

**Автономное образовательное учреждение  
высшего образования Ленинградской области  
«Государственный институт экономики, финансов, права и технологий»**

**Кафедра информационных технологий и высшей математики**

**Бенза Е. В., Бенза С. М.**

## **ВЫЧИСЛИТЕЛЬНЫЕ СИСТЕМЫ, СЕТИ И КОММУНИКАЦИИ**

Методические указания для выполнения лабораторных работ  
для студентов экономического факультета очной и очно - заочной  
формы обучения  
(уровень бакалавриат)

Гатчина  
2023

Методические указания для выполнения лабораторных и практических работ для студентов экономического факультета очной и очно - заочной формы обучения по курсу «Вычислительные системы, сети и коммуникации» рассмотрено и утверждено на заседании кафедры информационных технологий и высшей математики, протокол № 3 от 25 октября 2023 года.

.

Составители: доцент кафедры информационных технологий и высшей математики ГИЭФПТ, к.т.н., доцент Бенза Елена Владимировна; старший преподаватель кафедры информационных технологий и высшей математики ГИЭФПТ, системный администратор АО «Лентехностром» Бенза Сергей Маркович.

Рецензент: зав. кафедрой информационных технологий и высшей математики ГИЭФПТ, д.т.н., профессор Драбенко В. А.

Данное пособие содержит задания для выполнения лабораторных работ и практических, а также требования по содержанию и оформлению отчёта по работе.

Методические указания предназначены для студентов всех форм обучения по экономическим специальностям.

## ВВЕДЕНИЕ

В настоящее время использование сетевых технологий является необходимым условием работы с информацией, прежде всего её транспортированием. Практически все сферы человеческой деятельности сейчас используют различные сетевые сервисы и системы.

Деятельность специалиста по бизнес - информатике неразрывно связана с использованием различных сетевых технологий и сервисов. Применение в бизнесе подобных технологий позволяет любой организации повысить эффективность работы с информацией и улучшить качество и оперативность деловых коммуникаций.

Дисциплина «Вычислительные системы, сети и телекоммуникации» рассматривает различные технологии и стандарты построения локальных и глобальных сетей, основы программирования и администрирования сетевых служб и компонентов.

Целесообразность изучения данной дисциплины обосновывается необходимостью знания теоретических методов и практических приемов организации, проектирования и администрирования сетей, применяемых сегодня в сфере информационных технологий.

Целью освоения дисциплины является формирование у студентов целостного представления о вычислительных сетях различных топологий, масштабов и технологий коммутации, принципах их работы и построения и о возможности их использования в профессиональной деятельности.

Задачи дисциплины:

1. Сформировать у студентов представление об основных понятиях, используемых в вычислительных системах, сетях и телекоммуникациях;
2. Изучить принципы построения и функционирования вычислительных систем фон-Неймановского типа;
3. Дать представление о компонентах вычислительных систем и их параметрах;
4. Изучить состав и принципы функционирования модели открытых сетей OSI/ISO и сетевых протоколах семейства TCP/IP;
5. Дать представление об устройстве современных локальных вычислительных сетей (ЛВС), топологиях ЛВС, компонентах ЛВС и их параметрах, правилах монтажа;
6. Изучить основные понятия, связанные с операционными системами;
7. Изучить принципы построения, функциональную и структурную организацию современных операционных систем (ОС);

Изучить принципы управления ресурсами; методы управления вычислительными процессами, вводом-выводом, реальной памятью; виртуальной памятью; способы организации файловых систем, а также методы передачи данных в телекоммуникационных системах. Пособие рассчитано на студентов специальности 38.03.05 – «Бизнес – информатика», изучающих дисциплину «Вычислительные системы, сети и коммуникации».

Данное пособие содержит краткое изложение теоретических сведений,

необходимых для выполнения лабораторных работ; описание самих работ.

## ЛАБОРАТОРНАЯ РАБОТА № 1. Исследование кабеля «Витая пара» СХЕМА ОБЖИМА ВИТОЙ ПАРЫ

### Обжим и разводка, распайка, распиновка витой пары.

Структурированные кабельные системы (СКС) – сравнительно новая область, и терминология в ней пока еще не устоялась. Поэтому существует ряд взаимозаменяемых названий, употребляющихся при обжиге витой пары, таких как «распайка», «распиновка», «разводка» кабеля. По сути, весь это профессиональный жаргон – синонимы слова «схема», и обозначает описание взаимного расположения каждого контакта разъема, или порядок расположения контактов в схеме.

Разводка и обжим сетевого кабеля зависит от того, сколько витых пар в нем находится.

Стандартный 4-х парный кабель, используемый для построения компьютерных сетей, обжимается коннекторами RJ-45. Телефонный кабель обжимается коннекторами RJ-11 или RJ-12, в зависимости от количества активных контактов.

### Схема обжима витой пары 568В

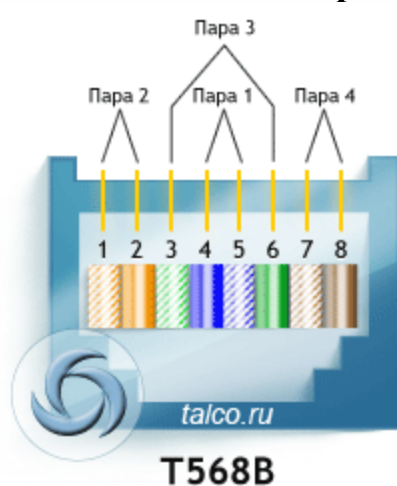


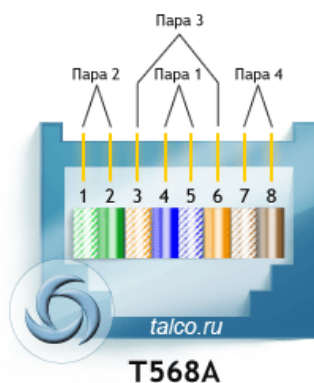
Рисунок 1 –Схема обжима витой пары 568В

Отличие схем **568А** и **568В** состоит только в оранжевой и зеленых парах.

**568А** - 1-2 - бело-зеленый - зеленый, 3-6 бело-оранжевый - оранжевый

**568В** - 1-2 - бело-оранжевый - оранжевый, 3-6 бело-зеленый - зеленый

## Схема обжима витой пары 568А



1. бело-зеленый
2. зеленый
3. бело-оранжевый
4. синий
5. бело-синий
6. оранжевый
7. бело-коричневый
8. коричневый

Рисунок 2 –Схема обжима витой пары 568А

**Схемы обжима витой пары** для разъемов RJ-45 различаются в зависимости от назначения соединительной линии, технологии и стандарта передачи данных. Может потребоваться как прямая, так и обратная (или перекрестная, т.н. кросс-линковая) обжимка патчкорда. В локальных вычислительных сетях Ethernet, использующих кабель "витая пара" или UTP (Unshielded Twisted Pair, англ.: неэкранированная витая пара), наиболее широко используются схемы прямого обжима в соответствии с телекоммуникационными стандартами кабельных систем коммерческих зданий **568А** и **568В**. Схемы разработаны таким образом, чтобы свести к минимуму взаимные наводки в парах, поэтому при реализации высокоскоростных сетей используют именно эти конфигурации.

Для обжима витой пары рекомендуем использовать только специальный обжимной инструмент: "зачистка", обжимные клещи, тестеры для проверки работоспособности обжатых пар. Стандартные клещи для обжимки коннекторов (кримпер) RJ 45 8P8C.

Оборудование для проверки – необходимый компонент, поскольку некачественный обжим всего одной жилы может нарушить работу всего соединения, а обнаружить без прибора его не так-то просто.

В итоге, если предполагается проводить обжим витых пар часто и в больших объемах – рекомендуем приобрести набор профессиональных инструментов.

### **Ход работы**

#### **Обжим витой пары:**

**1.** Отрежьте кусок витой пары нужной длины от бухты, при этом можно воспользоваться резакон, встроенным в обжимной инструмент.

**2.** Аккуратно снимите изоляцию с кабеля на длину примерно 3 см. Для этого лучше использовать специальный инструмент для зачистки изоляции витой пары, его лезвие выступает ровно на толщину изоляции, так вы не повредите проводники. Если такого инструмента нет под рукой, то можно воспользоваться обычным ножом или даже ножницами. Весь вопрос в удобстве и скорости.

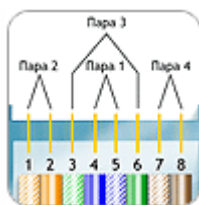


Рисунок 3 - Схема обжима витой пары

3. Расплетите проводники не более чем на 2 см (для минимизации электромагнитных помех) затем проводники следует развести друг от друга, выровняйте их в один ряд, при этом соблюдая схему обжима витой пары. Стандартно при обжиме UTP используют вариант T568B.

4. Обкусите проводники таким образом, чтобы их длина от изоляции была чуть больше сантиметра, рекомендованная длина 1/2 дюйма или 12,5 мм. Для этого можно воспользоваться инструментом для обрезки витой пары, или ножами встроенными в обжимной инструмент.

5. Аккуратно вставьте проводники в коннектор RJ-45. Обратите внимание чтобы расположение проводов относительно коннектора при обжиме второго конца провода полностью совпадало с первым.

6. Обязательно проверьте не перепутались ли проводники и правильно ли они вошли в коннектор, при этом все провода должны упереться в переднюю стенку коннектора.

7. Поместите коннектор с расположенными в нем проводниками в клещи, затем плавно, но сильно произведите *обжим витой пары*. Вторым коннектором обжимается по той же схеме что и первый, однако некоторых случаях (например при соединении активного сетевого оборудования или двух компьютеров без использования свитча) Вам может потребоваться обратная или cross-over схема обжима. В этом случае для второго коннектора используйте схему T568A.

8. Обязательно следует проверить правильность обжатия коннектора на предмет отсутствия контакта или несоблюдения последовательности в отдельных проводниках. Это лучше всего сделать специальным тестировочным инструментом.

### ПРАКТИЧЕСКАЯ РАБОТА №1. Расчет конфигурации сети Ethernet

Для упрощения расчетов используются справочные данные IEEE, содержащие значения задержек распространения сигналов в повторителях, приемопередатчиках и различных физических средах. В таблице №1 приведены данные, необходимые для значения расчетов PDV для всех физических стандартов сетей Ethernet. Битовый интервал обозначен как bt.

Таблица 1. - Данные для расчета значения PDV.

Тип сегмента	База левого сегмента, bt	База промежуточного сегмента, bt	База правого сегмента, bt	Задержка среды на 1 м, bt	Максимальная длина сегмента, м
10Base-5	11,8	46,5	169,5	0,0866	500

10Base-2	11,8	46,5	169,5	0,1026	185
10Base-T	15,3	42,0	165,0	0,113	100
10Base-FB	-	24,0	-	0,1	2000
10Base-FL	12,3	33,5	156,5	0,1	2000
FOIRL	7,8	29,0	152,0	0,1	1000
AUI(>2 м)	0	0	0	0,1026	2+48

Комитет старался максимально упростить выполнение расчетов, поэтому данные, приведенные в таблице, включают сразу несколько этапов прохождения сигнала. Например, задержки, вносимые повторителем, состоят из задержки входного трансивера, задержки блока повторения и задержки выходного трансивера. Тем не менее в таблице все эти задержки представлены одной величиной, названной базой сегмента.

Чтобы не нужно было два раза складывать задержки, вносимые кабелем, в таблице даются удвоенные величины задержек для каждого типа кабеля.

В таблице используется также такие понятия, как левый сегмент, правый сегмент и промежуточный сегмент. Поясним эти термины на примере сети, приведенной на рис 4. Левым сегментом называется сегмент, в котором начинается путь сигнала от входа передатчика (выход  $T_x$  на рис 4) конечного узла. На примере это сегмент 1. Затем сигнал проходит через промежуточные сегменты 2-5 доходит до приемника (вход  $R_x$ ) наиболее удаленного узла наиболее удаленного сегмента 6, который называется правым. Именно здесь в худшем случае происходит столкновение кадров и возникает коллизия, что и подразумевается в таблице.

С каждым сегментом связана постоянная задержка, названная базой, которая зависит только от типа сегмента и от положения сегмента на пути сигнала (левый промежуточный или правый). База правого сегмента, в котором возникает коллизия, намного превышает базу левого и промежуточных сегментов.

Кроме этого, с каждым сегментом связана задержка распространения сигнала вдоль кабеля сегмента, которая зависит от длины сегмента и вычисляется путем умножения времени распространения сигнала по одному метру кабеля (в битовых интервалах) на длину кабеля в метрах.

Расчет заключается в вычислении задержек, вносимых каждым отрезком кабеля (приведенная в таблице задержка сигнала на 1 м кабеля умножается на длину сегмента), а затем суммировании этих задержек с базами левого, промежуточных и правого сегментов. Общее значение PDV не должно превышать 575.

Так как левый и правый сегменты имеют разные величины базовой задержки, то в случае различных типов на удаленных краях сети необходимо

выполнить расчеты дважды: один раз принять в качестве левого сегмента сегмент одного типа, а во второй – сегмент другого типа. Результатом можно считать максимальное значение PDV. В нашем примере крайние сегменты сети принадлежат к одному типу – стандарту 10Base – T, поэтому двойной расчет не требуется, но если бы они были сегментами разного типа, то в первом случае нужно было бы принять в качестве левого сегмента между станцией и концентратором 1, а во втором считать левым сегмент между станцией и концентратором 5.

Приведенная на рисунке сеть в соответствии с правилом 4 хабов не является корректной – в сети между узлами сегментов 1 и 6 имеется 5 хабов, хотя не все сегменты являются сегментами 10Base-FB. Кроме того, общая длина сети равна 2800 м, что нарушает правило 2500 м. Рассчитаем значение PDV для нашего примера.

- Левый сегмент 1:  
 $15,3(\text{база}) + 100 * 0,113 = 26,6$
- Промежуточный сегмент 2:  
 $33,5 + 1000 * 0,1 = 133,5$
- Промежуточный сегмент 3:  
 $24 + 500 * 0,1 = 74,0$
- Промежуточный сегмент 4:  
 $24 + 500 * 0,1 = 74,0$
- Промежуточный сегмент 5:  
 $24 + 600 * 0,1 = 84,0$
- Правый сегмент 6:  
 $165 + 100 * 0,113 = 176,3$

Сумма всех составляющих дает значение PDV, равное 568,4.

Так как значение PDV меньше максимально допустимой величины 575, то эта сеть проходит по критерию времени двойного оборота сигнала несмотря на то, что ее общая длина превышает 2500 м, а количество повторителей больше 4.

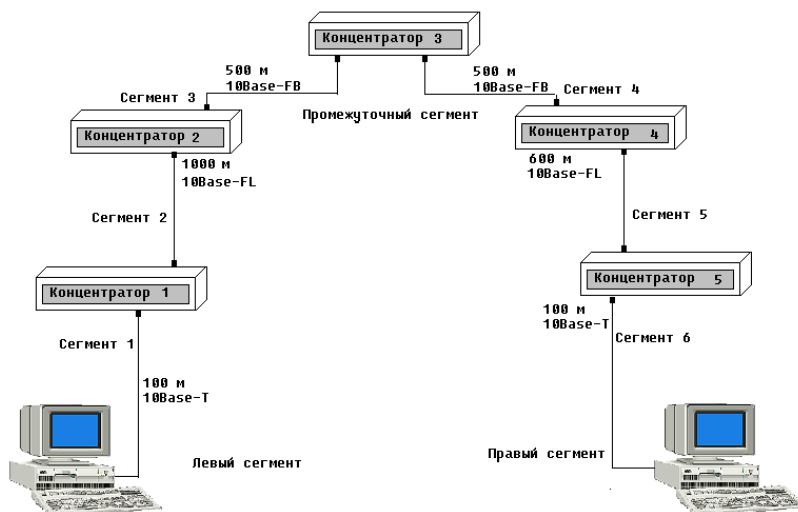


Рисунок 4. - Пример сети Ethernet, состоящей из сегментов различных физических стандартов.



## Задания для самостоятельной работы:

### Задание 1

Произвести расчеты сети при конфигурациях, отображенных в таблице №2. согласно вашего варианта

Таблица 2. – Данные для самостоятельной работы

№ Варианта	10Base-5	10Base-2	10Base-T	10Base-FB	10Base-FL
1	500	200	200	900	1000
2	400	250	190	100	500
3	370	180	195	000	850
4	510	190	150	200	670
5	390	210	175	900	850
6	290	200	160	800	970
7	420	210	185	750	850
8	420	175	195	950	860
9	600	190	190	850	1050
10	390	150	165	1000	970
11	380	250	180	100	690
12	450	300	150	800	750
13	420	250	200	700	700
14	380	225	190	500	650
15	500	270	160	400	600
16	410	170	140	1000	800
17	465	180	200	850	100

Таблица 3. – Данные для самостоятельной работы

Таблица 3. – Данные для самостоятельной работы

Задания 2,3,4		
1 сегмент –10 Base 2- 200 m	1 сегмент – 10 Base T-200 m	1 сегмент –10 Base T-300 m
2 сегмент-10 Base FL- 800 m	2 сегмент-10 Base FL-800 m	2 сегмент –10 Base FL-600 m
3 сегмент-10 Base FB-400 m	3 сегмент-10 Base T- 500 m	3 сегмент –10 Base T-1000 m
4 сегмент –10 Base FB-800 m	4 сегмент –10 Base T-1000 m	4 сегмент-10 Base FL-500 m
5 сегмент-10 Base FB-400 m	5 сегмент-10 Base T-400 m	
6 сегмент-10 Base 2-300 m		

## ЛАБОРАТОРНАЯ РАБОТА № 2. Тестирование работы сети.

### Диагностические утилиты TCP/IP

В состав TCP/IP входят диагностические утилиты, предназначенные для проверки конфигурации стека и тестирования сетевого соединения.

Утилита	Применение
hostname	Выводит имя локального хоста. Используется без параметров.
ipconfig	Выводит значения для текущей конфигурации стека TCP/IP: IP-адрес, маску подсети, адрес шлюза по умолчанию, адреса WINS (Windows Internet Naming Service) и DNS (Domain Name System)
ping	Осуществляет проверку правильности конфигурирования TCP/IP и проверку связи с удаленным хостом.
tracert	Осуществляет проверку маршрута к удаленному компьютеру путем отправки эхо-пакетов протокола ICMP (Internet Control Message Protocol). Выводит маршрут прохождения пакетов на удаленный компьютер.
arp	Выводит для просмотра и изменения таблиц трансляции адресов, используемую протоколом разрешения адресов ARP (Address Resolution Protocol - определяет локальный адрес по IP-адресу)

route	Модифицирует таблицы маршрутизации IP. Отображает содержимое таблицы, добавляет и удаляет маршруты IP.
netstat	Выводит статистику и текущую информацию по соединению TCP/IP.
nslookup	Осуществляет проверку записей и доменных псевдонимов хостов, доменных сервисов хостов, а также информации операционной системы, путем запросов к серверам DNS.

## 1. Проверка правильности конфигурации TCP/IP с помощью ipconfig.

При устранении неисправностей и проблем в сети TCP/IP следует сначала проверить правильность конфигурации TCP/IP. Для этого используется утилита ipconfig.

Эта команда полезна на компьютерах, работающих с DHCP (Dynamic Host Configuration Protocol), так как дает пользователям возможность определить, какая конфигурация сети TCP/IP и какие величины были установлены с помощью DHCP.

### Синтаксис:

```
ipconfig [/all | /renew[adapter] | /release]
```

### Параметры:

**all** выдает весь список параметров. Без этого ключа отображается только IP-адрес, маска и шлюз по умолчанию;

**renew[adapter]** обновляет параметры конфигурации DHCP для указанного сетевого адаптера;

**release[adapter]** освобождает выделенный DHCP IP-адрес;

**adapter** – имя сетевого адаптера;

**displaydns** выводит информацию о содержимом локального кэша клиента DNS, используемого для разрешения доменных имен.

Таким образом, утилита ipconfig позволяет выяснить, инициализирована ли конфигурация и не дублируются ли IP-адреса:

- если конфигурация инициализирована, то появляется IP-адрес, маска, шлюз;
- если IP-адреса дублируются, то маска сети будет 0.0.0.0;
- если при использовании DHCP компьютер не смог получить IP-адрес, то он будет равен 0.0.0.0 .

## 2. Тестирование связи с использованием утилиты ping.

Утилита ping (Packet Internet Grouper) используется для проверки конфигурирования TCP/IP и диагностики ошибок соединения. Она определяет доступность и функционирование конкретного хоста. Использование ping лучший способ проверки того, что между локальным компьютером и сетевым хостом существует маршрут. Хостом называется любое сетевое устройство (компьютер, маршрутизатор), обменивающееся информацией с другими сетевыми устройствами по TCP/IP.

Команда ping проверяет соединение с удаленным хостом путем отправки к этому хосту эхо-пакетов ICMP и прослушивания эхо-ответов. Ping ожидает каждый посланный пакет и печатает количество переданных и принятых пакетов. Каждый принятый пакет проверяется в соответствии с переданным сообщением. Если связь между хостами плохая, из сообщений ping станет ясно, сколько пакетов потеряно.

По умолчанию передается 4 эхо-пакета длиной 32 байта (возможны и другие варианты значения по умолчанию) - периодическая последовательность символов алфавита в верхнем регистре. Ping позволяет изменить размер и количество пакетов, указать, следует ли записывать маршрут, который она использует, какую величину времени жизни (ttl) устанавливать, можно ли фрагментировать пакет и т.д.. При получении ответа в поле time указывается, за какое время (в миллисекундах) отправленный пакет доходит до удаленного хоста и возвращается назад. Так как значение по умолчанию для ожидания отклика равно 1 секунде, то все значения данного поля будут меньше 1000 миллисекунд. Если вы получаете сообщение «Request time out» (Превышен интервал ожидания), то, возможно, если увеличить время ожидания отклика, пакет дойдет до удаленного хоста. Это можно сделать с помощью ключа -w.

Ping можно использовать для тестирования как имени хоста (DNS или NetBIOS), так и его IP-адреса. Если ping с IP-адресом выполнялась успешно, а с именем – неудачно, это значит, что проблема заключается в распознавании соответствия адреса и имени, а не в сетевом соединении.

Утилита ping используется следующими способами:

1) Для проверки того, что TCP/IP установлен и правильно сконфигурирован на локальном компьютере, в команде ping задается адрес петли обратной связи (loopback address):

```
ping 127.0.0.1
```

Если тест успешно пройден, то вы получите следующий ответ:

```
Ответ от 127.0.0.1: число байт=32 время<1мс TTL=128
```

```
Ответ от 127.0.0.1: число байт=32 время<1мс TTL=128
```

```
Ответ от 127.0.0.1: число байт=32 время<1мс TTL=128
```

```
Ответ от 127.0.0.1: число байт=32 время<1мс TTL=128
```

2) Чтобы убедиться в том, что компьютер правильно добавлен в сеть и IP-адрес не дублируется, используется IP-адрес локального компьютера:

```
ping IP-адрес_локального_хоста
```

3) Чтобы проверить, что шлюз по умолчанию функционирует и что можно установить соединение с любым локальным хостом в локальной сети, задается IP-адрес шлюза по умолчанию:

```
ping IP-адрес_шлюза
```

4) Для проверки возможности установления соединения через маршрутизатор в команде ping задается IP-адрес удаленного хоста:

```
ping IP-адрес_удаленного_хоста
```

**Синтаксис:**

ping [-t] [-a] [-n count] [-l length] [-f] [-i ttl] [-v tos] [-r count] [-s count] [ [-j host-list] | [-k host-list] ] [-w timeout] destination-list

### **Параметры:**

- t выполняет команду ping до прерывания. Control-Break - посмотреть статистику и продолжить. Control-C - прервать выполнение команды;
- a позволяет определить доменное имя удаленного компьютера по его IP-адресу;
- n count посылает количество пакетов ECHO, указанное параметром count;
- l length посылает пакеты длиной length байт (максимальная длина 8192 байта);
- f посылает пакет с установленным флагом «не фрагментировать». Этот пакет не будет фрагментироваться на маршрутизаторах по пути своего следования;
- i ttl устанавливает время жизни пакета в величину ttl (каждый маршрутизатор уменьшает ttl на единицу);
- v tos устанавливает тип поля «сервис» в величину tos;
- r count записывает путь выходящего пакета и возвращающегося пакета в поле записи пути. Count - от 1 до 9 хостов;
- s count позволяет ограничить количество переходов из одной подсети в другую (хопов). Count задает максимально возможное количество хопов;
- j host-list направляет пакеты с помощью списка хостов, определенного параметром host-list. Последовательные хосты могут быть отделены промежуточными маршрутизаторами (гибкая статическая маршрутизация). Максимальное количество хостов в списке, позволенное IP, равно 9;
- k host-list направляет пакеты через список хостов, определенный в host-list. Последовательные хосты не могут быть разделены промежуточными маршрутизаторами (жесткая статическая маршрутизация). Максимальное количество хостов – 9;
- w timeout указывает время ожидания (timeout) ответа от удаленного хоста в миллисекундах (по умолчанию – 1сек);
- destination-list указывает удаленный хост, к которому надо направить пакеты ping.

### **Пример использования утилиты ping:**

C:\WINDOWS>ping -n 10 www.netscape.com

Обмен пакетами с www.netscape.com [205.188.247.65] по 32 байт:

Ответ от 205.188.247.65: число байт=32 время=194мс TTL=48

Ответ от 205.188.247.65: число байт=32 время=240мс TTL=48

Ответ от 205.188.247.65: число байт=32 время=173мс TTL=48

Ответ от 205.188.247.65: число байт=32 время=250мс TTL=48

Ответ от 205.188.247.65: число байт=32 время=187мс TTL=48

Ответ от 205.188.247.65: число байт=32 время=239мс TTL=48  
Ответ от 205.188.247.65: число байт=32 время=263мс TTL=48  
Ответ от 205.188.247.65: число байт=32 время=230мс TTL=48  
Ответ от 205.188.247.65: число байт=32 время=185мс TTL=48  
Ответ от 205.188.247.65: число байт=32 время=406мс TTL=48  
Статистика Ping для 205.188.247.65:

Пакетов: послано = 10, получено = 10, потеряно = 0 (0% потерь)

Приблизительное время передачи и приема:

Наименьшее = 173мс, наибольшее = 406мс, среднее = 236мс

В случае невозможности проверить доступность хоста утилита выводит информацию об ошибке. Ниже приведен пример ответа утилиты ping при попытке послать запрос на несуществующий хост.

Обмен пакетами с 172.16.6.21 по 32 байт:

Превышен интервал ожидания для запроса.

Превышен интервал ожидания для запроса.

Превышен интервал ожидания для запроса.

Превышен интервал ожидания для запроса.

Статистика Ping для 172.16.6.21:

Пакетов: отправлено = 4, получено = 0, потеряно = 4 (100% потерь),

Приблизительное время передачи и приема:

наименьшее = 0мс, наибольшее = 0мс, среднее = 0мс

Утилита сообщает не об отсутствии хоста, а о том, что за отведенное время не был получен ответ на посланный запрос. Причиной этого не обязательно является отсутствие хоста в сети. Проблема может крыться в сбоях связи, перегрузке или неправильной настройке маршрутизаторов и т. п. Ошибка «сеть недоступна» (network unreachable) прямо указывает на проблемы маршрутизации.

### **3. Изучение маршрута между сетевыми соединениями с помощью утилиты *tracert*.**

Tracert - это утилита трассировки маршрута. Она использует поле TTL (time-to-live, время жизни) пакета IP и сообщения об ошибках ICMP для определения маршрута от одного хоста до другого.

Утилита tracert может быть более содержательной и удобной, чем ping, особенно в тех случаях, когда удаленный хост недостижим. С помощью нее можно определить район проблем со связью (у Internet-провайдера, в опорной сети, в сети удаленного хоста) по тому, насколько далеко будет отслежен маршрут. Если возникли проблемы, то утилита выводит на экран звездочки (\*), либо сообщения типа «Destination net unreachable», «Destination host unreachable», «Request time out», «Time Exceeded».

Утилита tracert работает следующим образом: посылаются по 3 пробных эхо-пакета на каждый хост, через который проходит маршрут до удаленного

хоста. На экран при этом выводится время ожидания ответа на каждый пакет (Его можно изменить с помощью параметра -w). Пакеты посылаются с различными величинами времени жизни. Каждый маршрутизатор, встречающийся по пути, перед перенаправлением пакета уменьшает величину TTL на единицу. Таким образом, время жизни является счетчиком точек промежуточной доставки (хопов). Когда время жизни пакета достигнет нуля, предполагается, что маршрутизатор пошлет в компьютер-источник сообщение ICMP "Time Exceeded" (Время истекло). Маршрут определяется путем отправки первого эхо-пакета с TTL=1. Затем TTL увеличивается на 1 в каждом последующем пакете до тех пор, пока пакет не достигнет удаленного хоста, либо будет достигнута максимально возможная величина TTL (по умолчанию 30, задается с помощью параметра -h).

Маршрут определяется путем изучения сообщений ICMP, которые присылаются обратно промежуточными маршрутизаторами.

Примечание: некоторые маршрутизаторы просто уничтожают пакеты с истекшим TTL и не будут видны утилите tracert.

**Синтаксис:**

```
tracert [-d] [-h maximum_hops] [-j host-list] [-w timeout]
имя_целевого_хоста
```

**Параметры:**

-d указывает, что не нужно распознавать адреса для имен хостов;

-h maximum\_hops указывает максимальное число хопов для того, чтобы искать цель;

-j host-list указывает нежесткую статическую маршрутизацию в соответствии с host-list;

-w timeout указывает, что нужно ожидать ответ на каждый эхо-пакет заданное число мсек.

**4. Утилита arp.**

Основная задача протокола ARP – трансляция IP-адресов в соответствующие локальные адреса. Для этого ARP-протокол использует информацию из ARP-таблицы (ARP-кэша). Если необходимая запись в таблице не найдена, то протокол ARP отправляет широковещательный запрос ко всем компьютерам локальной подсети, пытаясь найти владельца данного IP-адреса. В кэше могут содержаться два типа записей: статические и динамические. Статические записи вводятся вручную и хранятся в кэше постоянно. Динамические записи помещаются в кэш в результате выполнения широковещательных запросов. Для них существует понятие времени жизни. Если в течение определенного времени (по умолчанию 2 мин.) запись не была востребована, то она удаляется из кэша.

**Синтаксис:**

```
arp [-s inet_addr eth_addr] | [-d inet_addr] | [-a]
```

**Параметры:**

-s занесение в кэш статических записей;

-d удаление из кэша записи для определенного IP-адреса;  
-a просмотр содержимого кэша для всех сетевых адаптеров локального компьютера;

inet\_addr - IP-адрес;  
eth\_addr - MAC-адрес.

### 5. Утилита route.

Утилита **route** предназначена для работы с локальной таблицей маршрутизации. Она имеет следующий **синтаксис**:

route [-f] [-p] [команда [узел] [MASK маска] [шлюз] [METRIC метрика] [IF интерфейс]]

#### Параметры:

-f Очистка таблицы маршрутизации.  
-p При указании совместно с командой ADD создает постоянную запись, которая сохраняется после перезагрузки компьютера. По умолчанию записи таблицы маршрутов не сохраняются при перезагрузке.

*команда* одна из четырех команд:

PRINT - вывод информации о маршруте;

ADD - добавление маршрута;

DELETE - удаление маршрута;

CHANGE - изменение маршрута.

*узел* адресуемый узел

*маска* маска подсети; по умолчанию используется маска  
255.255.255.255

*шлюз* адрес шлюза

*метрика* метрика маршрута;

*интерфейс* идентификатор интерфейса, который будет использован для пересылки пакета

Для команд PRINT и DELETE возможно использование символов подстановки при указании адресуемого узла или шлюза. Параметр шлюза для этих команд может быть опущен. При добавлении и изменении маршрутов утилита route осуществляет проверку введенной информации на соответствие условию (УЗЕЛ & МАСКА) == УЗЕЛ. Если это условие не выполняется, то утилита выдает сообщение об ошибке и не добавляет или не изменяет маршрут.

Утилита осуществляет поиск имен сетей в файле networks. Поиск имен шлюзов осуществляется в файле hosts. Оба файла расположены в папке %systemroot%\system32\drivers\etc. Наличие и заполнение этих файлов не обязательно для нормального функционирования утилиты route и работы маршрутизации.

Хотя в большинстве случаев на рабочей станции это не требуется, можно вручную редактировать таблицы маршрутизации.

#### **Пример использования утилиты route:**



Добавление статического маршрута:  
route add 172.16.6.0 MASK 255.255.255.0 172.16.11.1 METRIC 1 IF 0x1000003

## 6. Утилита *netstat*.

Утилита *netstat* позволяет получить статическую информацию по некоторым из протоколов стека (TCP, UDP, IP, ICMP), а также выводит сведения о текущих сетевых соединениях. Особенно она полезна на брандмауэрах, с ее помощью можно обнаружить нарушения безопасности периметра сети.

### Синтаксис:

```
netstat [-a] [-e] [-n] [-s] [-p protocol] [-r]
```

### Параметры:

-a выводит перечень всех сетевых соединений и прослушиваемых портов локального компьютера;

-e выводит статистику для Ethernet-интерфейсов (например, количество полученных и отправленных байт);

-n выводит информацию по всем текущим соединениям (например, TCP) для всех сетевых интерфейсов локального компьютера. Для каждого соединения выводится информация об IP-адресах локального и удаленного интерфейсов вместе с номерами используемых портов;

-s выводит статистическую информацию для протоколов UDP, TCP, ICMP, IP. Ключ «/more» позволяет просмотреть информацию постранично;

-r выводит содержимое таблицы маршрутизации.

## 7. Утилита *nslookup*.

Утилита *nslookup* предназначена для диагностики службы DNS, в простейшем случае - для выполнения запросов к DNS-серверам на разрешение имен в IP-адреса. В общем случае утилита позволяет просмотреть любые записи DNS-сервера:

*A* – каноническое имя узла, устанавливает соответствие доменного имени ip-адресу.

*SOA* – начало полномочий, начальная запись, единственная для зоны;

*MX* – почтовые серверы (хосты, принимающие почту для заданного домена);

*NS* – серверы имен (содержит авторитетные DNS-серверы для зоны);

*PTR* – указатель (служит для обратного преобразования ip-адреса в символьное имя хоста)

и т. д.

Утилита *nslookup* достаточно сложна и содержит свой собственный командный интерпретатор.

В простейшем случае (без входа в командный режим) утилита *nslookup* имеет следующий

### Синтаксис:

```
nslookup хост [сервер]
```

### Параметры:

*Хост* DNS-имя хоста, которое должно быть преобразовано в IP-адрес.

*Сервер* Адрес DNS-сервера, который будет использоваться для разрешения имени. Если этот параметр опущен, то будут последовательно использованы адреса DNS-серверов из параметров настройки протокола TCP/IP.

***Примеры использования утилиты nslookup:***

1. Получение списка серверов имен для домена yandex.ru без входа в командный режим (с использованием ключей).

```
C:\> nslookup -type=ns yandex.ru
Server: dns01.catv.ext.ru
Address: 217.10.44.35
Non-authoritative answer:
yandex.ru    nameserver = ns4.yandex.ru
yandex.ru    nameserver = ns5.yandex.ru
yandex.ru    nameserver = ns2.yandex.ru
yandex.ru    nameserver = ns1.yandex.ru

ns2.yandex.ru internet address = 213.180.199.34
ns5.yandex.ru internet address = 213.180.204.1
```

2. Получение записи SOA домена yandex.ru с авторитетного сервера с использование командного интерпретатора nslookup.

```
C:\>nslookup
Default Server: dns04.catv.ext.ru
Address: 217.10.39.4
> set type=SOA
> server ns2.yandex.ru
Default Server: ns2.yandex.ru
Address: 213.180.199.34
> yandex.ru
Server: ns1.yandex.ru
Address: 213.180.193.1
>yandex.ru
    primary name server = ns1.yandex.ru
    responsible mail addr = sysadmin.yandex-team.r
    serial = 2009022707
    refresh = 1800 (30 mins)
    retry = 900 (15 mins)
    expire = 2592000 (30 days)
    default TTL = 900 (15 mins)
yandex.ru    nameserver = ns5.yandex.ru
yandex.ru    nameserver = ns1.yandex.ru
yandex.ru    nameserver = ns4.yandex.ru
```

```
yandex.ru    nameserver = ns2.yandex.ru
ns1.yandex.ru internet address = 213.180.193.1
ns2.yandex.ru internet address = 213.180.199.34
ns4.yandex.ru internet address = 77.88.19.60
ns5.yandex.ru internet address = 213.180.204.1
> exit
```

3. Получение адреса почтового сервера для домена yandex.ru.

```
C:\>nslookup
```

```
Default Server: dns01.catv.ext.ru
```

```
Address: 217.10.44.35
```

```
> set q=mx
```

```
> yandex.ru
```

```
Server: dns01.catv.ext.ru
```

```
Address: 217.10.44.35
```

```
Non-authoritative answer:
```

```
yandex.ru    MX preference = 10, mail exchanger = mx2.yandex.ru
```

```
yandex.ru    MX preference = 10, mail exchanger = mx3.yandex.ru
```

```
yandex.ru    MX preference = 10, mail exchanger = mx1.yandex.ru
```

```
yandex.ru    nameserver = ns2.yandex.ru
```

```
yandex.ru    nameserver = ns1.yandex.ru
```

```
yandex.ru    nameserver = ns4.yandex.ru
```

```
yandex.ru    nameserver = ns5.yandex.ru
```

```
mx1.yandex.ru internet address = 77.88.21.89
```

```
mx2.yandex.ru internet address = 93.158.134.89
```

```
mx3.yandex.ru internet address = 213.180.204.89
```

```
ns2.yandex.ru internet address = 213.180.199.34
```

```
ns4.yandex.ru internet address = 77.88.19.60
```

```
ns5.yandex.ru internet address = 213.180.204.1
```

```
>
```

Указав ключ type=any, можно получить все записи о узле или домене.

Ключи querytype, t, q эквивалентны type.

### **Задания на лабораторную работу**

1. Изучите методические указания к лабораторной работе.
2. Выполните упражнения.
3. Оформите отчет по лабораторной работе, описав выполнение упражнений и дав краткие ответы на контрольные вопросы.

### **Упражнение 1. Получение справочной информации по командам.**

Выведите на экран справочную информацию по всем рассмотренным утилитам (см. таблицу п.1). Для этого в командной строке введите имя утилиты без параметров и дополните /?.

Сохраните справочную информацию в отдельном файле.

Изучите ключи, используемые при запуске утилит.

## Упражнение 2. Получение имени хоста.

Выведите на экран имя локального хоста с помощью команды `hostname`. Сохраните результат в отдельном файле.

## Упражнение 3. Изучение утилиты `ipconfig`.

Проверьте конфигурацию TCP/IP с помощью утилиты `ipconfig`. Заполните таблицу:

Имя хоста	
IP-адрес	
Маска подсети	
Основной шлюз	
Используется ли DHCP (адрес DHCP-сервера)	
Описание адаптера	
Физический адрес сетевого адаптера	
Адрес DNS-сервера	
Адрес WINS-сервера	

## Упражнение 4. Тестирование связи с помощью утилиты `ping`.

1. Проверьте правильность установки и конфигурирования TCP/IP на локальном компьютере.
2. Проверьте функционирование основного шлюза, послав 5 эхо-пакетов длиной 64 байта.
3. Проверьте возможность установления соединения с удаленным хостом.
4. С помощью команды `ping` проверьте адреса (взять из списка локальных ресурсов на сайте [aspu.ru](http://aspu.ru)) и для каждого из них отметьте время отклика. Попробуйте изменить параметры команды `ping` таким образом, чтобы увеличилось время отклика. Определите IP-адреса узлов.

## Упражнение 5. Определение пути IP-пакета.

С помощью команды `tracert` проверьте для перечисленных ниже адресов, через какие промежуточные узлы идет сигнал. Изучите ключи команды.

- a) `gief.ru`
- b) `znanium.com`
- c) `mail.ru`

## Упражнение 6: Просмотр ARP-кэша.

С помощью утилиты `arp` просмотрите ARP-таблицу локального компьютера.

Внести в кэш локального компьютера любую статическую запись.

### **Упражнение 7: Просмотр локальной таблицы маршрутизации.**

С помощью утилиты route просмотреть локальную таблицу маршрутизации.

### **Упражнение 8. Получение информации о текущих сетевых соединениях и протоколах стека TCP/IP.**

С помощью утилиты netstat выведите перечень сетевых соединений и статистическую информацию для протоколов UDP, TCP, ICMP, IP.

### **Упражнение 9. Получение DNS-информации с помощью nslookup.**

- 1) Узнайте ip-адреса узлов, список которых приводится ниже.  
urait.ru, dzen.ru, book.ru.
- 2) Узнайте авторитетные (компетентные) сервера для этих узлов.
- 3) Получите запись SOA с одного из этих серверов для домена gief.ru.

## **ПРАКТИЧЕСКАЯ РАБОТА №2. «Создание проекта компьютерной сети здания, выбор сетевого оборудования и экономическое обоснование проекта»**

Цель работы: сделать примерный проект сегмента локальной сети, работающей на каждом этаже учебного корпуса.

Содержание работы:

1. Сделать примерную схему этажа учебного корпуса.
2. Определить место размещения сетевого оборудования (коммутаторов) на схеме.
3. Определить маршрут прокладки сетевых кабелей от коммутаторов до аудитории.
4. Выбрать коммутаторы по суммарному количеству компьютеров в помещениях этажа.
5. Рассчитать стоимость коммутаторов.
6. Выбрать тип сетевых кабелей для прокладки.
7. Рассчитать стоимость сетевых кабелей и разъемов для подключения компьютеров к коммутаторам.
8. Рассчитать суммарную стоимость всего оборудования необходимого для создания локальной сети корпуса.
9. Сделать отчёт о проделанной работе.

## РЕКОМЕНДАЦИИ ПО ОФОРМЛЕНИЮ ОТЧЁТА ПО ЛАБОРАТОРНЫМ И ПРАКТИЧЕСКИМ РАБОТАМ

### Содержание отчёта:

1. Титульный лист (приложение).
2. Содержание.
3. Цель работы.
4. Основная часть должна содержать задание, программный код, «скриншот» экрана с результатом выполнения работы.
5. Выводы по проделанной работе.
6. Список использованных источников.

Шрифт 14 пт (Times New Roman), с межстрочным интервалом – 1,5. Требования к полям: левое – 30 мм, правое – 10 мм, верхнее – 20 мм, нижнее – 20 мм. Выравнивание по ширине страницы. Каждый раздел, а также выводы и предложения начинаются с новой страницы. Точку в конце заголовка, располагаемого по ширине с абзацного отступа строки, не ставят. Не рекомендуется подчеркивать заголовки. Не допускается переносить часть слова в заголовке. Абзацы начинаются с новой (красной) строки, их печатают с отступом, равным 1,25 см. Каждый рисунок должен сопровождаться содержательной подписью, которая печатается под рисунком по центру страницы в одну строку с номером. Таблица должна иметь заголовок, помещаемый под словом «Таблица» над соответствующей таблицей. Слово «Таблица» и заголовок начинаются с прописной буквы с абзацного отступа и не подчеркиваются. Заголовки граф таблиц должны начинаться с прописных букв; подзаголовки со строчных, если они составляют одно предложение с заголовком, и с прописных, если они самостоятельные. При переносе таблицы на следующую страницу шапку таблицы следует повторить, и над ней помещают слова «Продолжение таблицы» с указанием ее номера. Если заголовок таблицы громоздкий, допускается его не повторять; в этом случае пронумеровывают графы и повторяют их нумерацию на следующей странице. Не допускается оставлять в таблице пустые графы. В этом случае либо ставится прочерк, либо пишется «нет данных». Таблицы следует нумеровать арабскими цифрами порядковой нумерацией в пределах всей работы. При оформлении таблицы в левом верхнем углу начиная с красной строки с прописной буквы пишут слово «Таблица», далее ставится ее номер и без абзацного отступа через тире пишется заголовок таблицы. Заголовок (название таблицы) следует писать с прописной буквы, без точки в конце.

**ВОПРОСЫ К ЭКЗАМЕНУ ПО ДИСЦИПЛИНЕ  
«ВЫЧИСЛИТЕЛЬНЫЕ СИСТЕМЫ, СЕТИ И КОММУНИКАЦИИ»  
ДЛЯ СТУДЕНТОВ ПО НАПРАВЛЕНИЮ ПОДГОТОВКИ –  
38. 03. 05 «БИЗНЕС-ИНФОРМАТИКА»**

1. Понятие канала связи. Какие бывают каналы связи и среды передачи.
2. Основные этапы развития ЭВМ и его программного обеспечения.  
Поколения ЭВМ.
3. Центральный процессор ЭВМ. Структура базового МП.
4. Локальные вычислительные сети.
5. Типы ЛВС.
6. Передающие среды ЛВС
7. Понятие канала связи. Какие бывают каналы связи и среды передачи.
8. Среда передачи данных – виды, параметры, области применения, история развития
9. Понятие интерфейса. Примеры интерфейсов. Что лежит в основе понятия интерфейс.
10. Функциональная и структурная организация ПК с магистральной архитектурой.
11. Архитектуры и топологии ЛВС. Краткий сравнительный анализ различных архитектур и топологий.
12. Правила прокладки сетей на основе витой пары. Категории витой пары в зависимости от параметров передачи.
13. Понятие одноранговой сети.
14. Адресация компьютеров в одноранговой сети. Протоколы, используемые в одноранговых сетях.
15. Структурированная кабельная сеть. Определение через свойства
16. Виды сетевого оборудования, Структурные схемы их устройства, основные параметры и способы подключения к ЛВС
17. Структуры пакетов протоколов IP. Свойства протокола IP. Когда его используют
18. Стек протоколов Интернет и базовые протоколы Интернет
19. Маршрутизатор: назначение, классификация, принцип работы, параметры
20. Серверы и рабочие станции – роль в сети, виды серверов, подключение серверов к сети
21. Способы соединения двух компьютеров. Схема прохождения сообщения от программы-передатчика к программе-приемнику
22. Понятие протокола. Модель OSI. Взаимодействие двух узлов сети в терминах модели OSI.
23. Подключение локальной сети к Интернет: способы и параметры подключения.
24. Топологическая схема одноранговой сети. Используемые среды передачи информации и сетевое оборудование

25. Параметры USB-порта в современном стандарте. Возможность построения сети на основе USB-соединений (топология, кол-во узлов, масштабы)
26. Что такое интернет-шлюзы. Виды, параметры, способы настройки
27. стек протоколов TCP/IP. Описание процесса взаимодействия протоколов
28. IP-адрес. Структура адреса и ее связь со структуризацией сетей. Понятие маски адреса
29. Клиент-серверное взаимодействие в сети. Когда используется.
30. Структуры пакетов протоколов UDP. Свойства протокола UDP. Когда его используют.
31. Разбиение сети на логические подсети с помощью настройки IP-адресов.
32. Понятие маршрута и маршрутизации Принцип маршрутизации, при исполнении какого протокола реализуется маршрутизация, какими техническими средствами реализуется маршрутизация.
33. Принципы маршрутизации, на каком уровне ISO решается задача маршрутизации и какими средствами.
34. Мультиплексирование и демупльтиплексирование.
35. Коммутация пакетов и коммутация каналов.
36. Виды каналов связи.

## **ПЕРЕЧЕНЬ ОСНОВНОЙ И ДОПОЛНИТЕЛЬНОЙ УЧЕБНОЙ ЛИТЕРАТУРЫ ПО ДИСЦИПЛИНЕ**

### **Основная:**

1. Гудыно, Л. П., Вычислительные системы, сети и телекоммуникации : учебное пособие / Л. П. Гудыно, А. А. Кириченко, ; под ред. А. П. Пятибрatова. — Москва : КноРус, 2022. — 372 с. — ISBN 978-5-406-09484-6. — URL: <https://book.ru/book/943143> (дата обращения: 08.09.2023). — Текст : электронный.
2. Микитенко, И. И. Вычислительные машины, сети и системы : лабораторный практикум / И. И. Микитенко. - Москва : Издательский Дом НИТУ «МИСиС», 2022. - 72 с. - Текст : электронный. - URL: <https://znanium.com/catalog/product/1914815> (дата обращения: 08.09.2023). – Режим доступа: по подписке.
3. Тарасов, В. Н. Основы проектирования и моделирования вычислительных сетей : учебное пособие / В. Н. Тарасов, Н.Ф. Бахарева. - Москва ; Вологда : Инфра-Инженерия, 2023. - 212 с. - ISBN 978-5-9729-1175-2. - Текст : электронный. - URL: <https://znanium.com/catalog/product/2096156> (дата обращения: 08.09.2023). – Режим доступа: по подписке
4. Чекмарев, Ю. В. Локальные вычислительные сети : учебное пособие / Ю. В. Чекмарев. - 3-е изд. - Москва : ДМК Пресс, 2023. - 201 с. - ISBN 978-5-89818-470-4. - Текст : электронный. - URL:



<https://znanium.com/catalog/product/2106240> (дата обращения: 08.09.2023). – Режим доступа: по подписке.

5. Шевченко, В. П., Вычислительные системы, сети и телекоммуникации : учебник / В. П. Шевченко. — Москва : КноРус, 2023. — 288 с. — ISBN 978-5-406-10963-2. — URL: <https://book.ru/book/947263> (дата обращения: 08.09.2023). — Текст : электронный.

#### **Дополнительная литература:**

1. . Вычислительные системы, сети и телекоммуникации: учебное пособие / А.П. Пятибратов под ред., Л.П. Гудыно, А.А. Кириченко. — Москва :КноРус, 2017. — 372 с. — Для бакалавров. <https://www.book.ru/book/920283>
2. Безручко В. Т. Информатика (курс лекций): Учебное пособие / В.Т. Безручко. - М.: ИД ФОРУМ: НИЦ ИНФРА-М, 2014. - 432 с.: ил. - (Высшее образование) <http://znanium.com/bookread2.php?book=429099>
3. Бройдо В.Л. Вычислительные системы, сети и телекоммуникации : учеб.пособие / В. Л. Бройдо, О. П. Ильина. - 4-е изд. - СПб. : ПИТЕР, 2011. - 560 с. - (Учебник для вузов). 4 экз.
4. Гусева, А.И. Вычислительные системы, сети и телекоммуникации : учебник / А. И. Гусева, В. С. Киреев. - М. : Академия, 2014. - 288 с. - (Высшее профессиональное образование. Бакалавриат). 2 экз.
5. Компьютерные науки. Деревья, операционные системы, сети / И.Ф. Астахова, И.К. Астанин, И.Б. Крыжко. [Электронный ресурс] - М.: ФИЗМАТЛИТ, 2013. - 88 с.
6. Кузнецов А. С. Теория вычислительных процессов: учебник/Кузнецов А.С., Царев Р.Ю., Князьков А.Н. - Краснояр.: СФУ, 2015. - 184 с. <http://znanium.com/bookread2.php?book=549796>