***На правах рукописи***

Минобрнауки России

Федеральное государственное бюджетное образовательное учреждение

высшего образования

**«Оренбургский государственный университет»**

Кафедра программного обеспечения вычислительной техники и автоматизированных систем

Методические указания для обучающихся по освоению дисциплины

*«Б1.Д.В.Э.1.2 Разработка сетевых клиентских приложений»*

Уровень высшего образования

МАГИСТРАТУРА

Направление подготовки

*09.04.04 Программная инженерия*

(код и наименование направления подготовки)

*Разработка информационно-телекоммуникационных систем*

(наименование направленности (профиля) образовательной программы)

Квалификация

*Магистр*

Форма обучения

*Очная*

Год набора 2023

Составитель \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Тагирова Л.Ф.

Методические указания рассмотрены и одобрены на заседании кафедры программного обеспечения вычислительной техники и автоматизированных систем

Заведующий кафедрой \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Горбачев Д.В.

Методические указания является приложением к рабочей программе по дисциплине Разработка сетевых клиентских приложений, зарегистрированной в ЦИТ под учетным номером \_\_\_\_\_

**Содержание**

|  |  |
| --- | --- |
| Введение........................................................................................................ | 4 |
| 1 Общие рекомендации по изучению дисциплины...................................... | 5 |
| 2 Методические указания по лекционным, практическим и лабораторным занятиям........ | 6 |
| 2.1 Методические указания по лекционным занятиям........................... | 6 |
| 2.2 Методические указания по практическим занятиям ….................... | 6 |
| 2.3 Методические указания по лабораторным занятиям........................ | 7 |
| 3 Методические указания по самостоятельной работе …..………….......... | 8 |
| 3.1 Указания по работе с литературой..................................................... | 8 |
| 4 Методические указания по изучению разделов дисциплины | 10 |
| 5 Методические указания по промежуточной аттестации по дисциплине | 12 |

# Введение

**Целью и задачами** изучения дисциплины Разработка сетевых клиентских приложений” является получение знаний принципов разработки независимых от программной и аппаратной платформы многопоточных сетевых приложений, использующих протоколы TCP и UDP.

**Задачи:**

Студенты должны уметь выбирать подходящие под задачу тип сетевого протокола, модель сетевого ввода-вывода, а также модель клиентского и серверного приложения; разрабатывать сетевое программного обеспечение с использованием выбранных моделей, пользоваться современными средствами разработки приложений.

Рабочая программа дисциплины “Разработка сетевых клиентских приложений” предусматривает контактную работу с преподавателем, которая может включать в себя лекции, практические занятия, лабораторные работы, консультации и промежуточную аттестацию, а также самостоятельную работу студента, которая включает в себя выполнение расчетно-графического задания (РГЗ), самоподготовку (проработка и повторение лекционного материала и материала учебников и учебных пособий, подготовку к практическим и лабораторным занятиям, подготовку к рубежному контролю, диф. зачету).

Цель данных методических рекомендаций – обеспечить студенту бакалавриата оптимальную организацию процесса изучения дисциплины, а также выполнения различных форм самостоятельной работы.

**1 Общие рекомендации по изучению дисциплины**

Перед изучением дисциплины “Разработка сетевых клиентских приложений” студент должен подробно ознакомиться с содержанием рабочей программы дисциплины, с целями и задачами дисциплины, ее связями с другими дисциплинами образовательной программы, с методическими разработками кафедры.

При изучении дисциплины целесообразно руководствоваться следующими общими рекомендациями:

* изучение дисциплины должно вестись систематически и сопровождаться составлением подробного конспекта. В конспект рекомендуется включать все виды учебной работы: лекции, самостоятельную проработку учебника, упражнения, решение задач, лабораторный практикум, ответы на вопросы самопроверки;
* после изучения какого-либо раздела по учебнику или конспекту лекции рекомендуется по памяти записать в тетрадь материал лекции и ответить на вопросы для самопроверки. Такой метод дает возможность проверить усвоение материала;
* после усвоения теории по одной теме нужно разобрать решения задач, относящихся к этой теме, и самостоятельно решить несколько задач. Решение задач, расчетно-графические и контрольные работы способствует лучшему пониманию и закреплению теоретических знаний;
* лабораторные занятия дают возможность непосредственно реализовать примеры, теория которых излагается в учебниках и на лекциях, поэтому студент должен активно участвовать в выполнении всех лабораторных работ;
* простое запоминание теоретического материала недостаточно. При изучении теории следует уделять разбору примеров по созданию веб-приложений;
* следует иметь в виду, что все темы программы являются в равной мере важными. Как и в любой другой науке, нельзя приступать к изучению последующих глав, не усвоив предыдущих. Теоретический материал каждой темы имеет существенное практическое назначение.

**2 Методические указания по лекционным, практическим и лабораторным занятиям**

**2.1 Методические указания по лекционным занятиям**

Лекции по дисциплине дают основной теоретический материал, являющийся базой для восприятия практического материала.

Материал лекций содержит семь тем, посвященных разработке программ на языке С++:

1. Введение в сетевое программирование.

2. Введение в архитектуру протоколов TCP/IP.

3. Интерфейсы прикладного программирования WinSock и сокеты UNIX, основные функции сетевого взаимодействия, определенные в стандарте POSIX.

4. Модели сетевого ввода-вывода.

5. Многоадресная и широковещательная рассылка.

6. Многопоточные сетевые приложения.

7. Обзор моделей построения многопоточных серверных приложений.

Перед очередной лекцией необходимо просмотреть по конспекту материал предыдущей лекции. При затруднениях в восприятии материала следует обратиться к рекомендованным литературным источникам или за консультацией к преподавателю. После прослушивания лекции необходимо прочитать соответствующие темы, уяснить основные термины, проблемные вопросы и подходы к их решению, а также рассмотреть дополнительный материал по теме (в т. ч. практический). Лекционный материал следует использовать при подготовке к практическим и семинарским занятиям.

Краткие записи лекций, их конспектирование поможет усвоить учебный материал. Конспект будет полезным тогда, когда записано самое существенное, основное и сделано это самим студентом. Не надо стремиться записать дословно всю лекцию. Такое «конспектирование» принесёт больше вреда, чем пользы. Запись лекций рекомендуется вести по возможности собственными формулировками. Можно запись осуществлять на одной странице, а следующую оставлять для проработки учебного материала самостоятельно в домашних условиях. Конспект лекции лучше подразделять на пункты, параграфы, соблюдая красную строку. Этому в большой степени будут способствовать пункты плана лекции, предложенные преподавателям. Принципиальные места,  
определения, программный код и другое следует сопровождать замечаниями  
«важно», «особо важно», «хорошо запомнить» и т.п. Можно делать это и с  
помощью разноцветных маркеров или ручек.

**2.2 Методические указания по практическим занятиям**

Практические занятия по дисциплине “Разработка сетевых клиентских приложений” способствуют лучшему усвоению теоретического материала, освоению компетенций, предусмотренных рабочей программой дисциплины, вырабатывают навыки самостоятельной творческой работы, развивают мыслительные способности. Основой практикума выступают типовые задачи, которые должен уметь решать студент, изучающий дисциплину.

Практические занятия проводятся по следующим тематикам:

| № занятия | № раздела | Тема | Кол-во часов |
| --- | --- | --- | --- |
| 1 | 1 | Построение и настройка конфигурации сети в пакете Packet Tracer | 4 |
| 2 | 5 | Многоадресная и широковещательная рассылка | 4 |
| 3 | 6 | Многопоточные сетевые приложения | 6 |
|  |  | Итого: | 14 |

Чтобы подготовиться к практическому занятию, необходимо:

* внимательно прочитать материал лекции по теме практического занятия, выписать необходимые для себя сведения, правила и т. п.;
* составить по лекционному материалу алгоритм, с помощью которого будет проще работать на практическом занятии;
* прочитать материалы учебников (учебных пособий, методических указаний), рекомендуемых к изучаемому разделу, сделать необходимые записи (сведения, которых нет в лекциях).

Решение задач во время аудиторного занятия, а также при самостоятельной работе лучше производить в специально предназначенной для этого рабочей тетради. При этом рекомендуется придерживаться следующих правил:

* оформление отчета, рисунки, графики, схемы алгоритмов выполнять в соответствии с требованиями ГОСТ;
* при оформлении программного кода следует использовать комментарии;
* программы следует разрабатывать с графическим интерфейсом.

На практических занятиях необходимо стремиться к самостоятельному написанию приложений, находя для этого наиболее эффективные методы.

**2.3 Методические указания по лабораторным занятиям**

Необходимые для освоения дисциплины "Программирование и алгоритмизация" навыки приобретаются на лабораторных занятиях и требуют подготовки к ним.

Лабораторные работы проводятся по следующим тематикам:

| № ЛР | № раздела | Наименование лабораторных работ | Кол-во часов |
| --- | --- | --- | --- |
| 1 | 1 | Разработка клиент-серверного приложения типа Chat | 4 |
| 2 | 3 | Интерфейсы прикладного программирования WinSock и сокеты UNIX, основные функции сетевого взаимодействия, определенные в стандарте POSIX | 6 |
| 3 | 4 | Модели сетевого ввода-вывода | 4 |
|  |  | Итого: | 14 |

В ходе подготовки к лабораторным занятиям студент должен:

* изучить теоретический и практический материал предстоящей лабораторной работы;
* разработать программу, согласно заданному варианту;
* продумать порядок выполнения всех пунктов задания.

После выполнения работы необходимо составить отчет. В отчёте приводятся: цель работы, теоретический материал, экранные формы работы программы, листинг, схемы алгоритмов работы программы.

Защита лабораторных работ проводится, в часы, отведенные на лабораторные занятия или по графику консультаций преподавателя.

**3 Методические указания по самостоятельной работе**

**3.1 Указания по работе с литературой**

При самостоятельной работе над учебниками и учебными пособиями рекомендуется придерживаться определенной последовательности. При разработке программ использовать электронные ресурсы, доступные в сети интернет.

После изучения теоретического материала следует решить примеры и задачи по созданию программ, закрепляя тем самым проработанный теоретический материал, а затем приступить к выполнению лабораторных и практических работ.

**4 Методические рекомендации по изучению разделов дисциплины**

**Раздел 1. Введение в сетевое программирование.**

***Содержание раздела.*** Краткая история развития прикладного программного интерфейса сокетов BSD для UNIX, его дальнейшее развитие и текущее состояние в операционных системах семейств Windows и UNIX.

***Рекомендации по изучению раздела.* Д**альнейшее развитие и текущее состояние в операционных системах семейств Windows и UNIX.

**Раздел 2. Введение в архитектуру протоколов TCP/IP.**

***Содержание раздела.*** Соотношение между OSI/ISO и TCP/IP. Межсетевой протокол IP: заголовок IP-сегмента, IP-адрес, фрагментация IP-сегментов, дополнительные данные IP-заголовка. Протокол управления передачей TCP: заголовок TCP-пакета, номер порта, принцип «скользящего окна», важные данные, этапы TCP-взаимодействия, таймеры (таймер повторной передачи, таймер возобновления передачи, таймер закрытия связи, таймеры поддержки соединения),

***Рекомендации по изучению раздела.*** Протокол дейтаграмм пользователя UDP. Межсетевой протокол управляющих сообщений ICMP. Алгоритмы повышения эффективности (задержка подтверждения, исключение малых окон, исключение коротких TCP-пакетов, алгоритм медленного старта).

**Раздел 3. Интерфейсы прикладного программирования WinSock и сокеты UNIX, основные функции сетевого взаимодействия, определенные в стандарте POSIX**

***Содержание раздела.*** Введение в сокеты. Структура адреса сокетов IPv4 и IPv6, универсальная структура адреса сокетов (sockaddr). Функции преобразования порядка байтов, строковых и двоичных представлений адреса и порта. Базовые функции управления TCP-сокетами: создание (socket), удаление (close), локальное связывание (bind), установка соединения с удаленным узлом (connect), перевод в состояние ожидания подключения (listen). Аргументы функций типа «значение-результат». Функции получения свойств и состояния сокетов (getsockopt, setsockopt), ioctl-команды управления сокетом, преобразования имен и адресов (gethostbyname, gethostbyaddr). Основные отличия UDP-сокетов от TCP-сокетов.

***Рекомендации по изучению раздела.*** Функции получения и отправки данных (read, write), векторный ввод-вывод (readv, writev, sendmsg, recvmsg). Получение размера готовых к приему**.** данных, проблема обрезания UDP-дейтаграмм. Методы создания клиентского приложения, использующего соединения TCP и UDP.

**Раздел 4. Модели сетевого ввода-вывода**.

***Содержание раздела.*** Мультиплексирование ввода-вывода (функции select и poll). Блокируемый и неблокируемый ввод-вывод. Ввод-вывод, управляемый сигналом.

***Рекомендации по изучению раздела.*** Асинхронный ввод-вывод. Сравнение моделей ввода-вывода. Создание однопоточного сервера, обслуживающего несколько подключений.

**Раздел 5. Многоадресная и широковещательная рассылка.**

***Содержание раздела.*** Многоадресная передача, адрес многоадресной передачи, область его действия. Многоадресная передача в глобальной сети. Понятие широковещательной группы, функции вхождения и выхода из широковещательной группы (mcast\_join и родственные функции), получение анонсов сеансов многоадресной передачи, отправка и получение данных в широковещательной группе.

***Рекомендации по изучению раздела.*** Создание клиентского приложения, использующего широковещательную и многоадресную рассылку. Сравнение многоадресной и широковещательной передачи в локальной сети.

**Раздел 6.** **Многопоточные сетевые приложения.**

***Содержание раздела.*** Функции стандарта POSIX для управления программными потоками: создание (pthread\_create) и остановка (pthread\_join, pthread\_exit), управление атрибутами и состоянием (группа функций pthread\_attr\_\*). Прикладные системные функции, безопасные в многопоточной среде. Собственные данные потоков.

***Рекомендации по изучению раздела.*** Ситуации взаимоблокировок и примитивы синхронизации: мьютексы и условные переменные. Создание простого многопоточного сетевого приложения.

**Раздел 7. Обзор моделей построения многопоточных серверных приложений.**

***Содержание раздела.*** Однопоточное серверное приложение. Сервер TCP с предварительным порождением процессов без блокировки и вызова accept. Сервер TCP с предварительным порождением процессов и защитой вызова accept блокировкой файла. Сервер TCP с предварительным порождением процессов и защитой вызова accept мьютексом. Сервер TCP с предварительным порождением процессов и передачей дескрипторов.

***Рекомендации по изучению раздела.*** Параллельный сервер TCP с одним потоком для каждого клиента. Сервер TCP с предварительным порождением потоков, каждый из которых вызывает accept. Сервер TCP с предварительным порождением потоков; основной поток вызывает accept. Сравнение моделей.

**5 Методические рекомендации по промежуточной аттестации**

Изучение дисциплины завершается промежуточной аттестации. Учебным планом по дисциплине “Разработка сетевых клиентских приложений” предусмотрен диф. зачет. К промежуточной аттестации допускаются только те студенты, которые сделали и защитили все лабораторные работы, сдали и защитили РГЗ.

Подготовка к промежуточной аттестации способствует закреплению, углублению и обобщению знаний, получаемых, в процессе обучения, а также применению их к решению практических задач. Готовясь к промежуточной аттестации, студент ликвидирует имеющиеся пробелы в знаниях, углубляет, систематизирует и упорядочивает свои знания по программированию. При подготовке к промежуточной аттестации основное направление дают программа учебной дисциплины и студенческий конспект, которые указывают, что наиболее важно знать и уметь делать. Основной материал должен прорабатываться по учебникам и учебным пособиям, так как конспекта далеко недостаточно для изучения дисциплины. Подготовку по каждому разделу следует заканчивать восстановлением по памяти его краткого содержания в логической последовательности.  
 За один - два дня до промежуточной аттестации назначается консультация. Во время консультации студент имеет полную возможность получить ответ на не- ясные ему вопросы. А для этого он должен проработать до консультации все темы дисциплины. Кроме того, преподаватель будет отвечать на вопросы других студентов, что будет для вас повторением и закреплением знаний. Кроме того преподаватель на консультации, как правило, обращает внимание на тс  
вопросы, по которым на предыдущих диф. зачетах ответы были неудовлетворительными, а также фиксирует внимание на наиболее трудных темах дисциплины. Поэтому посещение консультаций обязательно.

Оценка знаний студентов на промежуточной аттестации производится по следующим критериям:

* оценка *«отлично»* выставляется студенту, если он глубоко и прочно усвоил программный материал курса, исчерпывающе, последовательно, четко и логически стройно его излагает, умеет тесно увязывать теорию с практикой, свободно справляется с задачами и вопросами, причем не затрудняется с ответами при видоизменении заданий, правильно обосновывает принятые решения, владеет разносторонними навыками и приемами выполнения практических задач;
* оценка «хорошо» выставляется студенту, если он твердо знает материал курса, грамотно и по существу излагает его, не допуская существенных неточностей в ответе на вопрос, правильно применяет теоретические положения при решении практических вопросов и задач, владеет необходимыми навыками и приемами их выполнения;
* оценка «удовлетворительно» выставляется студенту, если он имеет знания только основного материала, но не усвоил его деталей, допускает неточности, недостаточно правильные формулировки, нарушения логической последовательности в изложении программного материала, испытывает затруднения при выполнении практических задач;
* оценка «неудовлетворительно» выставляется студенту, который не знает значительной части программного материала, допускает существенные ошибки, неуверенно, с большими затруднениями решает практические задачи или не справляется с ними самостоятельно.