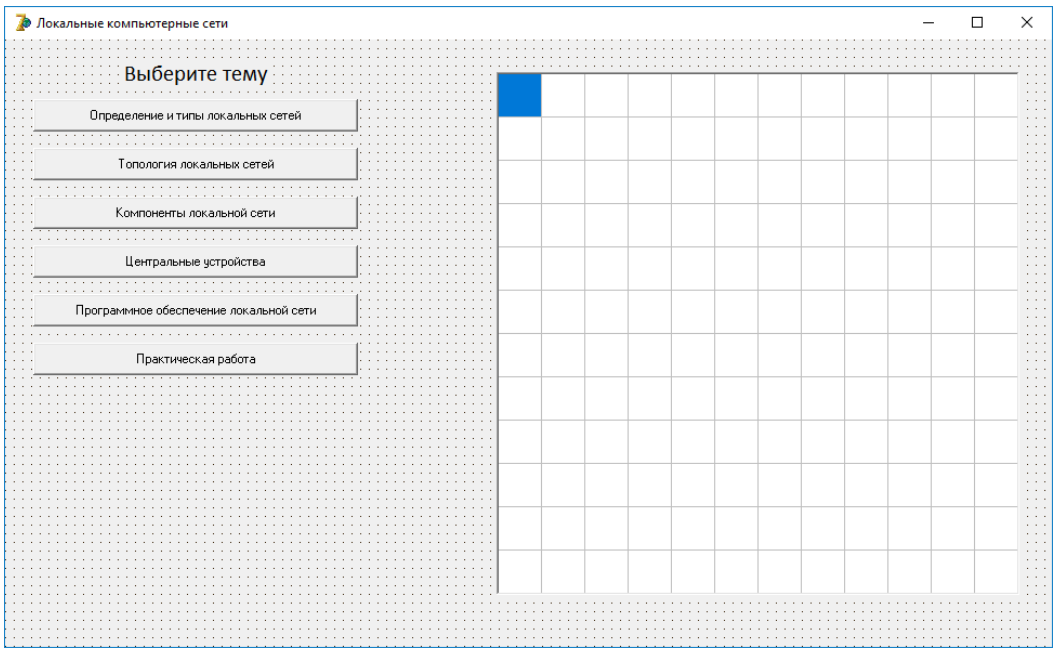


Рисунок 6 - Начальный этап создания интерфейса

Затем началась первая работа над кодом программы. Необходимо было, чтобы при нажатии какого-либо элемента Button, форма очищалась (все элементы становились невидимыми) и программа переходила к определенной теме, или к практическому заданию. Для этого была создана процедура tema, к которой программа каждый раз будет обращаться при нажатии одной из кнопок.

```
procedure TForm1.tema;  //При выборе темы скрывает остальные кнопки и надпись "Выберите тему"
begin
label1.Visible:=false;
button1.Visible:=false;
```

button2.Visible:=false;

...

button6.Visible:=false;

end;

В первую очередь началась разработка главной части программы – моделирования компьютерной сети. Необходимо было найти способ заполнения ячеек поля DrawGrid картинками. Были найдены изображения компьютера, свитча, хаба, проводов и других элементов, обрезаны и сжаты под размер элемента Image, 39 на 39 пикселей. Далее началась работа над выводом картинок в клетки поля. Так как для этого нет специальной функции, мы использовали функцию Canvas, позволяющую рисовать на компонентах формы, или самой форме.

Были использованы две процедуры - DrawGridSelectCell:

procedure TForm1.DrawGrid1SelectCell(Sender: TObject; ACol, ARow: Integer;

var CanSelect: Boolean);

begin

a:=acol;

b:=arow;

end;

иImageClick:

procedureTForm1.Image1Click(Sender: TObject); //В зависимости от нажатой картинки добавляет её в выделенную клетку

begin

*DrawGrid1.Canvas.Draw(a*40,b*40,image1.Picture.Graphic);*

mas[a,b]:=1;

end;

При нажатии на клетку поля, в памяти программы сохраняются индексы выделенной клетки, затем при нажатии на одну из картинок Imageеё изображение добавляется в нужную клетку. Также для последующих

вычислений был объявлен массив целочисленных элементов, уже не для визуального, а программного добавления элементов в поле.

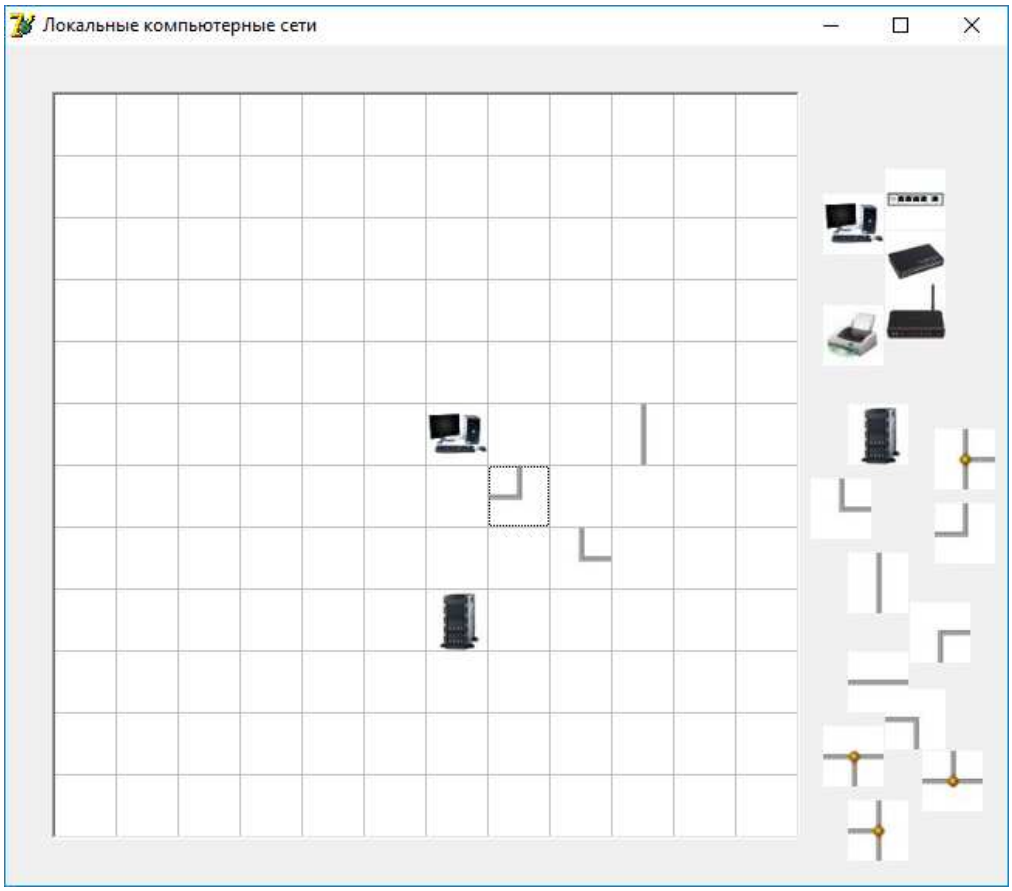


Рисунок 7 - Вывод картинок в DrawGrid

Затем необходимо было добавить возможность визуального очищения клетки и ее программного очищения. Для этого был добавлен компонент PopUpMenu, который вызывает контекстное меню. Далее была долгая работа с координатами для точного определения места клика через процедуры DrawGrid1MouseDown и N1Click, в итоге получилось следующее:

```
procedure TForm1.DrawGrid1MouseDown(Sender: TObject; Button:
TMouseButton;
  Shift: TShiftState; X, Y: Integer);
  var p:TPoint;
begin
  a:=x div 40;
  b:=y div 40;
```

```
 p:=GetClientOrigin;
 if Button = mbRight then PopupMenu1.Popup(x+p.x+drawgrid1.left,
y+p.y+drawgrid1.top);
end;
procedure TForm1.N1Click(Sender: TObject);
begin
 mas[a,b]:=0;
 DrawGrid1.canvas.Draw(a*40,b*40,image16.Picture.Graphic);
 proverka;
end;
```

Также была добавлена большая процедура proverka после каждого нажатия на компонент DrawGrid, так как изображения в ячейках пропадали после нажатия на них.

```
procedure TForm1.proverka; //Проверяет, на месте ли все изображения
после каждого нажатия на поле
begin
for i:=0 to 11 do
 for j:=0 to 11 do
  case mas[i,j] of
   1: DrawGrid1.canvas.Draw(i*40,j*40,image1.Picture.Graphic);
   2: DrawGrid1.canvas.Draw(i*40,j*40,image2.Picture.Graphic);
   3: DrawGrid1.canvas.Draw(i*40,j*40,image3.Picture.Graphic);
   4: DrawGrid1.canvas.Draw(i*40,j*40,image4.Picture.Graphic);
   5: DrawGrid1.canvas.Draw(i*40,j*40,image5.Picture.Graphic);
   6: DrawGrid1.canvas.Draw(i*40,j*40,image6.Picture.Graphic);
   7: DrawGrid1.canvas.Draw(i*40,j*40,image7.Picture.Graphic);
   8: DrawGrid1.canvas.Draw(i*40,j*40,image8.Picture.Graphic);
   9: DrawGrid1.canvas.Draw(i*40,j*40,image9.Picture.Graphic);
   10: DrawGrid1.canvas.Draw(i*40,j*40,image10.Picture.Graphic);
   11: DrawGrid1.canvas.Draw(i*40,j*40,image11.Picture.Graphic);
```

*12: DrawGrid1.canvas.Draw(i*40,j*40,image12.Picture.Graphic);*

*13: DrawGrid1.canvas.Draw(i*40,j*40,image13.Picture.Graphic);*

*14: DrawGrid1.canvas.Draw(i*40,j*40,image14.Picture.Graphic);*

*15: DrawGrid1.canvas.Draw(i*40,j*40,image15.Picture.Graphic);*

*16: DrawGrid1.canvas.Draw(i*40,j*40,image17.Picture.Graphic);*

*0:DrawGrid1.canvas.Draw(i*40,j*40,image16.Picture.Graphic);*

end;

end;

Процедура получается довольно массивной, поскольку компоненты Image не имеют индексов.

Далее в PopUpMenu был добавлен пункт «Инфо», в котором можно посмотреть описание добавленного элемента.

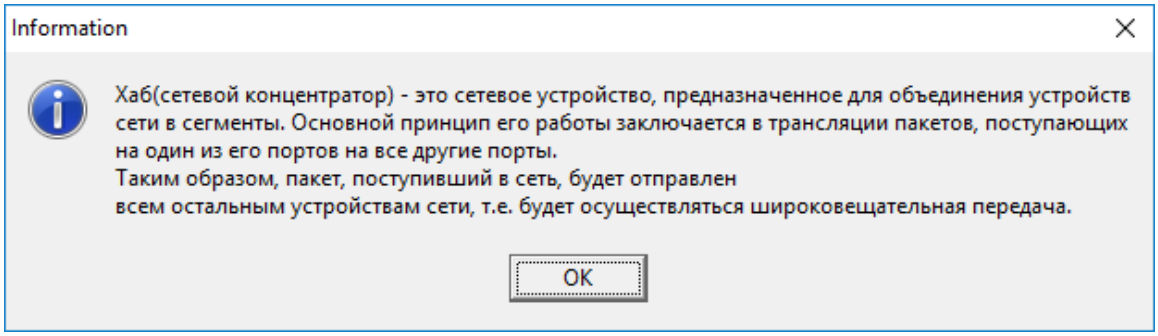


Рисунок 8 - Описание элементов

Код оказался объемным по причине длинного описания каждого элемента (приведены только начало и конец процедуры):

procedure TForm1.N2Click(Sender: TObject);

var kab:string;

begin

kab:=(«Соединение устройств и компьютеров в локальной сети осуществляется с помощью кабеля типа «витая пара».'+#13+'Обычно это четыре пары скрученных между собой изолированных проводников, помещенных в оболочку.'+#13+'Каждый проводник окрашен в свой цвет и имеет специальный номер. Витая пара бывает

```
нескольких типов.'+#13+'Чаще всего используют витую пару 5-й
категории со скоростью передачи до 100 Мбит в секунду.»);
  case mas[a,b] of
    1: MessageDlg('Компьютер – это устройство для сбора, обработки,
хранения и вывода информации.',mtInformation,[mbOK], 0);
    2: MessageDlg(«Свитч(сетевой коммутатор) - устройство,
предназначенное для соединения нескольких
узлов'+#13+'компьютерной сети в пределах одного сегмента.'+#13+'В
отличие от концентратора, который распространяет трафик от
одного подключенного'+#13+'устройства ко всем остальным,
коммутатор передает данные только непосредственно
получателю.'+#13+'Это повышает производительность и
безопасность сети, избавляя остальные сегменты'+#13+'сети от
необходимости (и возможности) обрабатывать данные, которые им
не предназначались.»,mtInformation, [mbOK], 0);
...
16: MessageDlg(«Если к локальной сети подключено более 10
  компьютеров, одноранговая сеть может
  оказаться'+#13+'недостаточно производительной.'+#13+'Для
  увеличения производительности, а также в целях обеспечения
  большей надежности хранения'+#13+'информации в сети
  выделяется мощный компьютер для хранения файлов и программных
  приложений.'+#13+'Такой компьютер называется сервером, а
  локальная сеть — сетью на основе сервера. Сеть на
  основе'+#13+'сервера имеет топологию звезда, т. е. сервер и все
  компьютеры сети подключаются к
  центральному'+#13+'концентратору или
  коммутатору.»,mtInformation, [mbOK], 0);
  end;
end;
```

Одним из пунктов PopUpMenu мы решили сделать пункт Очистить всё, для упрощения работы с программой.

```
procedure TForm1.N3Click(Sender: TObject);
begin
 for i:=0 to 11 do
  for j:=0 to 11 do
    mas[i,j]:=0;
 proverka;
end;
```

И, наконец, последняя составляющая этой части – сохранение работы для последующей проверки учителем.

Был выбран следующий способ: учащийся выполняет задание, нажимает правой кнопкой мыши на поле DrawGrid, и в контекстном меню выбирает пункт «Сохранить». Программа сохраняет скриншот всего экрана сначала в компонент Image, а затем в графический файл, находящийся в папке с программой. Благодаря этому учитель может в любое удобное время проверить все выполненные работы. Так как после нажатия кнопки «Сохранить» контекстное меню исчезает не сразу, а с задержкой, и перекрывает на скриншоте некоторые клетки, то было решено использовать компонент Timer:

```
procedure TForm1.N4Click(Sender: TObject);
begin
 timer1.enabled:=true;
end;
procedure TForm1.Timer1Timer(Sender: TObject);
var bmp:TBitmap;
begin
timer1.enabled:=false; //Запуск таймера, который по истечению 0.3 сек делает скриншот экрана
bmp := TBitmap.Create;
```

```
  bmp.Width := Screen.Width;
  bmp.Height := Screen.Height;
  BitBlt(bmp.Canvas.Handle, 0,0, Screen.Width, Screen.Height,
       GetDC(0), 0,0,SRCCOPY);
  Image32.Width := Screen.Width;
  Image32.Height := Screen.Height;
  Image32.Picture.Assign(bmp);
  Image32.Picture.SaveTofile('screenshot.bmp');
bmp.Free;
end;
```

По окончанию работы над данной частью началась работа над контролем знаний учащихся по изученной теме. Было решено выбрать метод тестирования как наиболее эффективный метод проверки знаний, умений и навыков обучающихся.

Тест включает в себя как задания закрытого, так и открытого типа. В данном тесте предлагаются 10 заданий с одним вариантом ответа, 7 заданий свободного изложения и дополнения, и 3 задания на установление соответствия. Вопросы были составлены с учетом учебной программы, взяты как из школьного учебника в главе Коммуникационные технологии и разработка Web-сайтов в разделе Локальные компьютерные сети[33], так и из дополнительного теоретического материала, который предоставлен в данной программе [9, 34, 40].

Сначала были созданы шаблоны теста для каждого типа заданий. Для вопросов с выбором одного ответа использовались поле Memo1 для вывода вопроса, 4 компонента RadioButton для выбора ответа, и кнопка Buttonдля перехода к следующему вопросу. В заданиях с вводом строки вместо компонентов RadioButton используется строка Editдля ввода ответа с клавиатуры. И в заданиях на соответствие4 компонента Labelдля вывода левой части соответствия и 4 динамичных кнопки Buttonдля правой части.

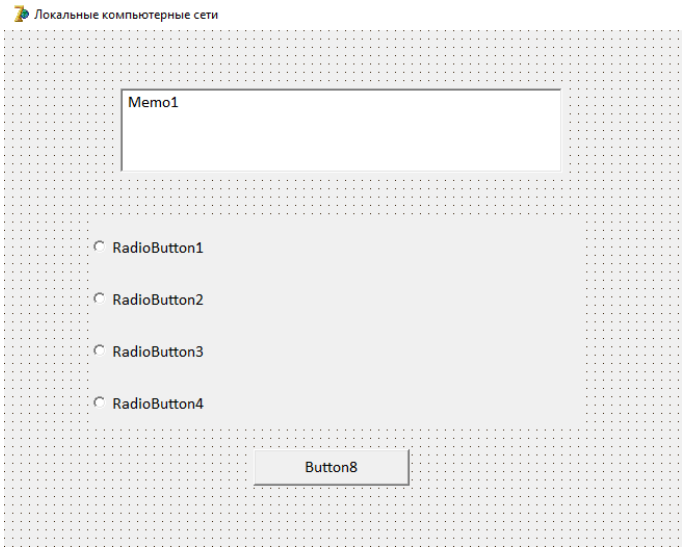


Рисунок 9 - Шаблон заданий с выбором одного ответа

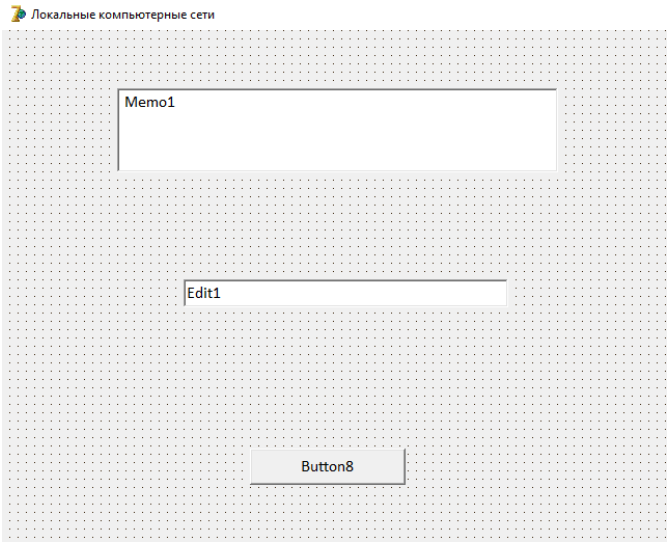


Рисунок 10 - Шаблон заданий с вводом строки

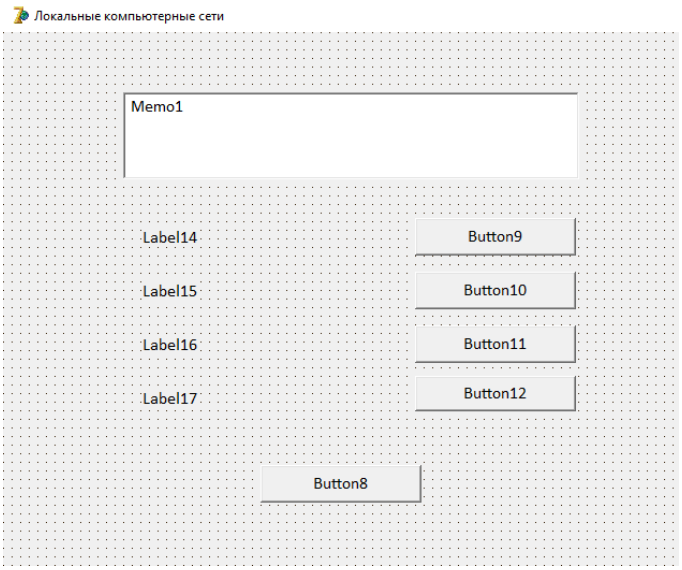


Рисунок 11 - Шаблон заданий на соответствие

Далее было необходимо создать маску, или шаблон для каждого вопроса для того, чтобы программа по индексам получала на ввод вопрос из текстового документа, искала следующий вопрос, его варианты ответа, правильный ответ и другие данные. Для вопросов с одним вариантов ответа она выглядит следующим образом:

Вопрос на выбор №1

Вариант ответа №1

Вариант ответа №2

Вариант ответа №3

Вариант ответа №4

Правильный ответ

Вопрос на выбор №2

Вариант ответа №1…

И так далее. Позднее будет описан алгоритм считывания и проверки вопросов. Второй тип – вопросы с дополнением и свободным изложением:

Вопрос с текстовым ответом №1

Правильный ответ №1

Правильный ответ №2

Правильный ответ №3

Правильный ответ №4

Правильный ответ №5

Вопрос с текстовым ответом №2…

И третий тип – установление соответствия:

Вопрос

Пункт 1)

Пункт 2)

Пункт 3)

Пункт 4)

Пункт а)

Пункт б)

Пункт в)

Пункт г)

Верная последовательность(4 цифры)

Объяснение выбранного способа записи вопросов дано в последующих абзацах.

Все вопросы теста хранятся в файле test.txtв корневом каталоге с программой. Так как отдельно считывать каждый вопрос из файла в случайном порядке не является возможным, было решено вывести текст из файла testв скрытое поле Memo2 при запуске программы. Memo позволяет вводить многострочный текст с клавиатуры, загружать его из файла, редактировать и сохранять в файл текстового формата [20]. Каждая строка поля Memoимеет свой индекс, поэтому к строкам можно легко обращаться и в любой момент сравнивать их, проверяя ответ.

Чтобы реализовать случайный порядок вопросов, была введена переменная kv для хранения количества вопросов, которые были показаны, и объявлен массив vopr[1..20] типа Integer для хранения состояния вопросов. Изначально все элементы массива имеют значение 0, затем, если учащемуся выводится определенный вопрос, и он на него отвечает, этому элементу присваивается значение 1. Чтобы все вопросы выводились в случайном порядке без сбоев, был разработан следующий алгоритм:

– программа генерирует случайное число v от 1 до 10 – вопрос на выбор одного варианта ответа, и выводит его на экран;

– когда выбран ответ и ученик хочет перейти к следующему вопросу, проверяется правильность ответа и переменной vopr[v] присваивается значение 1;

– программа генерирует следующее число v, и проверяет значение переменной vopr[v]. Если оно равно 0, т. е. вопрос еще не был показан, он выводится на экран, иначе число генерируется снова до тех пор, пока значение переменной vopr[v] не будет равным нулю;

– когда переменная kv достигает значения 10, программа начинает генерировать числа от 1 до 7 – номер вопроса на дополнение и свободного изложения;

– программа выводит по очереди 7 вопросов с вводом строки, пока переменная kv не примет значение, равное 17;

– наконец, для вывода 3 последних вопросов генерируется случайное число от 1 до 3. Когда переменная kv достигает значения 20, программа подводит итоги и высчитывает итоговую оценку по итогам тестирования.

Задания с выбором ответа. Данный вид заданий является наиболее простым для реализации. Даны 4 компонента RadioButton–варианты ответа. Маска этого вида вопросов имеет следующий вид:

*v*6+1 - Вопрос на выбор №1*

*v*6+2 - Вариант ответа №1*

*v*6+3 - Вариант ответа №2*

*v*6+4- Вариант ответа №3*

*v*6+5- Вариант ответа №4*

*v*6+6 -Правильный ответ*

Число 6 – количество строк, которое отводится на 1 вопрос. Например, если сгенерировано число 0, программа начинает счёт с нулевой строки. Следовательно, при генерации случайного числаv от 0 до 9 строка Memo2.lines[v*6+1] будет являться строкой с вопросом, Memo2.lines[v*6+2] – строкой с первым вариантом ответа, и так далее. Компонентам RadioButton1, RadioButton2, RadioButton3 и RadioButton4 присваиваются значения строк Memo2.lines[v*6+2], Memo2.lines[v*6+3], Memo2.lines[v*6+4] и Memo2.lines[v*6+5], соответственно. При выборе одного из вариантов ответа программа считывает текст, который хранит в себе элемент RadioButton, и, если он равен строкеMemo2.lines[v*6+6], то есть правильному ответу, то ответ засчитывается правильным и ученик получает за него 1 балл.

Если выполнение теста только началось, то программа работает по следующему фрагменту кода:

```
if button7.caption = 'Начать тест' then begin
  edit2.Visible:=false;
  label18.Visible:=false;
randomize();
radiobutton1.Visible:=true;
radiobutton2.Visible:=true;
radiobutton3.Visible:=true;
radiobutton4.Visible:=true;
memo1.Visible:=true;
button7.Top:=button7.Top+150;
button7.caption:='Следующий вопрос';
memo2.lines.loadfromfile('test.txt');
v:=random(10);
vopr[v]:=1;
memo1.text:=memo2.lines[v*6+1];
radiobutton1.caption:=memo2.lines[v*6+2];
 radiobutton2.caption:=memo2.lines[v*6+3];
 radiobutton3.caption:=memo2.lines[v*6+4];
 radiobutton4.caption:=memo2.lines[v*6+5];
end
```

Изначально все компоненты скрыты, их свойство Visible имеет значение False, пользователь видит только кнопку Начать тест. Затем у нужных элементов свойству Visible присваивается значение True, и необходимые компоненты (RadioButton1, RadioButton2, RadioButton3, RadioButton4, Memo1) отображаются на экране. Строка randomize() необходима для того, чтобы при каждом запуске программы числа генерировались по-новому, иначе будут одни и те же значения, и от генерации случайных чисел не будет смысла.

Если учащийся ответил на первый вопрос, программа впоследствии начинает выполнять следующий алгоритм:

```
else if (button7.caption = 'Следующий вопрос') and (kv<11) then begin
  while vopr[v]<>0 do v:=random(10);
  memo1.text:=memo2.lines[v*6+1];
  radiobutton1.caption:=memo2.lines[v*6+2];
  radiobutton2.caption:=memo2.lines[v*6+3];
  radiobutton3.caption:=memo2.lines[v*6+4];
  radiobutton4.caption:=memo2.lines[v*6+5];
vopr[v]:=1;
end
```

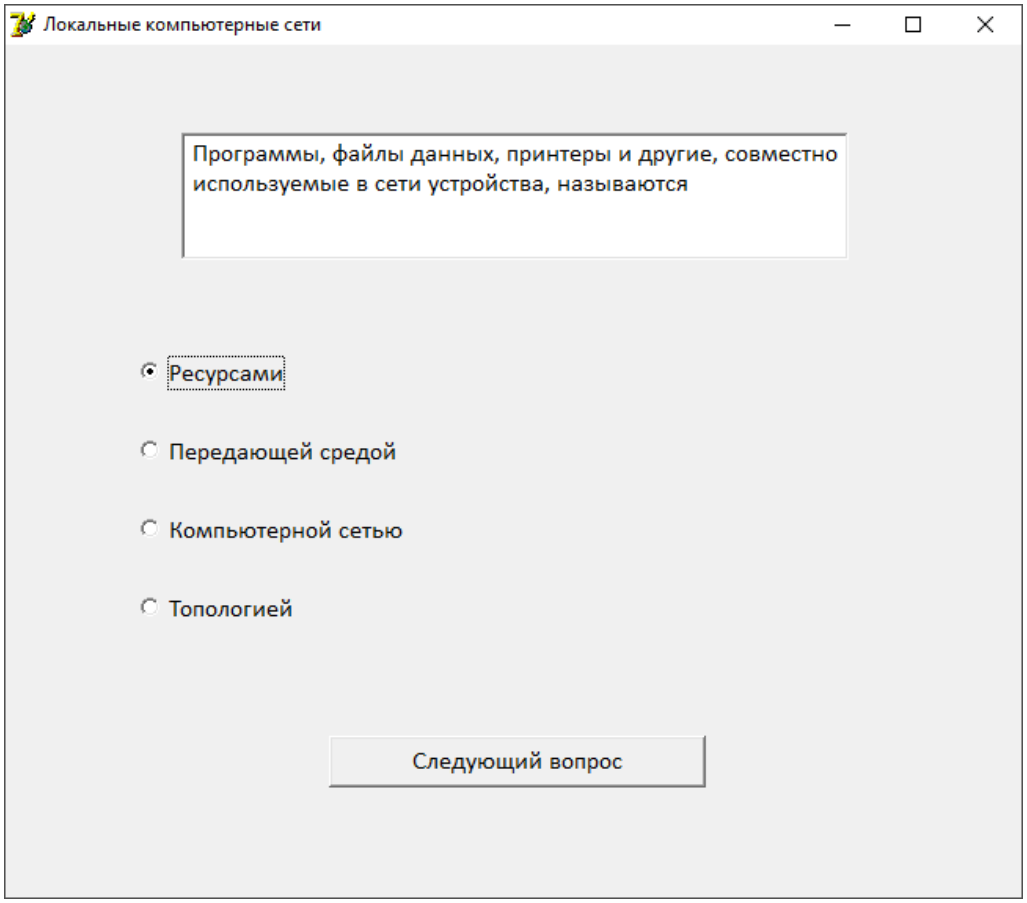


Рисунок 12 - Задание с одним вариантом ответа

При каждом нажатии на кнопку «Следующий вопрос» выполняется проверка ответа:

```
if (radiobutton1.checked=true) //Если выбран вариант ответа 1
  and (radiobutton1.Caption=memo2.lines[v*6+6])
thenball1:=ball1+1;
```

if (radiobutton2.checked=true)//Если выбран вариант ответа 2

*and (radiobutton2.Caption=memo2.lines[v*6+6])*

thenball1:=ball1+1;

И аналогично для RadioButton3 и RadioButton4. Выполняется это до тех пор, пока переменной kv не присвоится значение 11. Переменная ball1 содержит в себе количество баллов в первой части. Готовый вопрос выглядит следующим образом:

Далее идут задания с дополнением и свободным изложением. Здесь маска для каждого вопроса имеет другой вид:

*v*6+1 - Вопрос с текстовым ответом №1*

*v*6+2 - Правильный ответ №1*

*v*6+3 - Правильный ответ №2*

*v*6+4 - Правильный ответ №3*

*v*6+5 - Правильный ответ №4*

*v*6+6 - Правильный ответ №5*

Вводятся 5 различных вариантов ответа, учитывающие способы написания ответа, например, на вопрос «Сколько существует видов сетей по типу организации компьютеров в сети» можно дать следующие ответы: «2», «два», «Два», «2 вида», «Два вида», и так далее. Регистр букв в среде Delphiучитывается, поэтому для ответов с большой и маленькой буквы нужно ввести 2 разных ответа. Введенные 5 строк включают в себя наиболее вероятные письменные ответы школьников, и несут в себе один и тот же смысл.

При переходе с первого вида заданий на второй, компоненты RadioButton скрываются, а поле Edit становится видимым:

else if (button7.caption = 'Следующий вопрос')

and (kv>=11)

and (kv<=17) then

begin //Ввод ответа

while vopr[v]<>0 do v:=random(7)+10;

```
 radiobutton1.visible:=false;
radiobutton2.visible:=false;
 radiobutton3.visible:=false;
  radiobutton4.visible:=false;
   edit1.visible:=true;
memo1.text:=memo2.lines[v*6+1];
vopr[v]:=1;
  end
```

Проверка здесь выполняется следующим способом:

```
if ((edit1.Text=memo2.lines[v*6+2])   //Проверка текстового ответа
or (edit1.Text=memo2.lines[v*6+3])
  or (edit1.Text=memo2.lines[v*6+4])
  or (edit1.Text=memo2.lines[v*6+5])
  or (edit1.Text=memo2.lines[v*6+6]))
  and (edit1.Visible=true)
then ball2:=ball2+2;
```

Текст, введенный в Edit1, поочередно сравнивается со строками Memo2.Lines[v*6+2], Memo2.Lines[v*6+3], и так до пятого варианта ответа.В случае если введенный учеником ответ совпадает хоть с одним из предполагаемых учителем ответов, ему прибавляется 2 балла. Эти задания имеют следующий вид:

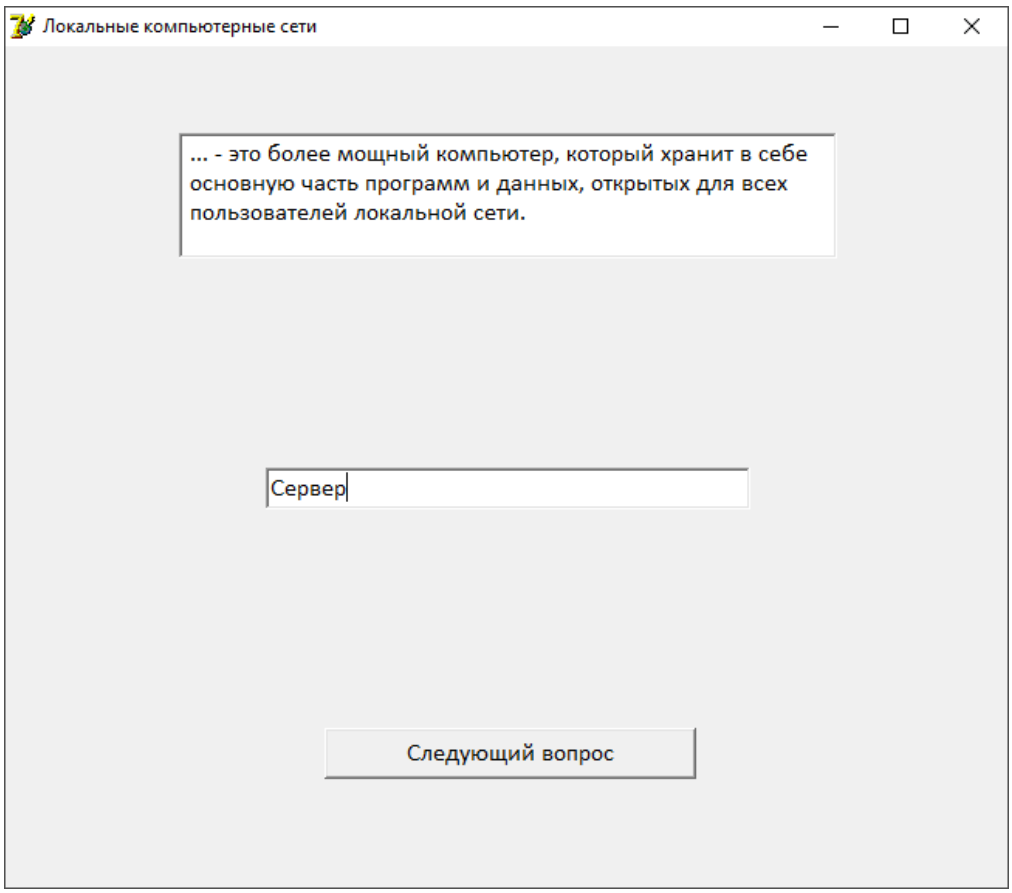


Рисунок 13 - Задания на дополнение или свободное изложение

Выполняется это до тех пор, пока переменная kvне примет значение 18.Тогда программа переходит к заключительной части теста – вопросы на установление соответствия. Алгоритм вывода и проверки вопросов в этой части наиболее сложный, поэтому приведена только часть кода, отвечающая за этот этап.

```
else if (button7.caption = 'Следующий вопрос') and (kv>17) then begin
label14.Visible:=true;
label15.Visible:=true;
label16.Visible:=true;
label17.Visible:=true;
randomize();
 while vopr[v]<>0 do v:=random(3)+17;
  s:=(v-7) mod 10;
 edit1.visible:=false;
button8.visible:=true;
```

```
  button9.visible:=true;
  button10.visible:=true;
  button11.visible:=true;
memo1.text:=memo2.lines[102+s*10+1];
label14.caption:=memo2.lines[102+s*10+2];
 Label15.caption:=memo2.lines[102+s*10+3];
Label16.caption:=memo2.lines[102+s*10+4];
Label17.caption:=memo2.lines[102+s*10+5];
  button8.caption:=memo2.lines[102+s*10+6];
  button9.caption:=memo2.lines[102+s*10+7];
  button10.caption:=memo2.lines[102+s*10+8];
  button11.caption:=memo2.lines[102+s*10+9];
voprr[v]:=1;
end
```

Как и в случае с предыдущими переходами, сначала Edit1 становится невидимым и на экране отображаются компоненты Label14, Label15, Label16, Label17,Button8, Button9, Button10, Button11. Компоненты Label– надписи, они находятся слева и несут в себе первую последовательность пунктов, т. е. 1), 2), 3) и 4), а компоненты Button –вторая последовательность, т.е. пункты а), б), в), г) – кнопки, при нажатии на которые их содержимое изменяется на следующий пункт, что и позволяет установить соответствие. Так как для этих вопросов отведено не из 6 строк, а из 10, было необходимо изменить способ работы с индексами, чтобы не нарушить работу программы.

Первый вопрос в данной категории начинается со 103-й строки, следовательно, маска вопроса будет иметь такой вид:

*102+s*10+1 - Вопрос*

*102+s*10+2- 1)*

*102+s*10+3 - 2)*

*102+s*10+4 - 3)*

*102+s*10+5 - 4)*

*102+s*10+6 - а)*

*102+s*10+7 - б)*

*102+s*10+8 - в)*

*102+s*10+9 - г)*

*102+s*10+10 - Верная последовательность (4 цифры)*

В первой строке дано то, что выводится на экран пользователя - пояснение к вопросу, либо просто фраза «Установите соответствие». Алгоритм проверки следующий:

```
if button8.Visible=true then begin    //Проверка соответствия
if button8.Caption=memo2.lines[102+s*10+6] then sootv:=sootv+'1';
if button8.Caption=memo2.lines[102+s*10+7] then sootv:=sootv+'2';
if button8.Caption=memo2.lines[102+s*10+8] then sootv:=sootv+'3';
if button8.Caption=memo2.lines[102+s*10+9] then sootv:=sootv+'4';
if button9.Caption=memo2.lines[102+s*10+6] then sootv:=sootv+'1';
...
if button11.Caption=memo2.lines[102+s*10+8] then sootv:=sootv+'3';
if button11.Caption=memo2.lines[102+s*10+9] then sootv:=sootv+'4';
if sootv=memo2.lines[102+s*10+10] then ball3:=ball3+2;
```

Как видно из кода, программа поочередно проверяет, соответствует ли левый первый пункт правому первому пункту, правому второму пункту, и так далее. Например, если учащийся считает, что к первому определению подходит понятие под буквой А, в пустую строку sootv записывается цифра 1, при выборе букв Б, В, Г – цифры 2, 3, 4, соответственно. Затем для второго определения также записывается цифра выбранного понятия. В итоге получается строка из 4 цифр, и, если она совпадает со строкой «Верная последовательность», то ответ на вопрос считается верным и учащийся получает 2 балла за выполнение задания. Так как вопросы в этой части несложные, в ней не учитываются частично правильные ответы. В итоге за всю правильно выполненную третью часть ученик может получить максимум 6 баллов.

Для переключения правых пунктов, в процедуре каждой кнопки хранится данный код:

```
procedure TForm1.button8Click(Sender: TObject);
begin
    if button8.caption=memo2.lines[102+s*10+6]
 then button8.caption:=memo2.lines[102+s*10+7] else
    if button8.caption=memo2.lines[102+s*10+7]
  then button8.caption:=memo2.lines[102+s*10+8] else
    if button8.caption=memo2.lines[102+s*10+8]
 then button8.caption:=memo2.lines[102+s*10+9] else
    if button8.caption=memo2.lines[102+s*10+9]
  then button8.caption:=memo2.lines[102+s*10+6];
  end;
```

Так выглядят задания этой части:

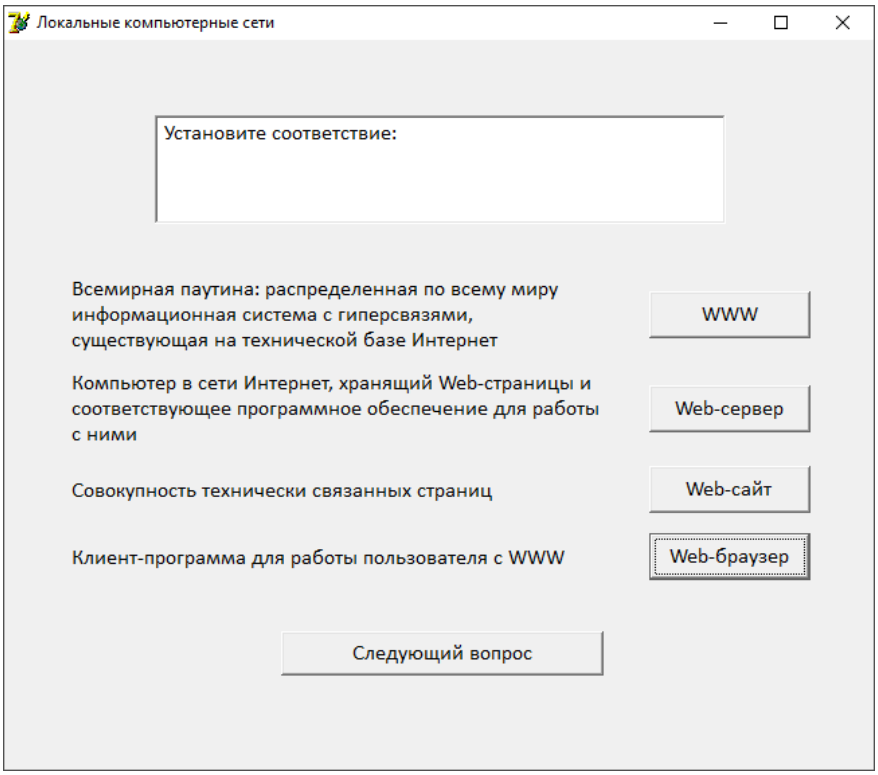


Рисунок 14 - Задания на установление соответствия

Когда переменной kv присваивается значение 20, это означает, что тестирование закончено, и надпись на кнопке Button7изменяется на

«Закончить тест». По итогу выполненного теста в переменных ball1, ball2 и ball3 хранятся баллы, полученные в каждой части. При нажатии на данную кнопку начинается подсчет результатов, и учащемуся выводится итоговая оценка. Далее приведен фрагмент процедуры вывода и подсчета оценки и как это выглядит:

```
else if button7.caption = 'Закончить тест' then begin
  label3.visible:=true;
  label11.visible:=true;
  label12.visible:=true;
  label13.visible:=true;
  label3.caption:=label3.Caption+inttostr(ball1)+' из 10';
  label11.caption:=label11.Caption+inttostr(ball2)+' из 16';
  label12.caption:=label12.Caption+inttostr(ball3)+' из 6';
  label19.visible:=true;
  label19.caption:=edit2.text;
  case ball1+ball2+ball3 of
    0..9:label13.caption:=label13.Caption+'2';
    10..20:label13.caption:=label13.Caption+'3';
    21..27:label13.caption:=label13.Caption+'4';
    28..32:label13.caption:=label13.Caption+'5';
  end;
  edit1.visible:=false;
  memo1.Visible:=false;
button7.caption:='Перейти к практическому заданию';
button7.Width:=button7.Width+20;
  button7.left:=button7.left-10;
  scrin:=false;
timer1.enabled:=true;
end
```

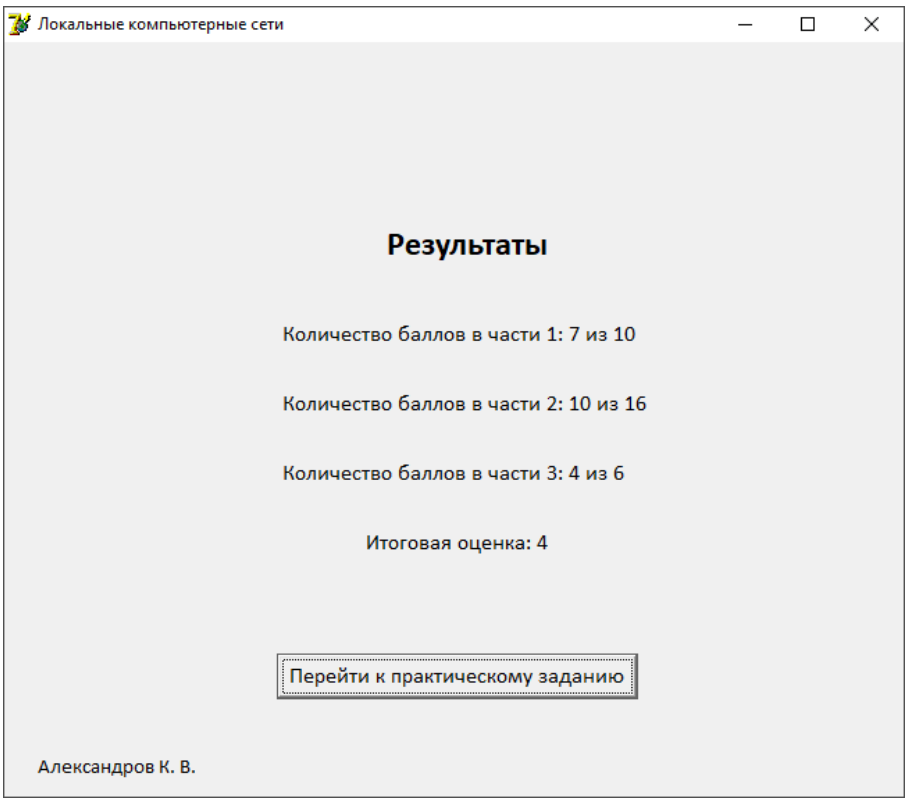


Рисунок 15 - Результаты тестирования

Если учащийся по итогу тестирования набрал от 0 до 9 баллов, он получает оценку «2», от 10 до 20 баллов - оценку «3», от 21 до 27 баллов – «4», и от 28 до 32 баллов – «5». В нижней части экрана в компонент Label19 выводится ФИО учащегося, которые он вводил в начале тестирования. После этого делается скриншот экрана, чтобы учитель впоследствии смог узнать и выставить оценки учащихся.

На этом разработка практической части была закончена. Следующим этапом стала теоретическая часть, предназначенная для вторичного изучения, повторения, либо закрепления материала.

Первым шагом были создана презентация в PowerPoint, в которую были добавлены материалы из учебника[33], с дополнительными изображениями и более систематизированным и расширенным содержанием. Затем все слайды был преобразованы в изображения с расширением 800x600 и загружены в компоненты Image для дальнейшей работы с ними.

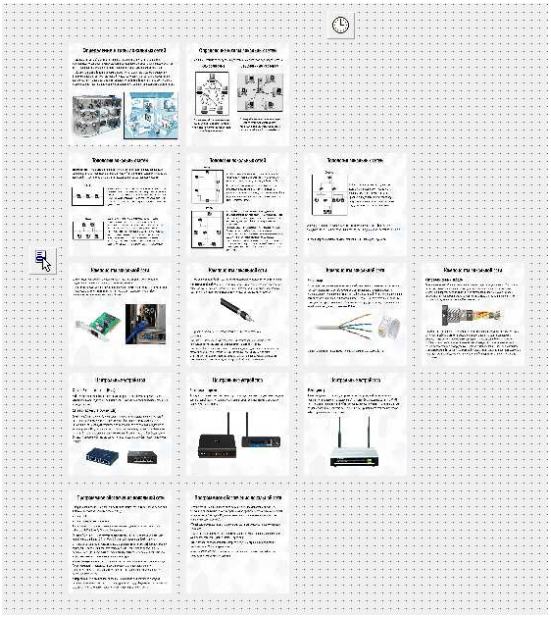


Рисунок 16 - Добавление теоретической части

Основной идеей для представления теории была форма презентации. Для осуществления такой идеи был составлен следующий алгоритм: при выборе темы компонент Image, соответствующий этой теме, становится видимым и его размер подгоняется под размер формы:

```
procedure TForm1.Button1Click(Sender: TObject);
begin
 tema;
 form1.Width:=805;
 form1.height:=610;
 image18.visible:=true;
 image18.left:=0;
image18.top:=-10;
 image18.width:=800;
 image18.height:=600;
 image19.left:=0;
 image19.top:=-10;
 image19.width:=800;
image19.height:=600;
end;
```

Далее, если нажать на это изображение, то оно сменится на следующее в данной теме, и так до тех пор, пока не дойдет до последнего изображения:

Рисунок 17 -Представление материала в виде презентации

Когда на экран выведено последнее изображение в подтеме, необходимо по нажатию на него вернуться в главное меню. Для этого при нажатии на него необходима другая процедура. Image становится невидимым, форма возвращает исходный размер:

```
procedureTForm1.Image19Click(Sender: TObject);
begin
  image19.Visible:=false;
  form1.Width:=360;
form1.Height:=370;
  temap;
end;
```

И выполняется процедура, обратная процедуре tema – temap:

```
procedure TForm1.temap; //Возвращает в главное меню программы после просмотра подтемы
begin
  label1.Visible:=true;
button1.Visible:=true;
```

```
 button2.Visible:=true;
 button3.Visible:=true;
 button4.Visible:=true;
 button5.Visible:=true;
button6.Visible:=true;
end;
```

На последнем этапе для упрощения работы школьников с программой и введения разнообразия заданий, было решено ввести дополнительный элемент в практическую работу программы. Над компонентом DrawGrid было добавлено поле Memo3. В нём выводятся разные задания для каждого ученика. Например, построить модель сети с топологией «Кольцо», используя как минимум 8 компьютеров, или построить модель сети с топологией «Дерево», используя как минимум 10 компьютеров. Также это необходимо для уменьшения шанса «списывания». Алгоритм работает следующим образом:

```
else if button7.Caption='Перейти к практическому заданию' then begin
form1.Height:=form1.height+50;
memo3.visible:=true;
r:=random(5);
case r of
0:memo3.Text:='Постройте модель сети с топологией "Шина"';
1:memo3.Text:='Постройте модель сети с топологией "Звезда"';
2:memo3.Text:='Постройте модель сети с топологией "Кольцо"';
3:memo3.Text:='Постройте модель сети с топологией "Дерево"';
4:memo3.Text:='Постройте модель сети со смешанной топологией';
end;
r:=random(4)+7;
memo3.Text:=memo3.text+', используя как минимум '+inttostr(r)+'
компьютеров. Объясните принцип работы модели.';
```

Число компьютеров генерируется случайным образом от 7 до 10. Таким образом, существует 20 различных вариантов заданий. Учитель также может попросить объяснить принцип работы модели, особенно это важно, если некоторым попалось задание со смешанной топологией.

На этом разработка программы заканчивается.

После тестирования программы, проверки ее работоспособности в разных ситуациях, были найдены некоторые недочеты и устранены. В конце был доработан интерфейс до понятного и удобного пользователю. В итоге программа работает оптимально и имеет следующий вид:

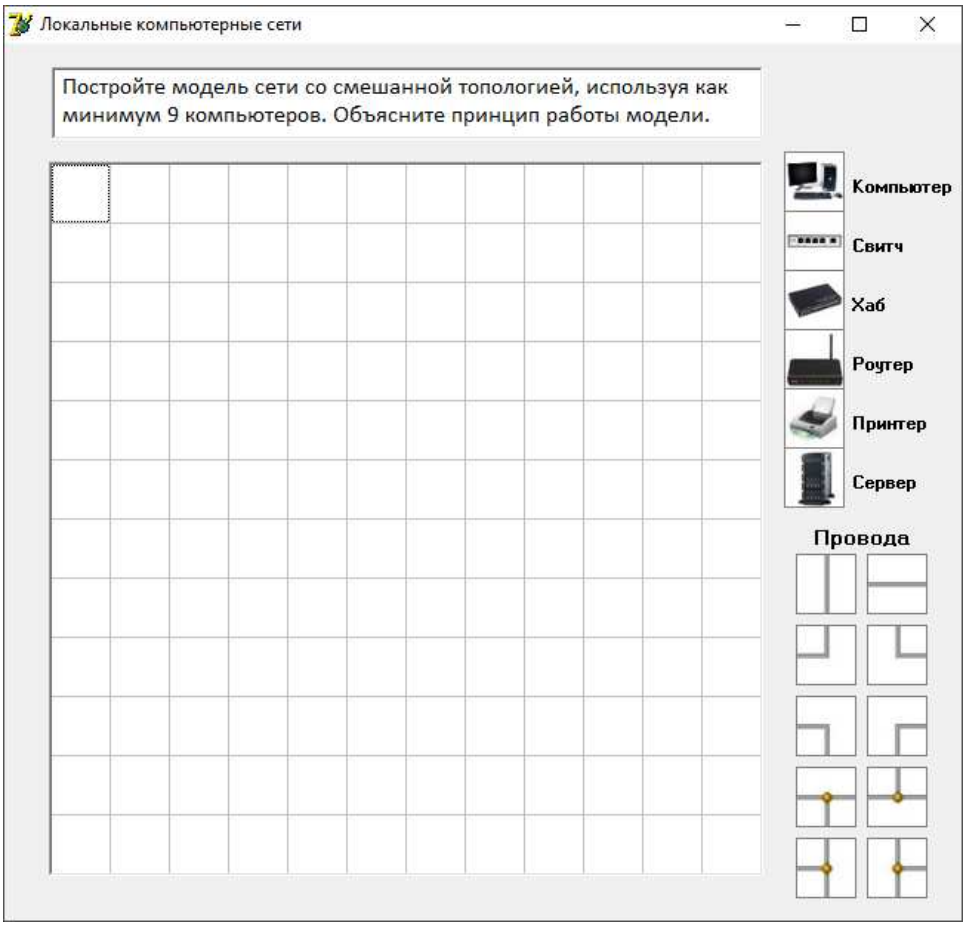


Рисунок 18 -Окончательный вид практической работы

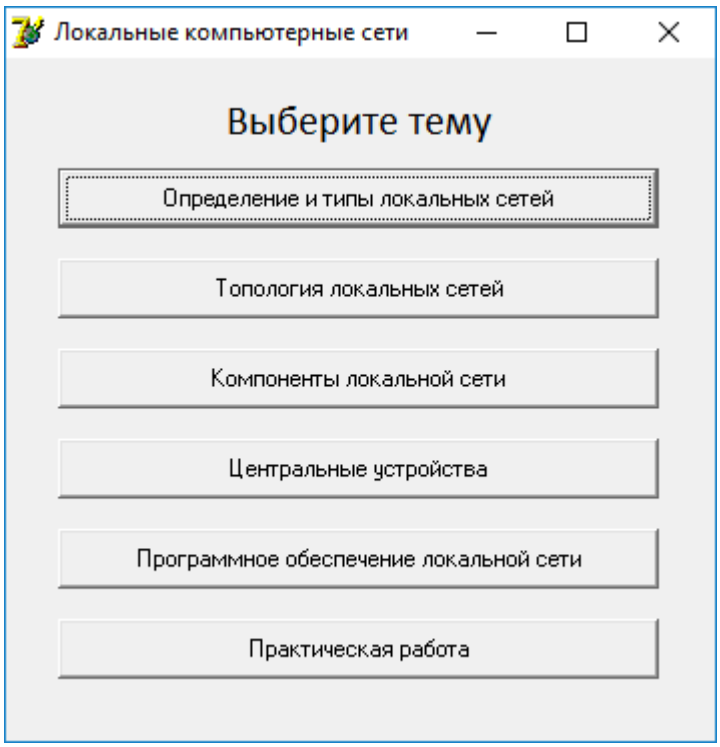


Рисунок 19 - Окончательный вид главного меню

2.2 Методические особенности применения программы для изучения темы «Локальные компьютерные сети»

Данную программу можно применять как на уроках изучения новой темы, так и на уроках закрепления материала. Ее можно даже дать учащемуся, который по какой-либо причине пропустил занятие, для самостоятельного изучения темы «Локальные компьютерные сети» дома, так как в ней все наглядно, доступно и содержательно. Также материал, данный в ней, соответствует материалу учебника. Конечно, учащийся не сможет при таком самостоятельном изучении дома задать учителю вопрос, если у него возникнут трудности в каком-либо моменте, но теория в программе сделана с таким учетом, чтобы у изучающего не осталось вопросов по теме.

Но для эффективного использования учащимися данной программы необходимо сначала показать и объяснить, как она работает, рассказать про её функционал, как добавлять, удалять элементы, как посмотреть теорию, как сохранить готовую работу, хоть эта программа и имеет довольно простой интерфейс, многие учащиеся могут не разобраться в ней.

Так как практическая часть программы – ее основное предназначение, следует использовать эту программу в конце урока, уже после изучения темы. Составим примерный конспект урока с применением данной программы.

Конспект урока **Тема:** Топология локальной компьютерной сети. **Цель:** Разобрать основные виды топологий компьютерной сети. **Задачи:** Обучающие: - научиться различать виды топологий компьютерных сетей. Воспитательные: - повышать познавательную	

активность учащихся (игровой момент в объяснении материала). Развивающие: - развитие познавательных интересов, внимательности, логического мышления. **Методы:** Объяснительно-иллюстративный. **Формы:** Фронтальная, индивидуальная. **Тип урока**: Изучения нового материала **Оборудование:** – компьютер; – мультимедийный проектор; – презентация по теме; – ПК учащихся; – моделирующая программа для изучения темы.	
Ход урока: **1. Оргмомент.** Приветствие, проверка готовности к уроку.	
2. Актуализация знаний. **Учитель:** Давайте вместе вспомним, что мы с вами разбирали на прошлом уроке, какие понятия узнали (Компьютерная сеть, локальная сеть, городская сеть, глобальная сеть)	
3. Изучение нового материала. **Учитель:** На прошлом уроке мы разобрали классификацию сетей по территориальному признаку. Сегодня мы узнаем, каким образом можно соединить компьютеры внутри сети. Открываем тетради. Записываем	

число и тему сегодняшнего урока. Тема: топология компьютерной сети. (Слайд 1) И начнем с определения топологии. Топология – (от греч. topos - место) физическое расположение компьютеров сети относительно друг друга и способ соединения их каналами связи. Запишем. (Слайд 2) Принято выделять 4 вида топологии. Каждый из них мы сейчас и рассмотрим. (Слайд 3) Первый вид: Топология типа шина, все рабочие станции подсоединены к общему кабелю (называемый шина или магистраль). Отправляемое рабочей станцией сообщение распространяется на все компьютеры сети. Каждая машина проверяет — кому адресовано сообщение и если ей, то обрабатывает его. А теперь давайте постараемся выделить ее достоинства и недостатки. Давайте запишем их. Достоинства: • Небольшое время установки сети; • Дешевизна (требуется меньше кабеля и сетевых устройств); • Простота настройки; • Выход из строя рабочей станции не отражается на работе сети;	

Недостатки:

- Любые неполадки канала связи уничтожают работу всей сети;
- Сложность поиска неисправностей;
- С добавлением новых рабочих станций падает производительность сети.

Второй вид: Топология типа кольцо, рабочие станции подключены последовательно друг к другу, образуя замкнутую сеть. (Слайд 4)

Компьютер в сети получает данные от стоящего предыдущим в списке адресатов и перенаправляет их далее, если они адресованы не ему.

Учитель: Как вы думаете, какие недостатки данной связи?

Учащиеся: отвечают.

Учитель: Давайте запишем их.

Достоинства:

- Простота установки;
- Возможность устойчивой работы без существенного падения скорости передачи данных при интенсивной загрузке сети;

Недостатки:

- Выход из строя одной рабочей станции, и другие неполадки (обрыв кабеля), отражаются на работоспособности всей сети;
- Сложность поиска неисправностей;

Третий вид: Топология типа

звезда, все компьютеры сети присоединены к центральному узлу. (Слайд 5)

Рабочая станция, которой нужно послать данные, отсылает их на концентратор, а тот определяет адресата и отдаёт ему информацию.

Что у нас получится, если один компьютер выйдет из строя? Верно. Он не сможет взаимодействовать с другими компьютерами в сети, но не нарушит остальную топологию.

А если из строя выйдет концентратор (центральный компьютер)?

Учащиеся: Вся сеть нарушится.

Учитель: Как вы думаете, сколько максимально можно подключить рабочих станций? –

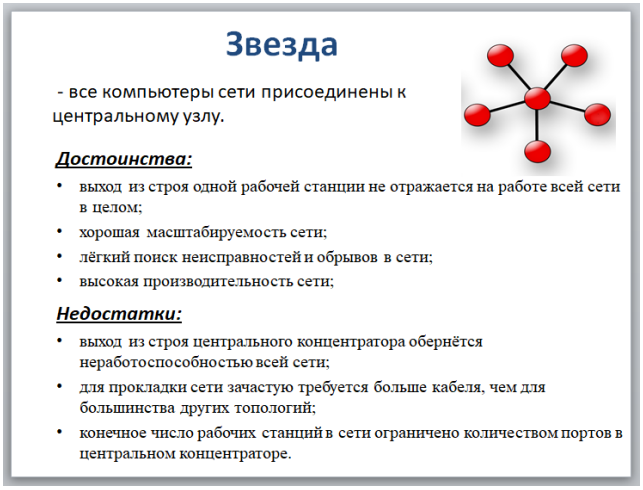
Учащиеся: предполагают.

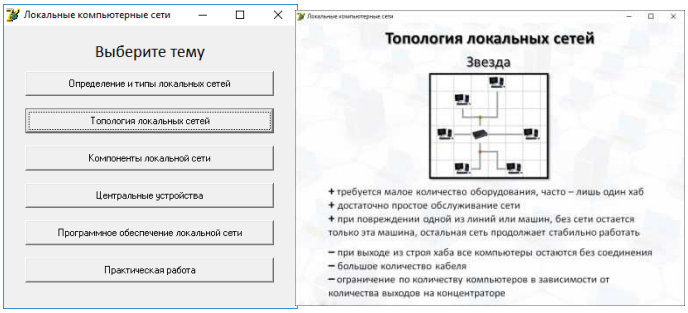
Учитель: Столько, сколько можно подключить к концентратору. Т.е. количество ограничено.

Давайте теперь запишем достоинства и недостатки этой сети.

Достоинства:

- выход из строя одной рабочей станции не отражается на работе всей сети в целом;
- хорошая масштабируемость сети;
- лёгкий поиск неисправностей и обрывов в сети;
- высокая производительность сети;



Недостатки: • выход из строя центрального концентратора обернётся неработоспособностью всей сети; • больший расход кабеля, чем для других топологий; • конечное число рабочих станций в сети ограничено количеством портов в центральном концентраторе. Три вида топологии мы с вами разобрали. Есть еще один. Четвертый вид: смешанная топология, объединение нескольких топологий. (Слайд 6). Достоинства и недостатки: зависят от составляющих ее видов топологий.	
4. Практическая часть **Учитель:** А сейчас мы будем решать практическое задание в специальной программе. Сначала я покажу вам, как она работает. Как вы видите, при открытии программы нам даётся на выбор 5 тем. Если кто-то не до конца понял материал сегодняшнего урока, можете ещё раз открыть тему Топология сетей и посмотреть момент, который вы упустили. (Показывает, как открыть). Кому всё понятно, сразу приступайте к выполнению заданий теста. Нажимаете кнопку Практическая работа. Вам	

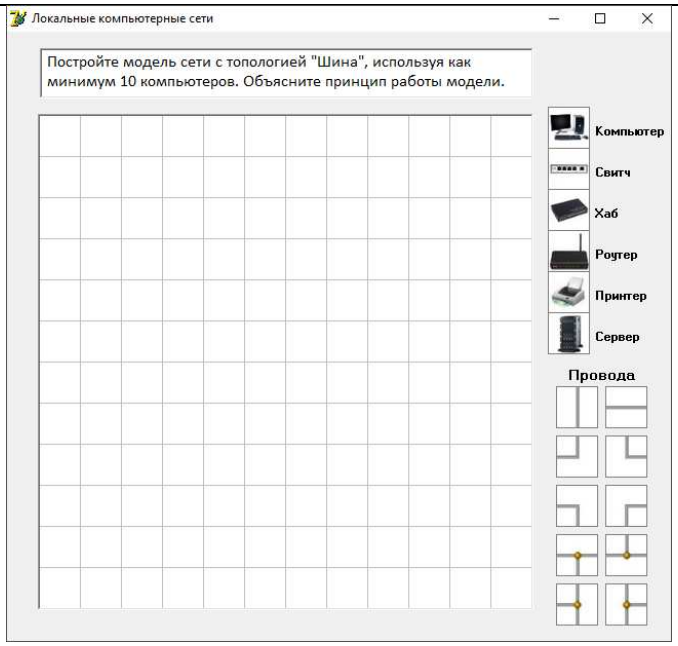
необходимо будет выполнить тест, состоящий из 20 вопросов. (Показывает, как выполнять)

После выполнения теста перед вами будет поле, разделенное на клетки.

Как вы видите, справа есть различные элементы сети, а также провода с разными направлениями и соединениями. Чтобы добавить элемент на поле, вы нажимаете на нужную клетку, а затем на то, что вы хотите в неё добавить. Можно посмотреть описание элемента, нажав на него правой кнопкой мыши и выбрав пункт «Инфо». Также вы видите в контекстном меню пункты «Удалить», чтобы очистить клетку, «Очистить всё», чтобы очистить сразу всё поле, и «Сохранить», когда вы полностью выполните задание, сохраните вашу работу. (Решение сохраняется в виде скриншота)

Вы разделитесь по вариантам, первый вариант будет строить топологию «Кольцо», состоящую из 10 компьютеров, а второй вариант топологию «Звезда», состоящий так же, из 10 компьютеров. Такие элементы, как Сервер и Принтер, вам сегодня не понадобятся, это мы изучим на последующих занятиях.

Кто быстро выполнит задание, я дам ещё одно задание, посложнее, на оценку. (Объединить 2 топологии и объяснить, как это



будет работать). Есть вопросы? **Учащиеся:** Садятся за компьютеры и выполняют задание.	
5. Рефлексия. **Учитель**: Итак, что нового вы сегодня узнали? Вам понравилось работать в этой программе? Вопросы по изученной теме.	
6. Заключительный этап. Домашнее задание.	

Таблица 1 - Конспект урока с применением программы

Таким образом, единственная сложность в проведении такого урока – доходчиво объяснить учащимся принцип работы программы. Данную программу можно использовать также и на последующих уроках, где учащимся будет необходимо построить модель сети на основе сервера, сеть в офисе с различными кабинетами и этажами, включить творчество и создать свою большую сеть, впоследствии рассказав, как она работает. Данный вид работы может заинтересовать учащихся и смотивировать их самостоятельно заняться изучением сетей, так как в 8 и 10 классе, в которых изучают данную тему, школьники уже задумываются о выборе будущей профессии, ищут то, в чём проявляются их способности, и данная тема может стать ключевой в их дальнейшем образовании.

ЗАКЛЮЧЕНИЕ

Таким образом, было выявлено, тема локальных компьютерных сетей достаточно актуальна, в будущем локальные сети во многих сферах вытеснят глобальную сеть Интернет, особенно на крупных предприятиях, где важна безопасность и анонимность информации.

Анализ особенностей изучения темы «Локальные компьютерные сети» показал, что учебный план не позволяет долго задерживаться на теме локальных сетей, поэтому необходимо за это время организовать работу так, чтобы учащиеся изучили данную тему с разных сторон, заинтересовались ей, и, возможно, задумались о её самостоятельном изучении.

Для этого мы, предварительно проанализировав учебную и методическую литературу по теме исследования и состояние изучения данной темы в учебных заведениях, разработали моделирующую программу, содержащую в себе как теоретические знания, представленные в наглядном, доступном виде, так и тестирование с практическим заданием для контроля и оценки знаний.

При наличии моделирующей программы учитель может помимо задания, данного в программе, предложить свое любое задание, в том числе и творческое, в котором учащемуся необходимо смоделировать определенную сеть, используя предоставленные ему элементы формы. Составить определенную топологию, придумать свою топологию, либо смоделировать локальную сеть в офисе из десятков компьютеров и сервера – все это можно сделать в разработанной программе.

Также для понимания того, в каком виде можно преподнести учащимся эту программу, как ее представить и объяснить, продемонстрировать её возможности, был составлен конспект урока с применением моделирующей программы.

Исходя из этого, следует сказать, что цель работы была достигнута, задачи выполнены.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

1. Абросимова, М. А. Информационные технологии в государственном и муниципальном управлении : учебное пособие / М. А. Абросимова. – Москва : КноРус, 2013. – 248 с.

2. Александров, К. В. Локальные компьютерные сети на современном этапе развития / К. В. Александров // Современные наукоёмкие инновационные технологии : сборник статей Международной научно-практической конференции. –Уфа: Аэтерна, 2018. – №2.– С. 28–30.

3. Ботт, Э. Локальные сети и безопасность Microsoft Windows XP : учебник / Э. Ботт. К. Зихерт. – Москва : Эком, 2007. – 944 с.

4. Бройдо, В. Л. Вычислительные системы, сети и телекоммуникации : учебник / В. Л. Бройдо. – Санкт-Петербург : Питер, 2014. – 703 с.

5. Величко, В. В. Телекоммуникационные системы и сети. Том 3. Мультисервисные сети : учебное пособие / В. В. Величко, Е. А. Субботин, В. П. Шувалов, А. Ф. Ярославцев. – Москва : Горячая Линия – Телеком, 2005. – 592 с.

6. Волкова, В. Н. Теория информационных процессов и систем : учебник и практикум для академического бакалавриата / В. Н. Волкова. – Москва : Юрайт, 2016. – 504 с.

7. Ги, К. Введение в локальные вычислительные сети : учебное пособие / К. Ги. – Москва : Радио и связь, 2004. – 176 с.

8. Гук, М. Аппаратные средства локальных сетей : энциклопедия / М. Гук. – Санкт-Петербург : Питер, 2004. – 573 с.

9. Епанешников, А. М. Локальные вычислительные сети : учебник / А. М. Епанешников, В. А. Епанешников. – Москва : Диалог–МИФИ, 2005. – 224 с.

10. Киселев, Г. М. Информационные технологии в педагогическом образовании : учебник / Г. М. Киселев, Р. В. Бочкова. – Москва : Дашков и К, 2013. – 308 с.

11. Козырев, А. А. Информационные технологии в экономике и управлении : учебник / А. А. Козырев. – Санкт-Петербург : Издательство Михайлова В. А., 2010. – 448 с.

12. Колбин, Р. В. Глобальные и локальные сети. Создание, настройка и использование : учебник / Р. В. Колбин. – Москва : Бином. Лаборатория знаний, 2008. – 614 с.

13. Кульгин, М. В. Коммутация и маршрутизация IP/IPX трафика : учебник / М. В. Кульгин. – Москва : Компьютер–пресс, 1998. – 320 с.

14. Куроуз, Д. Компьютерные сети. Нисходящий подход : учебник / Д. Куроуз, К. Росс. – Москва: Эксмо, 2016. – 916 с.

15. Лапчик, М. П. Методика обучения информатике : учебное пособие / М. П. Лапчик. – Санкт-Петербург : Лань, 2016. – 392 с.

16. Мелехин, В. Ф. Вычислительные системы и сети : учебник / В. Ф. Мелехин, Е. Г. Павловский. – Москва : Академия, 2013. – 208 с.

17. Михасев, В. Г. Компьютерные сети, Интернет и мультимедиа технологии : учебное пособие / В. Г. Михасев, Г. Б. Прончев. – Москва : МИПК им. И. Федорова, 2007. – 119 с.

18. Нужнов, Е. В. Компьютерные сети и телекоммуникации. Часть 1. Введение в компьютерные сети : учебник / Е. В. Нужнов. – Ростов-на-Дону : ЮФУ, 2014. – 163 с.

19. Олифер, Н. Компьютерные сети. Принципы, технологии, протоколы : учебник / В. Олифер, Н. Олифер. – Санкт-Петербург: Питер, 2016. – 944 с.

20. Осипов, Д. Л. Delphi. Программирование для Windows, OS X, iOS и Android : учебник / Д. Л. Осипов. – Санкт-Петербург : БХВ–Петербург, 2014. – 464 с.

21. Поляк–Брагинский, А. В. Локальные сети: модернизация и поиск неисправностей : учебник / А. В. Поляк–Брагинский. – Санкт-Петербург : БХВ–Петербург, 2012. – 640 с.

22. Поначугин, А. В. Использование суперкомпьютеров для решения

задач моделирования / А. В. Поначугин // Фундаментальные и прикладные исследования в современном мире. – 2015. – № 10-1. – С. 22-25.

23. Прончев, Г. Б. Компьютерные коммуникации. Простейшие вычислительные сети : учебник / Г. Б. Прончев, И. Н. Бухтиярова, В. В. Брутов, В. В. Фесенко. – Москва : КДУ, 2009. – 332 с.

24. Пятибратов, А. П. Вычислительные системы, сети и телекоммуникации. / А. П. Пятибратов, Л. П. Гудыно, А. А. Кириченко. – Москва : ФИС, 2002. – 568 с.

25. Растригин, Л. А. Вычислительные машины, системы, сети : учебник / Л. А. Растригин. – Москва : Наука, 2002. – 224 с.

26. Рыбакова, А. С. Информационные технологии: проблемы их внедрения, достоинства, недостатки / А. С. Рыбакова // Актуальные проблемы гуманитарных и естественных наук. – Москва : Институт Стратегических Исследований. – 2014. – № 11-2. – С. 24-27.

27. Самылкина, Н. Н. Построение тестовых заданий по информатике : методическое пособие / Н. Н. Самылкина. – Изд. 3-е. – Москва : Бином. Лаборатория знаний, 2016. – 179 с.

28. Санников, Е. В. Курс практического программирования в Delphi. Объектно–ориентированное программирование : учебник / Е. В. Санников. – Москва : Солон–Пресс, 2013. – 188 с.

29. Синаторов, С. В. Информационные технологии : учебное пособие / С. В. Синаторов. – Москва : Инфра-М, 2013. – 336 с.

30. Смолянинова, О. Г. Современные информационные технологии образования. Электронный портфолио : учебное пособие / О. Г. Смолянинова, Е. В. Достовалова, О. А. Лукьянова. – Красноярск : ИПК СФУ, 2009. – 134 с.

31. Советов, Б. Я. Информационные технологии : учебник для бакалавров / Б. Я. Советов, В. В. Цехановский. – Москва : Юрайт, 2013. – 263 с.

32. Степанов, А. Н. Архитектура вычислительных систем и компьютерных сетей : учебник / А. Н. Степанов. – Изд. 2–е. – Санкт-

Петербург : Питер, 2007. – 509 с.

33. Угринович, Н. Д. Информатика: учебник для 8 класса / Н. Д. Угринович. – Изд. 3-е, – Москва : Бином. Лаборатория званий, 2015. – 160 с.

34. Угринович, Н. Д. Практикум по информатике и информационным технологиям : практикум / Н. Д. Угринович, Л. Л. Босова, Н. И. Михайлова. – Москва : Бином. Лаборатория знаний, 2013. – 394 с.

35. Федотова, Е. Л. Информационные технологии и системы : учебное пособие / Е. Л. Федотова. – Москва : Инфра-М, 2013. – 352 с.

36. Флинт, Д. Локальные сети ЭВМ: архитектура, принципы построения, реализация : учебник / Д. Флинт. – Москва : Финансы и статистика, 2002. – 359 с.

37. Фролов, А. В. Локальные сети персональных компьютеров. Использование протоколов IPX, SPX, NETBIOS : учебник / А. В. Фролов, Г. В. Фролов. – Москва : Диалог–Мифи, 2013. – 160 с.

38. Хандадашева, Л. Н. Программное обеспечение. Вычислительные сети : учебник / Л. Н. Хандадашева, И. Г. Истомина. – Москва : МарТ, 2005. – 320 с.

39. Хлебников, А. А. Информационные технологии : учебник / А. А. Хлебников. – Москва : КноРус, 2014. – 472 с.

40. Чекмарев, Ю. В. Локальные вычислительные сети : учебник / Ю. В. Чекмарев. – Москва : ДМК Пресс, 2008. – 200 с.

41. Черников, Б. В. Информационные технологии управления : учебник / Б. В. Черников. – Москва : Инфра-М, 2013. – 368 с.

Федеральное государственное автономное
образовательное учреждение
высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт педагогики, психологии и социологии

УТВЕРЖДАЮ
Заведующий кафедрой

_____ _____
подпись инициалы, фамилия

« ___ » _____ 2018 г

ЗАДАНИЕ

НА ВЫПУСКНУЮ КВАЛИФИКАЦИОННУЮ РАБОТУ

в форме бакалаврской работы

Студенту Александрову Кириллу Валерьевичу
фамилия, имя, отчество

Направление 44.03.01. Педагогическое образование

Тема выпускной квалификационной работы Разработка моделирующей программы для изучения темы «Локальные компьютерные сети»

Утверждена приказом по университету № _____ от _____

Руководитель ВКР Е. В. Киргизова, доцент кафедры ВМИиЕ, кандидат педагогических наук

Исходные данные для ВКР

1. Курсовая работа на тему «Изучение темы «Компьютерные сети» в курсе информатики старшей школы»
2. Научная статья на тему «Локальные компьютерные сети на современном этапе развития»

Перечень разделов ВКР

1. Теоретические и методические аспекты изучения темы «Локальные компьютерные сети» в курсе информатики
2. Разработка моделирующей программы для изучения темы «Локальные компьютерные сети»

Перечень графического материала: иллюстрации, таблицы

Руководитель ВКР Е. В. Киргизова _____
инициалы и фамилия подпись

Задание принял к исполнению К. В. Александров _____
инициалы и фамилия подпись

« ___ » _____ 20__ г.

Федеральное государственное автономное
образовательное учреждение
высшего образования
«СИБИРСКИЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

Институт педагогики, психологии и социологии

Кафедра информационных технологий обучения и непрерывного образования

УТВЕРЖДАЮ
Заведующий кафедрой
_____ О.Г. Смолянинова
подпись
« 15 » 06 2018 г.

БАКАЛАВРСКАЯ РАБОТА

44.03.01 Педагогическое образование

Разработка моделирующей программы для изучения темы «Локальные компьютерные сети»

Руководитель _____ доцент, канд.пед.наук Е.В.Киргизова
подпись, дата

Выпускник _____ К.В.Александров
подпись, дата

Красноярск 2018