

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РФ
Федеральное государственное бюджетное образовательное учреждение высшего
образования
«ДАГЕСТАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
Факультет математики и компьютерных наук

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Современные операционные системы

Кафедра дискретной математики и информатики
факультета математики и компьютерных наук

Образовательная программа

02.04.02 Фундаментальная информатика и информационные технологии

Профиль подготовки

Информационные технологии

Уровень высшего образования

магистратура

Форма обучения

очно-заочная

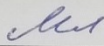
Статус дисциплины:

входит в обязательную часть ОПОП

Рабочая программа дисциплины «Современные операционные системы» составлена в 2021 году в соответствии с требованиями ФГОС ВО по направлению подготовки 02.04.02 Фундаментальная информатика и информационные технологии (уровень магистратуры) от «23» августа 2017 г. № 811.

Разработчик(и): кафедра дискретной математики и информатики, Раджабова Наима Шамильевна, к.ф.-м.н., доцент.

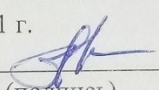
Рабочая программа дисциплины одобрена:
на заседании кафедры дискретной математики и информатики от «30» мая 2021 г., протокол № 9;

зав. кафедрой  Магомедов А. М.
(подпись)

и
на заседании Методической комиссии факультета математики и компьютерных наук от «23» июня 2021 г., протокол № 6;

председатель  Бейбалаев В. Д.
(подпись)

Рабочая программа дисциплины согласована с учебно-методическим управлением
« 09 » 09 2021 г.

Начальник УМУ  Гасангаджиева А. Г.
(подпись)

Аннотация рабочей программы дисциплины

Дисциплина «Современные операционные системы» входит в обязательную часть ОПОП магистратуры по направлению 02.04.02 Фундаментальная информатика и информационные технологии.

Дисциплина реализуется на факультете математики и компьютерных наук кафедрой дискретной математики и информатики.

Содержание дисциплины охватывает круг вопросов, связанных с основами построения современных операционных систем: организацию файловых систем, создание и управление процессами, межпроцессные взаимодействия, параллельное выполнение задач.

Дисциплина нацелена на формирование следующих компетенций выпускника: общепрофессиональных – ОПК-2; профессиональных – ПК-1, ПК-4.

Преподавание дисциплины предусматривает проведение следующих видов учебных занятий: практические, лабораторные занятия.

Рабочая программа дисциплины предусматривает проведение следующих видов контроля успеваемости: в форме контрольной работы, реферата и итогового экзамена в конце семестра.

Объем дисциплины составляет 5 зачетных единиц (180 часов), в том числе в академических часах по видам учебных занятий:

Объем дисциплины в очной форме

Семестр	Учебные занятия							Форма промежуточной аттестации (зачет, дифференцированный зачет, экзамен)	
	в том числе								
	Контактная работа обучающихся с преподавателем						СРС, в том числе экзамен		
	Всего	из них	Лекции	Лабораторные занятия	Практические занятия	КСР			консультации
2	180		10		10			124+36	экзамен

1. Цели освоения дисциплины

Целями освоения дисциплины «Современные операционные системы» являются:

- получение фундаментальных знаний по организации операционных систем, включая изучение таких аспектов, как организацию файловых систем, создание и управление процессами, межпроцессные взаимодействия;
- формирование представлений об общих принципах построения сетевых служб.

Задачи курса:

- ознакомить с вопросами организации ядра операционной системы: управление процессами и ветвями, синхронизация процессов;
- ознакомить с основами организации распределенных файловых систем;
- дать представление о базовых механизмах сетевого взаимодействия: потоки (Streams), связывание со стеком протоколов TCP/IP, программные гнезда (Sockets).

2. Место дисциплины в структуре ОПОП магистратуры

Дисциплина «Современные операционные системы» относится к обязательной части образовательной программы магистратуры по направлению 02.04.02 Фундаментальная информатика и информационные технологии и преподается на 2 курсе в 3 семестре (5 зачетных единиц). Изучение предмета завершается экзаменом в конце семестра.

Дисциплина «Современные операционные системы» логически и содержательно взаимосвязана с дисциплинами, в которых рассматриваются вопросы проектирования программных систем.

Для освоения данной дисциплины необходимо знание основ одного из языков системного программирования: С или Python, освоенных на занятиях по дисциплинам Основы программирования, Языки и методы программирования.

Результаты освоения данной дисциплины будут востребованы на занятиях по «Анализу информационных систем», «Сетевым технологиям». «Технологиям сети Интернет» и в проектно-производственной деятельности магистранта.

3. Компетенции обучающегося, формируемые в результате освоения дисциплины (перечень планируемых результатов обучения).

Код и наименование компетенции из ОПОП	Код и наименование индикатора достижения компетенций (в соответствии с ОПОП)	Планируемые результаты обучения	Процедура освоения
ОПК-2. Способен применять компьютерные/суперкомпьютерные методы, современное программное обеспечение (в том числе отечественного производства) для решения задач профессиональной деятельности	ОПК-2.1. Знает основные положения и концепции в области программирования, архитектуру языков программирования, теории	Знает: международные и российские стандарты жизненного цикла ПО. Умеет: составлять, компилировать и	Конспектирование вопросов практических занятий, подготовка и защита реферата,

	коммуникации, знает основную терминологию, знаком с перечнем ПО, включенного в Единый Реестр Российских программ.	запускать программы в рабочей среде Unix. Владеет: навыками структурного подхода к программированию.	участие в дискуссиях. Разработка и реализация проектов на лабораторных занятиях.
	ОПК-2.2. Умеет анализировать типовые языки программирования, составлять программы.	Знает: международные и российские стандарты жизненного цикла ПО. Умеет: составлять, компилировать и запускать программы в рабочей среде Unix. Владеет: навыками структурного подхода к программированию.	
	ОПК-2.3. Имеет практический опыт решения задач анализа) интеграции различных типов программного обеспечения, анализа типов коммуникации.	Знает: международные и российские стандарты жизненного цикла ПО. Умеет: составлять, компилировать и запускать программы в рабочей среде Unix. Владеет: навыками структурного подхода к	

		программированию.	
ПК-1. Способность понимать и применять в научно-исследовательской и прикладной деятельности современный математический аппарат, основные законы естествознания, современные языки программирования и программное обеспечение; операционные системы и сетевые технологии.	ПК-1.1. Знает основы ведения научной дискуссии и формы устного научного высказывания. Знает: языки программирования и основы архитектуры современных операционных систем для подготовки и представления доклада. Умеет: разрабатывать системные приложения на языке C; отвечать на вопросы по теме представляемого доклада или реферата. Владеет: навыками работы с современными инструментальными средствами (VMware Workstation Pro) для создания и отладки системных приложений.	Знает: языки программирования и основы архитектуры современных операционных систем для подготовки и Конспектирование вопросов практических занятий, подготовка и защита реферата, участие в дискуссиях. Разработка и реализация проектов лабораторных занятий.	
	ПК-1.2. Умеет вести корректную дискуссию в области информационных технологий, задавать вопросы и от-	Знает: языки программирования и основы архитектуры современных операционных систем для подготовки и	

вечать на поставленные вопросы по теме научной работы.	представления доклада. Умеет: разрабатывать системные приложения на языке C; отвечать на вопросы по теме представляемого доклада или реферата. Владеет: навыками работы с современными инструментальным и средствами (VMware Workstation Pro) для создания и отладки системных приложений.
ПК-1.3. Имеет практический опыт владения существующими методами и алгоритмами решения задач цифровой обработки сигналов, использования сети Интернет, аннотирования, реферирования, библиографическо-го разыскания и описания, опыт работы с научными источниками.	Знает: языки программирования и основы архитектуры современных операционных систем для подготовки и представления доклада. Умеет: разрабатывать системные приложения на языке C; отвечать на вопросы по теме представляемого доклада или реферата.

		Владеет: навыками работы с современными инструментальным и средствами (VMware Workstation Pro) для создания и отладки системных приложений.	
ПК-4. Способность применять в профессиональной деятельности современные языки программирования и методы параллельной обработки данных, операционные системы, электронные библиотеки и пакеты программ, сетевые технологии.	ПК-4.1. Знает современные языки программирования и методы параллельной обработки данных.	Знает один из языков программирования Python или C для реализации межпроцессных взаимодействий и методов распараллеливания процессов. Умеет реализовывать проекты, используя системные вызовы Unix и сетевые технологии. Владеет практическим опытом разработки и отладки проектов на основе системных вызовов Unix.	Конспектирование вопросов практических занятий, подготовка и защита реферата, участие в дискуссиях. Разработка и реализация проектов на лабораторных занятиях.
	ПК-4.2. Умеет реализовывать численные методы решения прикладных задач в профессиональной сфере деятельности, пакеты	Знает один из языков программирования Python или C для реализации межпроцессных взаимодействий и методов	

программного обеспечения, операционные системы, электронные библиотеки, сетевые технологии.	распараллеливания процессов. Умеет реализовывать проекты, используя системные вызовы Unix и сетевые технологии. Владеет практическим опытом разработки и отладки проектов на основе системных вызовов Unix.
ПК-4.3. Имеет практический опыт разработки интеграции информационных систем.	Знает один из языков программирования Python или C для реализации межпроцессных взаимодействий и методов распараллеливания процессов. Умеет реализовывать проекты, используя системные вызовы Unix и сетевые технологии. Владеет практическим опытом разработки и отладки проектов на основе системных вызовов Unix.

4. Объем, структура и содержание дисциплины.

4.1. Объем дисциплины составляет 5 зачетных единиц, 180 академических часов.

4.2. Структура дисциплины в очной форме

№ п/п	Разделы и темы дисциплины	Семестр	Неделя	Виды учебной работы, включая самостоятельную работу студентов и трудоемкость (в часах)					Формы текущего контроля успеваемости (по неделям семестра) Форма промежуточной аттестации (по семестрам)
				Лекции	Практические занятия	Лабораторные занятия	Самостоятельна я работа	Контроль сам. работы	
Модуль 1. Принципы создания ОС									
1	Введение в архитектуру UNIX	3	1		2	2	14		Прием лабораторных работ (ЛР) и реферата (Р)
2	Ядро ОС: управление процессами и ветвями	3	2		2	2	14		ЛР, Р
	Итого по модулю 1		36		4	4	28		ЛР, Р
Модуль 2. Создание и управление процессами									
3	Ядро ОС: взаимодействия между процессами	3	3		2	2	14		ЛР, Р
4	Файловая система и средства ввода/вывода.	3	4		2	2	14		
	Итого по модулю 2:		36		4	4	28		Модуль 2
Модуль 3. Организация взаимодействия между процессами									
	Параллельные процессы				2		16		
	Каналы, сигналы, семафоры					2	16		
	Итого по модулю 3:		36		2	2	32		Модуль 3
Модуль 4. Базовые механизмы сетевых взаимодействий									
5	Программный интерфейс сокетов						18		ЛР, Р
6	Удаленный вызов процедур						18		ЛР, Р
	Итого по модулю 4:		36				36		Модуль 4

	Подготовка к экзамену	2					36		Экзамен
	ИТОГО:	2	180		10	10	160		

4.3 Содержание дисциплины, структурированное по темам (разделам)

4.3.1 Содержание лекционных занятий по дисциплине

Лекции по курсу не предусмотрены.

4.3.2 Содержание лабораторно-практических занятий по дисциплине

Лабораторные и практические работы предусмотрены по всем темам модуля.

Целью каждой лабораторной работы является построение системного приложения с использованием соответствующих функций и команд.

Модуль 1. Принципы создания ОС

Практическая работа 1. Введение в архитектуру UNIX

Лабораторная работа 1. Введение в архитектуру UNIX

1. Внутренняя структура ядра
2. Краткая характеристика основных подсистем ядра
3. Основы работы в ОС UNIX.

Модуль 2. Создание и управление процессами

Практическая работа 2. Создание процессов и управление ими.

Лабораторная работа 2. Создание процессов и управление ими.

Практическая работа 3. Ядро ОС: управление процессами и ветвями

Лабораторная работа 3. Ядро ОС: управление процессами и ветвями

1. Типы процессов
2. Жизненный путь процесса
3. Создание процессов и управление ими в UNIX.

Практическая работа 4. Файловая система и средства ввода/вывода

Лабораторная работа 4. Файловая система и средства ввода/вывода

1. Краткий обзор файловых систем UNIX
2. Базовая архитектура драйверов
3. Поток Streams.

Модуль 3. Организация взаимодействия между процессами

Практическая работа 5. Ядро ОС: взаимодействие между процессами

Лабораторная работа 5. Взаимодействие между процессами.

1. Сигналы.
2. Каналы.
3. Сообщения.
4. Семафоры

Модуль 4. Базовые механизмы сетевых взаимодействий

Практическая работа 6. Программный интерфейс сокетов

Лабораторная работа 6. Программный интерфейс сокетов

1. Протоколы TCP/IP
2. Сокеты в клиент-серверном взаимодействии
3. Связывание со стеком протоколов TCP/IP.

Практическая работа 7. Удаленный вызов процедур

Лабораторная работа 7. Удаленный вызов процедур.

1. Описание механизма RPC
2. Связывание
3. Представление данных

Примерная лабораторная работа на тему «Создание процессов и управление ими».

Цель работы: ознакомить с системным вызовом для создания (распараллеливания) процессов.

В *Unix*-системах, *fork()* — системный вызов, создающий новый процесс (потомок), который является практически полной копией процесса-родителя, выполняющего этот вызов.

Задание1. Выясните результат выполнения следующей программы.

```
#include<sys/types.h>
#include <unistd.h>
#include <stdio.h>
int fork(void); void
exit(int status);
main() { int p=fork();
if(p==-1)
    {perror("Error fork");
exit(1);
}
if(p==0)
{
    printf("I am child!\n");
    printf("My process ID is &d\n",getpid());
printf("My parent process ID is &d\n",getppid());
} else
{ printf("I am parent!\n");
    printf("My process ID is %d\n",getpid());
printf("My parent process ID is %d\n",getppid());
```

```
}
}
```

Задание 2. Выясните результат выполнения команд *ps*, *ps -f*, *ps -l*, *pstree* (*pstree -up | less*)

Задание 3. Используя системный вызов *wait()* обеспечить выполнение процесса-потомка до процесса-родителя.

```
#include<sys/types.h>
#include <unistd.h>
#include <sys/wait.h>
int fork(void); void
exit(int status); int
wait(int *status); int
main()
{ int status;
  int p=fork();
  if(p==-1)
  { printf("Error!\n");
    exit(1); }
  If (p==0)
  {
    printf("Hello from child!\n");
    printf("ABCDEFGHIKLMN\n");
  } else
  { wait(&status);
    if
    (WIFEXITED(sta
tus)==0)
    printf("Not
done!\n"); else
    printf("Done!\n");
  } return
  0;
}
```

5. Образовательные технологии

Процесс изложения учебного материала сопровождается презентациями и демонстрацией решения задач в интерактивном режиме с использованием мультимедийного проектора. Предусмотрено регулярное общение с лектором и представителями российских и зарубежных компаний по электронной почте и по скайпу.

6. Учебно-методическое обеспечение самостоятельной работы студентов.

Пакет заданий для самостоятельной работы выдается в начале семестра, определяются предельные сроки их выполнения и сдачи. Самостоятельная работа студентов складывается из проработки учебного материала (настоятельно рекомендуется самостоятельное практическое решение всех разобранных упражнений), видео-лекций и соответствующих форумов интернет, решения всех заданий из индивидуальных заданий, решения рекомендуемых задач, подготовки к сдаче промежуточных отчетов по лабораторным работам и реферата, а также экзамена.

Критерии оценки по приему лабораторных работ.

Если студент владеет по данному модулю навыками решения типичных задач, то по этому модулю ему выставляются:

- 1) 50 баллов;
- 2) 40 баллов в случае наличия неточностей;
- 3) 20 баллов в случае наличия некоторых допустимых ошибок.

Эти баллы учитываются при выводе общего результата как интегральной оценки, складывающейся из текущего контроля и промежуточного контроля.

Примерное распределение времени самостоятельной работы студентов:

Вид самостоятельной работы	Примерная трудоёмкость, а.ч.		
	Очная	Очно-заочная	заочная
Текущая СРС			
работа с лекционным материалом, с учебной литературой	10		
опережающая самостоятельная работа (изучение нового материала до его изложения на занятиях)	5		
самостоятельное изучение разделов дисциплины	5		
подготовка к лабораторным работам и практическим занятиям	10		
подготовка к экзамену	15		
Творческая проблемно-ориентированная СРС			
поиск, изучение и презентация информации по заданной проблеме, анализ научных публикаций по заданной теме	10		
исследовательская работа, участие в конференциях, семинарах, олимпиадах	5		
Итого СРС:	60		

Учебно-методическое обеспечение СРС

№	Вид самостоятельной работы	Вид контроля	Учебно-методическое обеспечение
1.	Изучение рекомендованной литературы и материалов соответствующих форумов интернет.	Контрольный фронтальный опрос, прием и презентация рефератов.	См. разделы 7.3, 8, 9 данного документа

3.	Подготовка к отчетам по лабораторным работам.	Проверка выполнения работ, опрос по теме работы.	См. разделы 7.3, 8, 9 данного документа
4.	Подготовка к сдаче промежуточных форм контроля.	Контрольные работы по каждому модулю и прием рефератов.	См. разделы 7.3, 8, 9 данного документа

Пакет заданий для самостоятельной работы выдается в начале семестра, определяются предельные сроки их выполнения и сдачи.

7. Фонд оценочных средств для проведения текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины.

7.1 Типовые контрольные задания

7.1.1 Вопросы для самостоятельной работы

1. Классификация операционных систем. Различные схемы классификации.
2. Требования к характеристикам операционных систем. Зависимость требований от типа операционных систем.
4. Архитектура UNIX. Двухуровневая модель системы.
5. Работа в операционной системе UNIX. Типы файлов.
6. Структура файловой системы UNIX. Краткое описание основных каталогов.
7. Создание и навигация по файлам и каталогам: *touch, ls, mv, rm, cp, ln, pwd, cd, mkdir, rmdir, find*.
8. Выполнение программ в среде UNIX(создание, компиляция, запуск).
9. Основные системные функции для работы с файлами.
10. Понятие процесса. Типы процессов. Атрибуты процесса.
11. Жизненный путь процесса.
12. Сигналы.
13. Системные вызовы для создания процессов. Элементы создания процессов.

7.1.2 Контрольные вопросы для экзамена

1. Назначение и функции операционных систем (ОС).
2. Ядро ОС: управление процессами и ветвями, синхронизация процессов, обработка прерываний, управление памятью, распределение времени процессора, приоритетное планирование, управление доступом.
3. Стандарты интерфейсов с прикладными программами (POSIX).
4. Файловая система и средства ввода/вывода.
5. Управление вычислительным процессом.
6. Базовые механизмы сетевых взаимодействий.
7. Потoki (Streams).
8. Связывание потоков со стеком протоколов TCP/IP.
9. Программные гнезда (Sockets).

10. Вызовы удаленных процедур.
11. Распределенные файловые системы.
12. Сетевая файловая система.
13. Организация распределенной обработки информации.

7.1.3 Темы для рефератов

1. Структура файловой системы UNIX. Краткое описание основных каталогов. Доступ к файлам
2. Процессы. Типы процессов. Атрибуты процесса. Жизненный путь процесса. Создание процессов. Основы управления процессом. Утилиты UNIX для управления процессами. Состояния процесса.
3. Сигналы. Группы и сеансы. Управление сигналами.
4. Взаимодействие между процессами. (каналы и сообщения).
5. Взаимодействие между процессами. (семафоры и разделяемая память).
6. Пример использования сокетов TCP и UDP.
7. Файловые системы Unix. System V.
8. Файловые системы Unix. BSD(FFS).
9. Подсистема ввода-вывода. Управление передачей данных. Драйверы и модули.
10. Поддержка сети в Unix. Программный интерфейс сокетов.

7.2 Методические материалы, определяющие процедуру оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций.

Общий результат выводится как интегральная оценка, складывающаяся из текущего контроля - 50% и промежуточного контроля - 50%. Текущий контроль по дисциплине включает:

- посещение занятий - 10 баллов,
- подготовка реферата – 20 баллов,
- выполнение лабораторных заданий – 20 баллов,
- выполнение домашних (аудиторных) контрольных работ - 50 баллов.

Промежуточный контроль по дисциплине включает:

- прием лабораторных работ – 50 баллов,
- подготовка реферата – 60 баллов,

8. Перечень основной и дополнительной учебной литературы, необходимой для освоения дисциплины.

а) адрес сайта курса

<https://explorecoursesit.blogspot.com/>

б) основная литература:

- 1.Робачевский, А. М. Операционная система UNIX [Текст] / А. М. Робачевский, С. А. Немнюгин, О. Л. Стесик. — СПб.: БХВ-Петербург, 2010. — 656с.
2. Таненбаум, Э. Современные операционные системы[Текст]/ Э. Таненбаум. — Спб.: Питер, 2011. — 1120 с.
3. Карпов, В. Основы операционных систем : практикум / В. Карпов, К. Коньков. - Москва : Национальный Открытый Университет «ИНТУИТ», 2016. - 301 с. : ил. - Библиогр. в кн. ; То же [Электронный ресурс]. - URL: <http://biblioclub.ru/index.php?page=book&id=429022>
4. Волосатова Т.М. Основные концепции операционной системы UNIX [Электронный ресурс] : учебное пособие / Т.М. Волосатова, С.В. Грошев, С.В. Родионов. — Электрон. текстовые данные. — М. : Московский государственный технический университет имени Н.Э. Баумана, 2010. — 96 с. — 2227-8397. — Режим доступа: <http://www.iprbookshop.ru/31491.html>

в) дополнительная литература:

5. Назаров, Станислав Викторович. Современные операционные системы : учеб. пособие / Назаров, Станислав Викторович, А. И. Широков. - М. : Изд-во Интернет-Ун-та Информ. Технологий: БИНОМ. Лаб. знаний, 2011. - 279 с. - (Основы информационных технологий). - ISBN 978-5-9963-0416-6 : 253-00.
6. Назаров С.В. Современные операционные системы [Электронный ресурс] / С.В. Назаров, А.И. Широков. — Электрон. текстовые данные. — М. : Интернет-Университет Информационных Технологий (ИНТУИТ), 2016. — 351 с. — 978-5-9963-0416-5. — Режим доступа: <http://www.iprbookshop.ru/52176.html>

W. R. Stevens, S. A. Rago, Advanced Programming in the UNIX® Environment: Second Edition, Addison Wesley Professional, 2013.

9.Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины.

- 1) eLIBRARY.RU [Электронный ресурс]: электронная библиотека / Науч. электрон. б-ка. — Москва, 1999 — . Режим доступа: <http://elibrary.ru/defaultx.asp> (дата обращения: 01.04.2021). — Яз. рус., англ.
- 2) Moodle [Электронный ресурс]: система виртуального обучения: [база данных] / Даг. гос. ун-т. — Махачкала, г. — Доступ из сети ДГУ или, после регистрации из сети ун-та, из любой точки, имеющей доступ в интернет. — URL: <http://moodle.dgu.ru/> (дата обращения: 22.05.2021).

3) Электронный каталог НБ ДГУ [Электронный ресурс]: база данных содержит сведения о всех видах лит, поступающих в фонд НБ ДГУ/Дагестанский гос. ун-т. – Махачкала, 2010
– Режим доступа: <http://elib.dgu.ru>, свободный (дата обращения: 21.05.2021).

Видеокурсы лекций:

<https://www.coursera.org/>

1) <https://www.udacity.com/>

2) <https://www.intuit.ru/>

Форумы и блоги по компьютерным наукам и программированию:

1) www.stackoverflow.com

2) <http://www.cyberforum.ru/>

10. Методические указания для обучающихся по освоению дисциплины.

Все упражнения, приведенные на лекции с решениями, следует прорабатывать сразу после лекции. Самостоятельная работа студентов складывается из:

- проработки материала практических занятий (настоятельно рекомендуется самостоятельное практическое решение всех разобранных упражнений);
- изучения рекомендованной литературы и материалов соответствующих форумов интернет;
- подготовки к отчетам по лабораторным работам;
- подготовки к сдаче промежуточных форм контроля (контрольных работ и сдаче реферата).

Пакет лабораторных заданий рассчитан на семестр. Рекомендуется выполнять и сдавать задания своевременно с прохождением соответствующего материала.

Модули и темы для самостоятельного изучения	Виды и содержание самостоятельной работы
Модуль 1. Принципы создания ОС Тема 1. Введение в архитектуру UNIX	Проработка материала практических занятий. Изучение рекомендованной литературы и материалов соответствующих форумов интернет для подготовки реферата.
Модуль 1. Принципы создания ОС Тема 2. Ядро ОС: управление процессами и ветвями	Проработка материала практических занятий. Изучение рекомендованной литературы и материалов соответствующих форумов интернет для подготовки реферата.

Модуль 2. Взаимодействие между процессами Тема 3. Ядро ОС: взаимодействие между процессами	Проработка материала практических занятий. Подготовка к отчетам по лабораторным работам. Подготовка к сдаче промежуточных форм контроля.
Модуль 2. Взаимодействие между процессами Тема 4. Файловая система и средства ввода/вывода	Проработка материала практических занятий. Изучение рекомендованной литературы и материалов соответствующих форумов интернет для подготовки реферата. Подготовка к отчетам по лабораторным работам
Модуль 3. Базовые механизмы сетевых взаимодействий Тема 5. Программный интерфейс сокетов	Проработка материала практических занятий. Изучение рекомендованной литературы и материалов соответствующих форумов интернет для подготовки реферата. Подготовка к отчетам по лабораторным работам.
Модуль 3. Базовые механизмы сетевых взаимодействий Тема 6. Удаленный вызов процедур	Проработка материала практических занятий. Изучение рекомендованной литературы и материалов соответствующих форумов интернет для подготовки отчетов по лабораторным работам.

Все упражнения, приведенные на лекции с решениями, следует прорабатывать сразу после лекции. Самостоятельная работа студентов складывается из:

- проработки материала практических занятий (настоятельно рекомендуется самостоятельное практическое решение всех разобранных упражнений);
- изучения рекомендованной литературы и материалов соответствующих форумов интернет;
- подготовки к отчетам по практическим работам;
- подготовки к сдаче промежуточных форм контроля (контрольных работ и сдаче реферата).

Пакет практических заданий рассчитан на семестр. Рекомендуется выполнять и сдавать задания своевременно с прохождением соответствующего материала.

11. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень программного обеспечения и информационных справочных систем.

При осуществлении образовательного процесса студентами и профессорско-преподавательским составом используются следующее программное обеспечение: Microsoft Windows, Ubuntu Linux, Microsoft Visual Studio Ultimate, Rational Rose, Skype. Также

студентам предоставляется доступ к российским и международным электронным библиотекам через компьютеры университета.

12. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине.

Имеется необходимая литература в библиотеке, медиапроектор и компьютер для проведения лекций-презентаций.

Лабораторные занятия проводятся в компьютерных классах с необходимым программным обеспечением.

Вся основная литература предоставляется студенту в электронном формате.