

Документ подписан простой электронной подписью  
Информация о владельце:  
ФИО: Белгородский Валерий Савельевич  
Должность: Ректор  
Дата подписания: 18.09.2023 11:55:56  
Уникальный программный ключ:  
8df276ee93e17c18e7bee9e7cad2d0ed9ab82473

Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Российский государственный университет им. А.Н. Косыгина  
(Технологии. Дизайн. Искусство)»

Институт химических технологий и промышленной экологии  
Кафедра Автоматизированных систем обработки информации и управления

## РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ

### Системное программное обеспечение

Уровень образования	бакалавриат
Направление подготовки	13.03.01 Теплоэнергетика и теплотехника
Направленность (профиль)	Информационные системы и технологии в топливно-энергетическом комплексе, Промышленная теплоэнергетика
Срок освоения образовательной программы по очной форме обучения	4 года
Форма обучения	очная

Рабочая программа учебной дисциплины «Системное программное обеспечение» основной профессиональной образовательной программы высшего образования, рассмотрена и одобрена на заседании кафедры, протокол № 7 от 15.02.2023 г.

Разработчик рабочей программы учебной дисциплины:

доцент

М.Е. Беспалов

Заведующий кафедрой

В.И. Монахов

## **1. ОБЩИЕ СВЕДЕНИЯ**

Учебная дисциплина «Системное программное обеспечение» изучается в четвертом семестре.

Курсовая работа/Курсовой проект – не предусмотрены.

### **1.1. Форма промежуточной аттестации:**

Зачет с оценкой

### **1.2. Место учебной дисциплины в структуре ОПОП**

Учебная дисциплина «Системное программное обеспечение» относится к обязательной части программы.

Основой для освоения дисциплины являются результаты обучения по предшествующим дисциплинам и практикам:

- Информационные и коммуникационные технологии в профессиональной деятельности;

- Программирование;

Результаты обучения по учебной дисциплине, используются при изучении следующих дисциплин и прохождения практик:

- Информационное моделирование с использованием специализированного программного обеспечения;

- Проектирование автоматизированных систем обработки информации и управления.

## **2. ЦЕЛИ И ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ**

Целями изучения дисциплины «Системное программное обеспечение» являются:

- изучение принципов управления согласованной работой устройств;
- освоение принципов управления ресурсами системы, установки прав на ресурсы и основы администрирования системы;

- овладение настройкой операционной среды системы в целом и личной среды пользователя;

- овладение средствами управления файловыми системами;

- освоение организации резервного копирования и инкрементного архивирования систем;

- изучение основ сетевой файловой системы и протоколов сетевого монтирования ресурсов;

- овладение системой журналов и ведением журнальных файлов;

- овладение регулярными выражениями и их применением для анализа журналов;

- овладение средствами измерения и контроля эффективности системы;

- изучение основ языка Perl и его использования как основного инструмента системного администратора;

- изучение управления пользователями и группами, созданием и удалением учетных записей;

- формирование у обучающихся компетенций, установленных образовательной программой в соответствии с ФГОС ВО по данной дисциплине.

Результатом обучения по учебной дисциплине является овладение обучающимися знаниями, умениями, навыками и опытом деятельности, характеризующими процесс

формирования компетенций и обеспечивающими достижение планируемых результатов освоения учебной дисциплины.

2.1. Формируемые компетенции, индикаторы достижения компетенций, соотнесённые с планируемыми результатами обучения по дисциплине:

Код и наименование компетенции	Код и наименование индикатора достижения компетенции	Планируемые результаты обучения по дисциплине
ОПК-1 Способен понимать принципы работы современных информационных технологий и использовать их для решения задач профессиональной деятельности	ИД-ОПК-1.2 Умение обрабатывать, анализировать и представлять данные с помощью специализированных инструментов и программ	<ul style="list-style-type: none"> <li>– Знает основные понятия операционных систем и назначение всех разновидностей системного программного обеспечения.</li> <li>– Знает основные типовые приёмы автоматизированной обработки текстовой информации средствами операционной системы семейства Unix/Linux .</li> <li>– Умеет по содержательному описанию задач администрирования разрабатывать программы их автоматизации.</li> </ul>
ОПК-2 Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения	ИД-ОПК-2.2 Создание компьютерных программ, алгоритмов или моделей для решения практических задач	<ul style="list-style-type: none"> <li>– Владеет основными приёмами использования типового системного программного обеспечения.</li> <li>– Применяет основные режимы настройки, конфигурирования и применения современных операционных систем.</li> <li>– Применяет навыки программирования средствами командной оболочки (BASH, PowerShell) и языка Perl для реализации типовых задач системного администрирования.</li> <li>– Владеет навыками отладки программ на языках разработки системных сценариев.</li> </ul>
	ИД-ОПК-2.3 Умение объяснять и обосновывать принципы работы созданных алгоритмов и программ	

### 3. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

Общая трудоёмкость учебной дисциплины по учебному плану составляет:

по очной форме обучения –	3	з.е.	108	час.
---------------------------	---	------	-----	------

3.1. Структура учебной дисциплины для обучающихся по видам занятий  
(очная форма обучения)

Структура и объем дисциплины									
Объем дисциплины по семестрам	форма промежуточной аттестации	всего, час	Контактная аудиторная работа, час				Самостоятельная работа обучающегося, час		
			лекции, час	практические занятия, час	лабораторные занятия, час	практическая подготовка, час	<i>курсовая работа/ курсовой проект</i>	самостоятельная работа обучающегося, час	промежуточная аттестация, час
4 семестр	Зачет с оценкой	108	18		26	10		54	
Всего:	Зачет с оценкой	108	18		26	10		54	

## 3.2. Структура учебной дисциплины для обучающихся по разделам и темам дисциплины: (очная форма обучения)

Планируемые (контролируемые) результаты освоения: код(ы) формируемой(ых) компетенции(й) и индикаторов достижения компетенций	Наименование разделов, тем; форма(ы) промежуточной аттестации	Виды учебной работы				Самостоятельная работа, час	Виды и формы контрольных мероприятий, обеспечивающие по совокупности текущий контроль успеваемости; формы промежуточного контроля успеваемости
		Контактная работа					
		Лекции, час	Практические занятия, час	Лабораторные работы/ индивидуальные занятия, час	Практическая подготовка, час		
<b>Четвертый семестр</b>							
ОПК-1: ИД-ОПК-1.2	<b>Раздел I. Соотношение понятий системного программного обеспечения (СПО) и операционной системы (ОС)</b>	6		8		12	Формы текущего контроля по разделу I: 1. письменный отчет о выполнении лабораторных работ. 2. защита лабораторных работ.
ОПК-2: ИД-ОПК-2.2 ИД-ОПК-2.3	Тема 1.1 Понятия системного программного обеспечения (СПО), операционной системы (ОС), операционной среды и операционной оболочки.	2				1	
	Тема 1.2 Основные семейства операционных систем.	2				1	
	Тема 1.3 Файловые подсистемы ОС UNIX/Linux . Подсистема ввода-вывода Простейшие системные средства работы с текстовой информацией в ОС UNIX/Linux .	2				2	
	Лабораторная работа № 1.1 Изучение основных команд ОС UNIX/Linux			2		2	
	Лабораторная работа № 1.2 Построение проектов в ОС UNIX/Linux (знакомство с утилитой GNU make и редактором Vim).			2		2	
	Лабораторная работа № 1.3 Применение команд ОС UNIX/Linux для работы с файловой подсистемой.			2		2	
	Лабораторная работа № 1.4 Построение конвейеров команд ОС UNIX/Linux.			2		2	

Планируемые (контролируемые) результаты освоения: код(ы) формируемой(ых) компетенции(й) и индикаторов достижения компетенций	Наименование разделов, тем; форма(ы) промежуточной аттестации	Виды учебной работы				Самостоятельная работа, час	Виды и формы контрольных мероприятий, обеспечивающие по совокупности текущий контроль успеваемости; формы промежуточного контроля успеваемости
		Контактная работа					
		Лекции, час	Практические занятия, час	Лабораторные работы/индивидуальные занятия, час	Практическая подготовка, час		
ОПК-1: ИД-ОПК-1.2	<b>Раздел II. Системное программное обеспечение для автоматизации обработки текстовой информации в ОС UNIX/Linux</b>	4		8		15	Формы текущего контроля по разделу II: 1. письменный отчет о выполнении лабораторных работ. 2. защита лабораторных работ. 3. контрольная работа.
ОПК-2: ИД-ОПК-2.2 ИД-ОПК-2.3	Тема 2.1 Приёмы работы с текстом средствами системных утилит <code>grep</code> , <code>sed</code> , <code>GNU awk (gawk)</code> .	2				3	
	Тема 2.2 Создание сценариев на языке командного интерпретатора <code>GNU BourneAgain SHell (BASH)</code> .	2				2	
	Лабораторная работа № 2.1 Регулярные выражения и команда <code>grep</code> . Поточковый редактор <code>sed</code> .			2		2	
	Лабораторная работа № 2.2 Анализ и обработка табличных данных с помощью утилиты <code>GNU awk (gawk)</code> .			2		2	
	Лабораторная работа № 2.3 Настройка командной оболочки <code>BASH</code> .			2		2	
	Лабораторная работа № 2.4 Написание сценариев на языке командного интерпретатора <code>BASH</code> .			2		4	
ОПК-1: ИД-ОПК-1.2	<b>Раздел III. Системное программное обеспечение для управления процессами, потоками, памятью.</b>	4		6	4	12	Формы текущего контроля по разделу III: 1. письменный отчет о выполнении лабораторных работ. 2. защита лабораторных работ.
ОПК-2: ИД-ОПК-2.2	Тема 3.1 Управление процессами и потоками в ОС UNIX/Linux.	2					
ИД-ОПК-2.3	Тема 3.2 Управление памятью. Методы, алгоритмы и средства.	2					

Планируемые (контролируемые) результаты освоения: код(ы) формируемой(ых) компетенции(й) и индикаторов достижения компетенций	Наименование разделов, тем; форма(ы) промежуточной аттестации	Виды учебной работы				Самостоятельная работа, час	Виды и формы контрольных мероприятий, обеспечивающие по совокупности текущий контроль успеваемости; формы промежуточного контроля успеваемости
		Контактная работа					
		Лекции, час	Практические занятия, час	Лабораторные работы/индивидуальные занятия, час	Практическая подготовка, час		
	Лабораторная работа № 3.1 Команды ОС UNIX/Linux для управления процессами.			2		4	
	Лабораторная работа № 3.2 Механизмы межпроцессного взаимодействия (сигналы, программные (pipes) и именованные (FIFO) каналы) .			2	2	4	
	Лабораторная работа № 3.3 Управление потоками. Синхронизация потоков на основе семафоров.			2	2	4	
ОПК-1: ИД-ОПК-1.2	<b>Раздел IV. Системное программное обеспечение для системного администрирования ОС UNIX/Linux и ОС Windows.</b>	4		4	6	15	Формы текущего контроля по разделу IV: 1. письменный отчет о выполнении лабораторных работ. 2. защита лабораторных работ. 3. контрольная работа. 4. тестирование.
ОПК-2: ИД-ОПК-2.2	Тема 4.1 ОС Windows и ОС UNIX/Linux изнутри.	4				6	
ИД-ОПК-2.3	Лабораторная работа № 4.1 «Однострочники» на Perl как средство решения задач системного администрирования.			2	2	6	
	Лабораторная работа № 4.2 PowerShell как командная оболочка и среда для создания сценариев в ОС Windows.			2	2	3	
	Зачет с оценкой						Зачет с оценкой
	<b>ИТОГО за четвертый семестр</b>	<b>18</b>		<b>26</b>	<b>10</b>	<b>54</b>	

## 3.3. Краткое содержание учебной дисциплины

№ пп	Наименование раздела и темы дисциплины	Содержание раздела (темы)
<b>Раздел I</b>	<b>Раздел I. Соотношение понятий системного программного обеспечения (СПО) и операционной системы (ОС)</b>	
Тема 1.1	Понятия системного программного обеспечения (СПО) операционной системы (ОС), операционной среды и операционной оболочки.	Функции ОС, используемые в СПО и дополняемые СПО (пользовательский интерфейс операционной среды, управление задачами, управление памятью, управление вводом-выводом).
Тема 1.2	Основные семейства операционных систем.	История семейства ОС UNIX/Linux Генеалогия семейства операционных систем и некоторые известные версии UNIX. Сходства и различия ОС FreeBSD и ОС Linux. Операционные системы фирмы Microsoft Отличия семейства UNIX/Linux от операционных систем Windows и FreeDOS.
Тема 1.3	Файловые подсистемы ОС UNIX/Linux . Подсистема ввода-вывода Простейшие системные средства работы с текстовой информацией в ОС UNIX/Linux .	Структура каталогов ОС UNIX/Linux . Полномочия на доступ к файлам и каталогам. Управление пользователями и их группами. Монтирование и размонтирование файловой системы. Индексные и файловые дескрипторы, целостность файловой системы. Типы и базовая архитектура драйверов, встраивание драйверов в ядро ОС. Фильтры, перенаправление ввода-вывода, конвейеры команд в ОС UNIX/Linux.
<b>Раздел II</b>	<b>Системное программное обеспечение для автоматизации обработки текстовой информации в ОС UNIX/Linux</b>	
Тема 2.1	Приёмы работы с текстом средствами системных утилит grep, sed, GNU awk (gawk).	Команды find и grep в ОС UNIX/Linux как инструмент для администрирования. Команда grep и регулярные выражения . Редактирование конфигурационных файлов или фильтрация вывода средствами утилиты sed. Сравнительное решение задач с помощью утилит sed и GNU awk (gawk).
Тема 2.2	Создание сценариев на языке командного интерпретатора GNU BourneAgain Shell (BASH).	Исполнение командных файлов . Синтаксические особенности BASH как языка программирования сценариев. Манипуляции с переменными. Оператор сравнения test («[ ]»). Условное и циклическое управление. Пользовательские функции. Инструменты отладки программы. Обработка ошибок средствами командной оболочки.
<b>Раздел III</b>	<b>Системное программное обеспечение для управления процессами, потоками, памятью</b>	
Тема 3.1	Управление процессами и потоками в ОС UNIX/Linux.	Создание процессов и потоков. Модели процессов и потоков. Преимущества и недостатки использования потоков. Структуры данных процесса. Состояния процесса. Контекст процесса. Принципы планирования заданий, процессов и потоков. Команды для управления процессами. Взаимодействие и синхронизация процессов и потоков. Межпроцессное взаимодействие в ОС FreeBSD. Сигналы и управление ими . Сокеты. Семфоры и мьютексы.
Тема 3.2	Управление памятью. Методы, алгоритмы и	Функции ОС по управлению памятью Распределение памяти



	средства.	Страничная организация виртуальной памяти Оптимизация функционирования страничной виртуальной памяти Сегментная организация виртуальной памяти Сегментно-страничная виртуальная память
<b>Раздел IV</b>	<b>Системное программное обеспечение для системного администрирования ОС UNIX/Linux и ОС Windows.</b>	
Тема 4.1	ОС Windows и ОС UNIX/Linux изнутри.	Обзор и сравнительная характеристика системного программного обеспечения ОС Windows и ОС UNIX/Linux. Возможности применения языка Perl в качестве инструмента автоматизации задач администрирования операционной системы. Мониторинг производительности ОС Windows с помощью системного монитора perfmon.

### 3.4. Организация самостоятельной работы обучающихся

Самостоятельная работа студента – обязательная часть образовательного процесса, направленная на развитие готовности к профессиональному и личностному самообразованию, на проектирование дальнейшего образовательного маршрута и профессиональной карьеры.

Самостоятельная работа обучающихся по дисциплине организована как совокупность аудиторных и внеаудиторных занятий и работ, обеспечивающих успешное освоение дисциплины.

Аудиторная самостоятельная работа обучающихся по дисциплине выполняется на учебных занятиях под руководством преподавателя и по его заданию. Аудиторная самостоятельная работа обучающихся входит в общий объем времени, отведенного учебным планом на аудиторную работу, и регламентируется расписанием учебных занятий.

Внеаудиторная самостоятельная работа обучающихся – планируемая учебная, научно-исследовательская, практическая работа обучающихся, выполняемая во внеаудиторное время по заданию и при методическом руководстве преподавателя, но без его непосредственного участия, расписанием учебных занятий не регламентируется.

Внеаудиторная самостоятельная работа обучающихся включает в себя:

- подготовку к лабораторным занятиям, экзамену;
- изучение учебных пособий;
- подготовку к выполнению лабораторных работ и отчетов по ним;
- подготовку к контрольной работе
- подготовку к тестированию;
- подготовку к промежуточной аттестации в течение семестра.

Самостоятельная работа обучающихся с участием преподавателя в форме иной контактной работы предусматривает групповую и индивидуальную работу с обучающимися и включает в себя:

- проведение индивидуальных и групповых консультаций по отдельным темам/разделам дисциплины;
- проведение консультаций перед экзаменом.

Перечень разделов/тем/, полностью или частично отнесенных на самостоятельное изучение с последующим контролем:

№ пп	Наименование раздела /темы дисциплины, выносимые на самостоятельное изучение	Задания для самостоятельной работы	Виды и формы контрольных мероприятий (учитываются при проведении текущего контроля)	Трудоемкость, час
1	Сборка ядра ОС Linux из исходного кода	Выполнить и описать процесс сборки клона ОС Linux из исходного кода.	Отчет о выполненной работе.	2
2	Разветвление процессов (форкинг) в ОС UNIX/Linux	Рассмотреть возможность программного управления свойствами фоновых процессов (демонов).	Отчет о выполненной работе.	2

### 3.5. Применение электронного обучения, дистанционных образовательных технологий

Реализация программы учебной дисциплины с применением электронного обучения и дистанционных образовательных технологий регламентируется действующими локальными актами университета.

В электронную образовательную среду могут быть перенесены отдельные виды учебной деятельности:

использование ЭО и ДОТ	использование ЭО и ДОТ	объем, час	включение в учебный процесс
смешанное обучение	лекции	17	в соответствии с расписанием учебных занятий
текущий контроль	тестирование	2	в соответствии с расписанием учебных занятий
Промежуточная аттестация		1	в соответствии с расписанием экзаменов

ЭОР обеспечивают в соответствии с программой дисциплины:

- организацию самостоятельной работы обучающегося, включая контроль знаний обучающегося (самоконтроль, текущий контроль знаний и промежуточную аттестацию),
- методическое сопровождение и дополнительную информационную поддержку электронного обучения (дополнительные учебные и информационно-справочные материалы).

#### 4. РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ, КРИТЕРИИ ОЦЕНКИ УРОВНЯ СФОРМИРОВАННОСТИ КОМПЕТЕНЦИЙ, СИСТЕМА И ШКАЛА ОЦЕНИВАНИЯ

##### 4.1. Соотнесение планируемых результатов обучения с уровнями сформированности компетенций.

Уровни сформированности компетенции(-й)	Итоговое количество баллов в 100-балльной системе по результатам текущей и промежуточной аттестации	Оценка в пятибалльной системе по результатам текущей и промежуточной аттестации	Показатели уровня сформированности		
			универсальной(-ых) компетенции(-й)	общепрофессиональной(-ых) компетенций	профессиональной(-ых) компетенции(-й)
				ОПК-1: ИД-ОПК-1.2 ОПК-2: ИД-ОПК-2.2 ИД-ОПК-2.3	
высокий		зачтено (отлично)		<p>Обучающийся:</p> <ul style="list-style-type: none"> <li>– знает особенности конфигурирования и администрирования операционных систем семейства ОС UNIX/Linux;</li> <li>– умеет оптимизировать сценарии автоматизации типовых задач администрирования операционных систем;</li> <li>– знать назначение и особенности практического применения базовых видов системного программного обеспечения;</li> <li>– уметь использовать современные средства скриптового программирования для решения задач</li> </ul>	

				<p>администрирования операционных систем;</p> <ul style="list-style-type: none"> <li>– владеть навыками отладки программ на скриптовых языках (BASH, Perl).</li> </ul>	
повышенный		зачтено (хорошо)		<p>Обучающийся:</p> <ul style="list-style-type: none"> <li>– знает основные типы системного программного обеспечения;</li> <li>– умеет реализовывать алгоритмы решения задач администрирования современных операционных систем;</li> <li>– знает особенности синтаксиса программирования Perl для решения типовых задач системного администрирования;</li> <li>– владеет навыками отладки программ на языке Perl и языке командного интерпретатора BASH;</li> </ul>	
базовый		зачтено (удовлетворительно)		<p>Обучающийся:</p> <ul style="list-style-type: none"> <li>– знает основные понятия современных операционных систем;</li> <li>– знает основные типы и назначение системного программного обеспечения.</li> </ul>	
низкий		неудовлетворительно/ не зачтено	<p><i>Обучающийся:</i></p> <ul style="list-style-type: none"> <li>– демонстрирует фрагментарные знания теоретического и практического материала;</li> <li>– испытывает серьёзные затруднения в применении теоретических положений при решении практических задач профессиональной направленности стандартного уровня сложности, не владеет необходимыми для этого навыками и приёмами;</li> </ul>		

			<ul style="list-style-type: none"> <li>– не знает основных принципов функционирования и применения системных утилит операционных систем;</li> <li>– не способен самостоятельно реализовать алгоритмы на языке программирования Perl;</li> <li>– выполняет задания только по образцу и под руководством преподавателя;</li> <li>– ответ отражает отсутствие знаний на базовом уровне теоретического и практического материала в объеме, необходимом для дальнейшей учебы.</li> </ul>
--	--	--	---

## **5. ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ТЕКУЩЕГО КОНТРОЛЯ УСПЕВАЕМОСТИ И ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ, ВКЛЮЧАЯ САМОСТОЯТЕЛЬНУЮ РАБОТУ ОБУЧАЮЩИХСЯ**

При проведении контроля самостоятельной работы обучающихся, текущего контроля и промежуточной аттестации по учебной дисциплине «Программирование» проверяется уровень сформированности у обучающихся компетенций и запланированных результатов обучения по дисциплине, указанных в разделе 2 настоящей программы.

### 5.1. Формы текущего контроля успеваемости, примеры типовых заданий:

№ пп	Формы текущего контроля	Примеры типовых заданий
1	Лабораторная работа № 1.1 Изучение основных команд ОС UNIX/Linux	1. Ознакомиться со следующими командами для пользовательской работы в ОС Unix: man, arpopos, cat, cat, cd, cp, ls, ln, mkdir, pwd, mv, rm, rmdir, echo, less, touch, grep, date, history 2. Используя man, определить параметры, которые следует считать основными при использовании данных команд. 3. Определить абсолютный путь своего домашнего каталога. 4. Определить значения следующих переменных окружения: PATH, MANPATH, PAGER. 5. Определить границы файлового пространства, где система позволяет создавать собственные файлы и каталоги (возможно использование автоматического скрипта). 6. Проверить, возможно ли вмешательство в личное файловое пространство другого пользователя.

№ пп	Формы текущего контроля	Примеры типовых заданий
2	Лабораторная работа № 1.2 Построение проектов в ОС UNIX/Linux (знакомство с утилитой GNU make и редактором Vim).	<ol style="list-style-type: none"> <li>1. Используя текстовый редактор Vim, создать простейшую программу на языке C (C++) с использованием, как минимум, двух исходных файлов (с программным кодом).</li> <li>2. Для автоматизации сборки проекта утилитой make создать make-файл (см. п. «Пример создания более сложного make-файла»).</li> <li>3. Выполнить программу (скомпилировать, при необходимости отладить).</li> <li>4. Показать, что при изменении одного исходного файла и последующем вызове make будут исполнены только необходимые команды компиляции (неизмененные файлы перекомпилированы не будут) и изменены атрибуты и/или размер объектных файлов (файлы с расширением .o).</li> <li>5. Создать make-файл с высоким уровнем автоматизированной обработки исходных файлов программы согласно следующим условиям: <ul style="list-style-type: none"> <li>- имя скомпилированной программы (выполняемый или бинарный файл), флаги компиляции и имена каталогов с исходными файлами и бинарными файлами (каталоги src, bin и т. п.) задаются с помощью переменных в makefile;54</li> <li>- зависимости исходных файлов на языке C (C++) и цели в makeфайле должны формироваться динамически;</li> <li>- наличие цели clean, удаляющей временные файлы;</li> <li>- каталог проекта должен быть структурирован следующим образом: <ul style="list-style-type: none"> <li>- src – каталог с исходными файлами;</li> <li>- bin – каталог с бинарными файлами (скомпилированными);</li> <li>- makefile</li> </ul> </li> </ul> </li> </ol>

№ пп	Формы текущего контроля	Примеры типовых заданий
3	Лабораторная работа № 1.3 Применение команд ОС UNIX/Linux для работы с файловой подсистемой.	<ol style="list-style-type: none"> <li>1. Ознакомиться с командами определения и изменения прав доступа к файлам (команды <code>id</code>, <code>groups</code>, <code>ls -l</code>, <code>stat</code>, <code>chmod</code>, <code>chown</code>, <code>chgrp</code>, <code>umask</code>).</li> <li>2. Найти запись в файле <code>/etc/passwd</code>, соответствующую вашему регистрационному имени.</li> <li>3. Определить свой UID, узнать, к каким группам относится ваше регистрационное имя, объяснить вывод команд <code>id</code>, <code>groups</code>.</li> <li>4. Определить список групп, в которые входит пользователь <code>root</code>. Узнать, какими правами доступа обладают вновь создаваемые файлы и каталоги (т. е. создать новый файл и новый каталог, и просмотреть для них права доступа).</li> <li>5. Определить значение <code>umask</code>, при котором создаваемые файлы и каталоги будут недоступны для чтения, записи и исполнения никому, кроме владельца.</li> <li>6. Сделать свой домашний каталог видимым для всех пользователей группы <code>users</code>.</li> <li>7. Создать в домашнем каталоге подкаталог <code>tmp</code>, файлы в котором сможет создавать, удалять и переименовывать любой входящий в группу <code>users</code>, при этом содержимое этого подкаталога не должно быть видимым всем прочим пользователям.</li> </ol>

№ пп	Формы текущего контроля	Примеры типовых заданий
4	Лабораторная работа № 1.4 Построение конвейеров команд ОС UNIX/Linux.	<ol style="list-style-type: none"> <li>1.Создайте конвейер для получения списка только числовых идентификаторов пользователей, работающих в системе.</li> <li>2.Создайте конвейер для получения списка только имен и прав доступа к файлам, которые в данный момент находятся в Вашем рабочем каталоге.</li> <li>3. Измените построенный конвейер так, чтобы список сохранялся в файле spisok Вашего домашнего каталога, а на экран выводилось только число файлов в списке.</li> <li>4. Распечатайте файл .profile столбцами по 20 символов, разделяя на страницы по 20 строк, и сохраните результат в файле example.</li> <li>5. Выведите на экран содержимое файла /etc/passwd, упорядоченное по полю с именем пользователя.</li> <li>6.Выведите на экран содержимое файла /etc/passwd, упорядоченное по третьему полю.</li> <li>7. Выведите на экран все строки файла /etc/passwd, содержащие слово st. Сохраните результат в файле st_users. Подсчитайте количество строк в файле st_users.</li> <li>8. Выведите на экран все строки файла /etc/passwd, не содержащие слово st. Подсчитайте количество таких строк.</li> <li>9. Создайте файл с именем, содержащим две точки, окруженные другими символами (вида prefix.infix.suffix). Выведите на экран полное имя файла, часть имени файла без суффикса (т.е. часть имени до правой точки), имя файла без префикса (т.е. часть имени файла справа от первой точки) и среднюю часть имени файла, заключенную между точками.</li> <li>10. Создайте конвейер, который будет подсчитывать количество пользователей, работающих в системе.</li> <li>11. Измените построенный в предыдущей задаче конвейер так, чтобы он сохранял список пользователей в файле users.</li> <li>12. Создайте конвейер, который выводил бы на экран упорядоченный по алфавиту список зарегистрированных пользователей, содержащий только имена пользователей, их идентификационные номера и имена домашних каталогов.</li> </ol>



№ пп	Формы текущего контроля	Примеры типовых заданий
5	Лабораторная работа № 2.1 Регулярные выражения и команда grep. Поточковый редактор sed.	<p>1. Создать текстовый файл, содержащий набор строк вида: 123 178 176 755 713 873181 С помощью утилиты grep найти строки, в которых есть цифра 7, после которой находится одна из цифр — 1, 3 или 5.</p> <p>2. Создать текстовый файл, содержащий набор строк вида: starfish starless samscripтер stellar microsrar ascender sacrifice scalar С помощью утилиты grep найти строки, начинающиеся на букву s и заканчивающиеся на букву r.</p> <p>3. Создать текстовый файл, содержащий простейшие адреса электронной почты вида username@website.com. С помощью утилиты grep найти строки, содержащие правильные простейшие адреса. Проверить возможность использования более сложного регулярного выражения для распознавания адресов, содержащих другие допустимые символы.</p> <p>4. На произвольном примере продемонстрировать работу утилиты tr Создать текстовый файл, содержащий допустимые и недопустимые IP-адреса, например 127.0.0.1 255.255.255.255 12.34.56 123.256.0.0 1.23.099.255 0.79.378.111</p>

№ пп	Формы текущего контроля	Примеры типовых заданий
		<p>С помощью утилиты grep и руководства man найти строки, содержащие допустимые четырехбайтовые IP адреса.</p> <p>5. Создать текстовый файл, содержащий корректные и некорректные номера телефонов ведомственной АТС объемом 399 номеров, номера с 000 до 399 – корректные, 0, 400, 900 – некорректные.</p> <p>С помощью утилиты grep и руководства man найти строки, содержащие допустимые номера телефонов.</p>
6	<p>Лабораторная работа № 2.2</p> <p>Анализ и обработка табличных данных с помощью утилиты GNU awk (gawk).</p>	<p><b>Вариант 1.</b></p> <p>В заданном текстовом файле первый символ каждой строки заменить на первый символ предыдущей строки. Первая строка остается без изменений.</p> <p>В заданном текстовом файле определить товары, в названии которых фигурирует "BALL", и их общее количество</p> <p>В заданном текстовом файле определить города, в которых есть более одного покупателя</p> <p><b>Вариант 2.</b></p> <p>В заданном текстовом файле перенести первое слово каждой строки в начало следующей строки.</p> <p>В заданном текстовом файле определить разность между максимальной и минимальной зарплатой</p> <p>В заданном текстовом файле определить для каждого штата число покупателей, в названии которых есть "SPORT"</p> <p><b>Вариант 3.</b></p> <p>В заданном текстовом файле после строк, которые заканчиваются точкой или запятой, вставить пустую строку.</p> <p>В заданном текстовом файле определить трех сотрудников, получающих самую высокую зарплату</p> <p>В заданном текстовом файле считая, что первое слово в названии товара - название фирмы, определить фирмы, все товары которых выставлены в продажу в один день</p> <p><b>Вариант 4</b></p> <p>В заданном текстовом файле удалить вторую строку, начинающуюся с буквы 'H'.</p> <p>В заданном текстовом файле определить количество сотрудников на каждой должности</p> <p>В заданном текстовом файле определить отделы, в которых нет менеджеров ("MANAGER")</p>

№ пп	Формы текущего контроля	Примеры типовых заданий
		<p><b>Вариант 5</b>            В заданном текстовом файле поменять местами два первых и два последних символа каждой строки.            В заданном текстовом файле определить города, в которых есть отделы "SALES"            В заданном текстовом файле определить три самых больших заказа.</p>
7	Лабораторная работа № 2.3 Настройка командной оболочки BASH.	<p><b>Вариант 1.</b>            Сравните вывод команд <code>date</code> и <code>LANG=C date</code></p> <p><b>Вариант 2.</b> Измените приглашение командной строки так, чтобы в нем выводилось время, имя пользователя и текущий рабочий каталог. Заканчиваться приглашение должно символом доллара. Сохраните команду определяющую приглашение в файле <code>.bashrc</code>, чтобы приглашение восстанавливалось при каждом запуске оболочки.</p> <p><b>Вариант 3.</b>            Просмотрите значения переменных <code>PPID</code>, <code>PWD</code>, <code>OLDPWD</code>, <code>UID</code>, <code>RANDOM</code> и <code>HOST-NAME</code> и сохраните их в файле отчёта в формате “имя_переменной=значение”.</p> <p><b>Вариант 4.</b>            : Создайте псевдонимы:  <code>L s</code> для команды <code>ls -G</code>  <code>l l</code> для команды <code>ls -l</code>  <code>l a</code> для команды <code>ls -a</code>  <code>ra</code> для команды <code>ps ax</code>  <code>x</code> для команды <code>startx</code>            Сохраните команды, определяющие псевдонимы в файле <code>.bashrc</code>, чтобы они выполнялись каждый раз при запуске оболочки.</p> <p><b>Вариант 5.</b>            : Определите функцию <code>lab</code>, которая будет выводить строку "Лабораторная работа N", где N передаётся функции в качестве аргумента, а затем строку содержащую Ваши имя и фамилию. Сохраните определение функции в файле <code>.bashrc</code>. Добавьте результат работы</p>

№ пп	Формы текущего контроля	Примеры типовых заданий
8	Лабораторная работа № 2.4 Написание сценариев на языке командного интерпретатора BASH	<p>функции в файл отчета.</p> <p>Написать интерактивную программу (скрипт) на языке shell, которая выводит список действий с номерами и ожидает ввода номера пункта, после чего начинает выполнение заданных в этом пункте команд. Скрипт должен корректно обрабатывать ситуацию окончания входного потока данных и позволять работать с относительными и абсолютными именами файлов и каталогов, которые могут содержать: пробелы и символы табуляции; символы '*', '?', '[', ']', '-', '~', '\$' или состоять только из них.</p> <p>Для команд удаления необходимо запрашивать собственное подтверждение дополнительно. Текст запроса должен соответствовать системному (выводимому командой gm). Ожидание ввода ответа происходит на новой строке.</p> <p>Каждый запрос на ввод должен сопровождаться сообщением на отдельной строке, содержащим информацию о том, какие данные должны быть введены. Ожидание ввода происходит с новой строки.</p> <p>Выход из скрипта должен осуществляться только при выборе соответствующего пункта меню или окончании стандартного потока ввода.</p> <p><b>Вариант 1.</b></p> <ul style="list-style-type: none"> <li>Напечатать имя текущего каталога</li> <li>Сменить текущий каталог</li> <li>Выполнить введенную команду</li> <li>Создать каталог</li> <li>Удалить каталог вместе с его содержимым</li> <li>Выйти из программы</li> </ul> <p><b>Вариант 2.</b></p> <ul style="list-style-type: none"> <li>Напечатать имя текущего каталога</li> <li>Сменить текущий каталог</li> <li>Создать файл</li> <li>Предоставить всем право на запись в файл</li> <li>Удалить файл</li> <li>Выйти из программы</li> </ul> <p><b>Вариант 3.</b></p> <ul style="list-style-type: none"> <li>Напечатать имя текущего каталога</li> <li>Напечатать содержимое текущего каталога</li> <li>Создать каталог</li> </ul>

№ пп	Формы текущего контроля	Примеры типовых заданий
		<p>Предоставить всем право на запись в каталог            Убрать всем право на запись в каталог            Выйти из программы</p> <p><b>Вариант 4.</b>            Напечатать имя текущего каталога            Сменить текущий каталог            Вывести текущую дату и время в формате Wed Feb 2 01:01 MSK 2020            Вывести содержимое файла на экран            Скопировать файл            Выйти из программы</p> <p><b>Вариант 5.</b>            Напечатать имя текущего каталога            Сменить текущий каталог            Напечатать содержимое текущего каталога            Создать косвенную (символьную) ссылку на файл</p>
9.	Лабораторная работа № 3.1 Команды ОС UNIX/Linux для управления процессами.	<ol style="list-style-type: none"> <li>1. Освойте работу с командой ps. Попробуйте запускать ее с различными аргументами. Если вывод команды не помещается на экране, используйте команду less.</li> <li>2. Выведите в файл отчета1 следующую информацию о запущенных Вами процессах: pid, ppid, tty, uid, command. Вывод должен быть отсортирован по номеру процесса.</li> <li>3. Выведите в файл отчета информацию о процессах, запущенных пользователем root. Вывод должен быть отсортирован по номеру процесса.</li> <li>4. Ключи -c, -v, -j, -i изменяют формат вывода команды. Попробуйте выполнить команду ps с каждым из этих ключей. Результат сохраните в файле отчета.</li> <li>5. Создайте в домашнем каталоге именованный канал fifo. Выполните команду cat /home/labs/dao.txt &gt;fifo Теперь с другого терминала выполните команду cat fifo</li> <li>6. При помощи команд tty, w, uname, uptime выведите в файл отчета имя текуще-го терминала, информацию о пользователях, работающих в системе, название и версию операционной системы, время работы системы.</li> <li>7. Установите обработчик сигнала SIGINT. Для этого выполните команду: trap "echo и получен сигнал SIGINT " INT Пошлите несколько раз оболочке сигнал SIGINT. Для этого следует нажать клавиши CTRL-C.</li> <li>8. Добавьте в начало файла отчета строку с номером лабораторной работы и Вашим именем.</li> </ol>
	Лабораторная работа № 3.2 Механизмы межпроцессного	Запустить несколько заданий (например, команд просмотра файлов less), возвращаясь в командную строку комбинацией клавиш Ctrl-Z и изучить действие команд ps, jobs, fg, bg, kill, killall.

№ пп	Формы текущего контроля	Примеры типовых заданий
	взаимодействия (сигналы, программные (pipes) и именованные (FIFO) каналы)	<p>Обеспечить синхронизацию процессов и передачу данных между ними на примере двух приложений «клиент» и «сервер», создав два процесса (два исполняемых файла) – процесс «клиент» (первый исполняемый файл) и процесс «сервер» (второй исполняемый файл). С помощью механизмов межпроцессного взаимодействия обеспечить передачу информации от «клиента» к «серверу» и наоборот. В качестве типа передаваемой информации можно использовать: данные, вводимые с клавиатуры; данные, считываемые из файла; данные, генерируемые случайным образом и т. п.</p> <p>4. Обмен данными между процессами «клиент»-«сервер» осуществить с использованием программных каналов (именованных либо неименованных, по указанию преподавателя).</p>
	Лабораторная работа № 3.3 Управление потоками. Синхронизация потоков на основе семафоров.	<p>Разработать многопоточную программу с использованием минимум двух потоков и различных средств синхронизации. Например: два потока записывают и читают информацию из одного файла; два потока увеличивают значение общей переменной; два потока с различной частотой считывают и записывают данные в общий буфер памяти. Необходимо обеспечить синхронизированную работу потоков в критической секции с использованием семафоров. Убедиться в результативности применения средств синхронизации потоков, сравнив результаты работы программ с использованием и без использования средств синхронизации.</p>
	Лабораторная работа № 4.1 «Однострочники» на Perl как средство решения задач системного администрирования.	<p>Проработать представленные ниже однострочные выражения на Perl. Результаты выполнения "однострочников" оформить в виде отчёта, файл с которым прикрепить к данному заданию.</p> <pre>prompt\$ ls -lAF   perl -e 'while (&lt;&gt;) { print \$_ }'</pre> <pre>prompt\$ ls -lAF   perl -e 'while (&lt;&gt;) { next if /^[dt]/; print \$_; }'</pre> <pre>prompt\$ ls -lAF   perl -e 'while (&lt;&gt;) { next if /^[dt]/; print +(split)[4], "\n" } '</pre> <pre>prompt\$ ls -lAF   perl -e 'while (&lt;&gt;) { next if /^[dt]/; \$sum += (split)[4] } print "\$sum\n"'</pre> <pre>prompt\$ find .   xargs perl -e '@ARGV = grep( -d \$_ , @ARGV); print "@ARGV"'</pre> <pre>prompt\$ cat afile   perl -MIO::Handle -e 'STDOUT-&gt;autoflush(1); while (&lt;&gt;) { print }'</pre> <pre>prompt\$ cat afile   perl -w -Mstrict -e 'my \$var = 17; print \$var'</pre>
	Лабораторная работа № 4.2 PowerShell как командная оболочка и среда для создания сценариев в ОС Windows.	<ol style="list-style-type: none"> <li>1. Привести пример использования конвейера из двух командлетов «Get-Process   Get-Member» в полном и сокращенном вариантах. Расшифровать столбцы: «MemberType»; «Definition».</li> <li>2. Получить перечень свойств процессов для заданной категории свойств. Категория свойств процессов «-MemberType Property».</li> <li>3. С помощью оболочки «PowerShell» найти несколько конфигурационных файлов,</li> </ol>

№ пп	Формы текущего контроля	Примеры типовых заданий
		<p>заканчивающихся на «*.format.pslxml». Выделить файл «DotNetTypes.format.pslxml»</p> <p>4. Вывести список процессов, упорядоченных по процессорному времени. Вывести список процессов, упорядоченных в порядке убывания процессорного времени.</p> <p>5. Сформировать команду-конвейер для получения пяти процессов, использующих наибольшие объемы памяти.</p> <p>6. Отфильтровать данные по остановленным службам, используя команду-конвейер из двух командлет (свойство «Status» равно «Stopped»).</p> <p>7. Подсчитать объем памяти, занимаемый каталогом (каталог по выбору). Решение задачи – обеспечить последовательностью команд. Исправить некорректную запись в команде-конвейер.</p> <p>8. Подсчитать объем памяти, занимаемый каталогом (каталог по выбору). Решение задачи обеспечить командлетом «MeasureObject» в двух вариантах:          Вариант 1. Расчет памяти, занимаемой каталогом.          Вариант 2. Расчет памяти, занимаемой каталогом с минимумом и максимумом.</p>
10	Контрольная работа по теме «Приёмы работы с текстом средствами системных утилит grep, sed, GNU awk (gawk).»	<ol style="list-style-type: none"> <li>1. Выведите в файл отчета все строки файла text.txt содержащие слово "Программа". Добавьте к файлу отчета количество найденных строк.</li> <li>2. Добавьте к файлу отчета все строки файла text.txt, которые начинаются с буквы "Б" (возможно после нескольких пробелов).</li> <li>3. Добавьте к файлу отчета все непустые строки файла text.txt, не содержащие русскую букву "а" (в любом регистре).</li> <li>4. Добавьте в файл отчета имена файлов каталога /home/labs, которые содержат слово "include".</li> <li>5. Используя редактор sed, замените в файле text.txt все слова Снарк на СНАРК. Результат сохраните в файле snark.txt.</li> <li>6. При помощи sed выведите из файла /etc/passwd идентификаторы пользователей и их имена, разделенные символом табуляции, в формате:             <pre>0 root 1 daemon 2 operator ...</pre> <p>Вывод должен быть отсортирован по идентификатору пользователя. Результат добавьте к файлу отчета.</p> </li> <li>7. Напишите сценарий для sed, который будет менять местами каждые две строки файла.</li> </ol>

№ пп	Формы текущего контроля	Примеры типовых заданий
		<p>Возможно sed придется запускать с ключом -p.</p> <p>8 .Напишите сценарий для редактора sed, который будет добавлять:</p> <p>(a) В начало файла строку "&lt;html&gt;&lt;body&gt;".</p> <p>(b) В начало всех строк начинающихся словом "Приступ" строку "&lt;H2&gt;", а в конец строку "&lt;/H2&gt;".</p> <p>(c) В конец всех остальных строк "&lt;br&gt;".</p> <p>(d) К концу последней строки "&lt;/body&gt;&lt;/html&gt;"</p> <p>Результат работы сценария сохраните в файле snark.html в каталоге public_html домашнего каталога (каталог потребуется создать). Сценарий сохраните под именем sed1 в домашнем каталоге.</p>
10	Контрольная работа по теме «ОС Windows и ОС UNIX/Linux изнутри»	<p>В текстовом файле содержится информация о файлах каталога. Выбрать файлы по заданным шаблонам: а) *.txt; б) lab*.html; в) filedat*.*;</p> <p>2.В текстовом файле содержится информация о документах, входящих в состав информационного сайта. По заданному имени файла найти все связанные с ним документы.</p> <p>3.По заданному имени раздела вывести все файлы, включая вложенные подкаталоги.</p> <p>4.Определить суммарный объем файлов по заданному имени раздела.</p> <p>5.Содержимое текстового файла записать в обратном порядке следования строк в новый файл.</p>
11	Письменное тестирование	<p>1. Сколько способов доступа реализовано в файловой системе UNIX?</p> <p>1) два: для обычного пользователя и для суперпользователя;</p> <p>2) три: хозяин, группа и остальные;</p> <p><b>3) три: запись, чтение и использование;</b></p> <p>4) четыре: суперпользователь, хозяин, группа и остальные.</p> <p>2. Можно ли в Unix удалить чужой файл из своего каталога?</p> <p>1) удалить можно только ссылку, файл удалить нельзя;</p> <p><b>2) да, удаление - это операция над каталогом, а не над файлом ;</b></p> <p>3) да, если права доступа к файлу это позволяют;</p> <p>4) нет, это разрешено только суперпользователю.</p> <p>3. Какая из нижеуказанных команд установит права -wx--x-w- на файл somefile:</p> <p>1) chmod 414 somefile</p>



№ пп	Формы текущего контроля	Примеры типовых заданий
		2) chmod 412 somefile 3) chmod 314 somefile <b>4) chmod 312 somefile</b> 5) chmod 313 somefile

5.2. Критерии, шкалы оценивания текущего контроля успеваемости:

Наименование оценочного средства (контрольно-оценочного мероприятия)	Критерии оценивания	Шкалы оценивания	
		100-балльная система	Пятибалльная система
Лабораторная работа	Работа выполнена полностью. Обучающийся отладил и оптимизировал программы, написал отчет о выполнении работы с соблюдением всех требований. Предусмотрена обработка критических ситуаций. При защите работы обучающийся показал полный объем знаний, умений в освоении пройденной темы и применение их на практике		5
	Работа выполнена полностью. Обучающийся отладил и оптимизировал программы, написал отчет о выполнении работы с соблюдением всех требований. Допущены одна-две ошибки при защите работы.		4
	Программа не оптимизирована. Допущено более двух ошибок при защите работы.		3
	Работа выполнена не полностью. Не проведена оптимизация или отладка программы.		2
Контрольная работа	Задание контрольной работы выполнено полностью. Дан развернутый ответ. При выполнении задания обучающийся предусмотрел обработку критических ситуаций.		5
	Задание контрольной работы выполнено полностью. Дан развернутый ответ. При выполнении задания обучающийся предусмотрел обработку критических ситуаций. Допущена одна ошибка или два-три недочета.		4
	Задание контрольной работы выполнено полностью. При выполнении задания обучающийся предусмотрел обработку критических ситуаций. Допущены одна-две ошибки.		3
	Задание контрольной работы выполнено не полностью. При выполнении задания		2

Наименование оценочного средства (контрольно-оценочного мероприятия)	Критерии оценивания	Шкалы оценивания	
		100-балльная система	Пятибалльная система
	обучающийся не предусмотрел обработку критических ситуаций. Допущена более двух серьезных ошибок.		
Письменное тестирование	За выполнение каждого тестового задания испытуемому выставляются баллы. Используется номинальная шкала оценивания: за правильный ответ к каждому заданию выставляется 1 балл, за не правильный — 0 баллов.		5 85% - 100%
			4 65% - 84%
			3 41% - 64%
			2 40% и менее 40%

### 5.3. Промежуточная аттестация:

Форма промежуточной аттестации	Типовые контрольные задания и иные материалы для проведения промежуточной аттестации:
Зачет с оценкой: в устной форме	<p>Примеры вопросов и контрольных заданий к зачету</p> <ol style="list-style-type: none"> <li>1. Свопинг и виртуальная память.</li> <li>2. Классификация файловых систем ОС UNIX/Linux и ОС Linux.</li> <li>3. Напишите программу, которая запрашивает имя пользователя и выводит его характеристики (группа, имя личного каталога, его содержимое).</li> <li>4. Синхронизация процессов и потоков. Средства синхронизации.</li> <li>5. Сегментно - страничное распределение памяти.</li> <li>6. Символическая связь. Команда создания символической связи. Может ли символическая связь указывать: <ul style="list-style-type: none"> <li>- на директорий?</li> <li>- на другую символическую связь?</li> <li>- на имя в другой файловой системе?</li> <li>- на несуществующее имя в иерархическом дереве?</li> </ul> </li> <li>7. Понятие ядра ОС на примере UNIX.</li> </ol>

	8. Семафоры и их применение в ОС. 9. Создайте конвейер, который выводил бы на экран упорядоченный по алфавиту список зарегистрированных пользователей, содержащий только имена пользователей, их идентификационные номера и имена домашних каталогов.

#### 5.4. Критерии, шкалы оценивания промежуточной аттестации учебной дисциплины:

Форма промежуточной аттестации	Критерии оценивания	Шкалы оценивания	
		100-балльная система	Пятибалльная система
Зачет с оценкой: в устной форме	<p>Обучающийся:</p> <ul style="list-style-type: none"> <li>– демонстрирует знания отличающиеся глубиной и содержательностью, дает полный исчерпывающий ответ, как на основные вопросы билета, так и на дополнительные;</li> <li>– свободно владеет научными понятиями, ведет диалог и вступает в научную дискуссию;</li> <li>– логично и доказательно раскрывает проблему, предложенную в билете;</li> <li>– свободно выполняет практические задания повышенной сложности, предусмотренные программой, демонстрирует системную работу с основной и дополнительной литературой.</li> </ul> <p>Ответ не содержит фактических ошибок и характеризуется глубиной, полнотой, уверенностью суждений, иллюстрируется примерами, в том числе из собственной практики.</p>		<i>Зачтено (отлично)</i>
	<p>Обучающийся:</p> <ul style="list-style-type: none"> <li>– показывает достаточное знание учебного материала, но допускает несущественные фактические ошибки, которые способен исправить самостоятельно, благодаря наводящему вопросу;</li> <li>– недостаточно раскрыта проблема по одному из вопросов билета;</li> <li>– недостаточно логично построено изложение вопроса;</li> <li>– успешно выполняет предусмотренные в программе практические задания средней сложности, активно работает с основной литературой,</li> <li>– демонстрирует, в целом, системный подход к решению</li> </ul>		<i>Зачтено (хорошо)</i>

Форма промежуточной аттестации	Критерии оценивания	Шкалы оценивания	
Наименование оценочного средства		100-балльная система	Пятибалльная система
	<p>практических задач, к самостоятельному пополнению и обновлению знаний в ходе дальнейшей учебной работы и профессиональной деятельности.</p> <p>В ответе раскрыто, в основном, содержание билета, имеются неточности при ответе на дополнительные вопросы.</p>		
	<p>Обучающийся:</p> <ul style="list-style-type: none"> <li>– показывает знания фрагментарного характера, которые отличаются поверхностностью и малой содержательностью, допускает фактические грубые ошибки;</li> <li>– не может обосновать закономерности и принципы, объяснить факты, нарушена логика изложения, отсутствует осмысленность представляемого материала, представления о межпредметных связях слабые;</li> <li>– справляется с выполнением практических заданий, предусмотренных программой, знаком с основной литературой, рекомендованной программой, допускает погрешности и ошибки при теоретических ответах и в ходе практической работы.</li> </ul> <p>Содержание билета раскрыто слабо, имеются неточности при ответе на основные и дополнительные вопросы билета, ответ носит репродуктивный характер. Неуверенно, с большими затруднениями решает практические задачи или не справляется с ними самостоятельно.</p>		<i>Зачтено (удовлетворительно)</i>
	<p>Обучающийся, обнаруживает существенные пробелы в знаниях основного учебного материала, допускает принципиальные ошибки в выполнении предусмотренных программой практических заданий.</p> <p>На большую часть дополнительных вопросов по содержанию экзамена затрудняется дать ответ или не дает верных ответов.</p>		<i>Не зачтено</i>

### 5.5. Система оценивания результатов текущего контроля и промежуточной аттестации.

Оценка по дисциплине выставляется обучающемуся с учётом результатов текущей и промежуточной аттестации.

Форма контроля	100-балльная система	Пятибалльная система
Текущий контроль:		
- лабораторные работы		2 – 5
- контрольная работа		2 – 5
- письменное тестирование		2 – 5
Промежуточная аттестация (зачет с оценкой)		Зачтено (отлично) Зачтено (хорошо)
<b>Итого за семестр</b> (дисциплину) экзамен		Зачтено (удовлетворительно) Не зачтено (неудовлетворительно)

## 6. ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ

Реализация программы предусматривает использование в процессе обучения следующих образовательных технологий:

- проблемная лекция;
- проектная деятельность;
- групповых дискуссий;
- анализ ситуаций и имитационных моделей;
- поиск и обработка информации с использованием сети Интернет.

## 7. ПРАКТИЧЕСКАЯ ПОДГОТОВКА

Практическая подготовка в рамках учебной дисциплины реализуется при проведении практических занятий, связанных с будущей профессиональной деятельностью.

## 8. ОРГАНИЗАЦИЯ ОБРАЗОВАТЕЛЬНОГО ПРОЦЕССА ДЛЯ ЛИЦ С ОГРАНИЧЕННЫМИ ВОЗМОЖНОСТЯМИ ЗДОРОВЬЯ

При обучении лиц с ограниченными возможностями здоровья и инвалидов используются подходы, способствующие созданию безбарьерной образовательной среды: технологии дифференциации и индивидуального обучения, применение соответствующих методик по работе с инвалидами, использование средств дистанционного общения, проведение дополнительных индивидуальных консультаций по изучаемым теоретическим вопросам и практическим занятиям, оказание помощи при подготовке к промежуточной аттестации.

При необходимости рабочая программа дисциплины может быть адаптирована для обеспечения образовательного процесса лицам с ограниченными возможностями здоровья, в том числе для дистанционного обучения.

Учебные и контрольно-измерительные материалы представляются в формах, доступных для изучения студентами с особыми образовательными потребностями с учетом нозологических групп инвалидов:

Для подготовки к ответу на практическом занятии, студентам с ограниченными возможностями здоровья среднее время увеличивается по сравнению со средним временем подготовки обычного студента.

Для студентов с инвалидностью или с ограниченными возможностями здоровья форма проведения текущей и промежуточной аттестации устанавливается с учетом

индивидуальных психофизических особенностей (устно, письменно на бумаге, письменно на компьютере, в форме тестирования и т.п.).

Промежуточная аттестация по дисциплине может проводиться в несколько этапов в форме рубежного контроля по завершению изучения отдельных тем дисциплины. При необходимости студенту предоставляется дополнительное время для подготовки ответа на зачете или экзамене.

Для осуществления процедур текущего контроля успеваемости и промежуточной аттестации обучающихся создаются, при необходимости, фонды оценочных средств, адаптированные для лиц с ограниченными возможностями здоровья и позволяющие оценить достижение ими запланированных в основной образовательной программе результатов обучения и уровень сформированности всех компетенций, заявленных в образовательной программе.

## 9. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Характеристика материально-технического обеспечения дисциплины составляется в соответствии с требованиями ФГОС ВО.

Материально-техническое обеспечение дисциплины при обучении с использованием традиционных технологий обучения.

Наименование учебных аудиторий, лабораторий, мастерских, библиотек, спортзалов, помещений для хранения и профилактического обслуживания учебного оборудования и т.п.	Оснащенность учебных аудиторий, лабораторий, мастерских, библиотек, спортивных залов, помещений для хранения и профилактического обслуживания учебного оборудования и т.п.
<b>119071, г. Москва, Малый Калужский переулок, дом 1</b>	
аудитории для проведения занятий лекционного типа	комплект учебной мебели, технические средства обучения, служащие для представления учебной информации большой аудитории: – ноутбук; – проектор, – экран
<b>119071, г. Москва, Малый Калужский переулок, дом 1, строение 2</b>	
Аудитории № 1217-1219: компьютерный класс для проведения лабораторных и практических занятий групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, по практической подготовке	Комплект учебной мебели, технические средства обучения, служащие для представления учебной информации: 20 персональных компьютеров с подключением к сети «Интернет» и обеспечением доступа к электронным библиотекам и в электронную информационно-образовательную среду организации.
<b>119071, г. Москва, Малый Калужский переулок, дом 1, строение 3</b>	
Помещения для самостоятельной работы обучающихся	Оснащенность помещений для самостоятельной работы обучающихся
читальный зал библиотеки:	– компьютерная техника; – подключение к сети «Интернет»;

Технологическое обеспечение реализации программы осуществляется с использованием элементов электронной информационно-образовательной среды университета.

## 10. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

№ п/п	Автор(ы)	Наименование издания	Вид издания (учебник, УП, МП и др.)	Издательство	Год издания	Адрес сайта ЭБС или электронного ресурса	Количество экземпляров в библиотеке Университета
10.1 Основная литература, в том числе электронные издания							
1	Вавренюк А.Б., Курышева О.К., Кутепов С.В., Макаров В.В.	Операционные системы. Основы UNIX	Учебное пособие	М.:НИЦ ИНФРА-М	2023	<a href="https://znanium.com/catalog/product/2000878">https://znanium.com/catalog/product/2000878</a>	5
2	Зверева О. М	Операционные системы	Учебное пособие	Екатеринбург : Изд-во Уральского ун-та	2020	<a href="https://znanium.com/catalog/product/1936355">https://znanium.com/catalog/product/1936355</a>	
10.2 Дополнительная литература, в том числе электронные издания							
1	Широков, А. И., Кирдяшов Ф.Г., Мурадханов С.Э.	Операционные системы и среды: основные понятия теории	Учебник	М. : Изд. Дом НИТУ «МИСиС»	2018	<a href="https://znanium.com/catalog/product/1232238">https://znanium.com/catalog/product/1232238</a>	-
2	Симмондс, К	Встраиваемые системы на основе Linux	Учебное пособие	М. : ДМК Пресс	2017	<a href="https://znanium.com/catalog/product/1027865">https://znanium.com/catalog/product/1027865</a>	
3	Нейл, Д.	Практическое использование Vim	Учебное пособие	М. : ДМК Пресс, 2017	2017	<a href="https://znanium.com/catalog/product/1028084">https://znanium.com/catalog/product/1028084</a>	
4	Гунько, А. В.	Системное программирование в среде Linux	Учебное пособие	Новосибирск : Изд-во НГТУ	2020	<a href="https://znanium.com/catalog/product/1870577">https://znanium.com/catalog/product/1870577</a>	
3	Монахов В.И., Кузьмич И.В., Степанова О.П., Стрельников Б.А.	Средства межпроцессного взаимодействия и регулярных выражений в операционной системе FreeBSD	Монография	М.: МГУДТ	2016		1
10.3 Методические материалы (указания, рекомендации по освоению дисциплины авторов РГУ им. А. Н. Косыгина)							
1	Стрельников Б.А., Кузьмич И.В., Степанова О.П.	Системное программное обеспечение. Управление процессами в	Методические указания	М.: МГУДТ	2013		+ 15 (на кафедре)

		вычислительных системах. Методические указания к выполнению лабораторных работ.					
2	Стрельников Б.А., Степанова О.П., Кузьмич И.В.	Методические указания к выполнению лабораторных работ. Работа с файлами в операционной системе Unix	Методические указания.	М.: МГУДТ	2013		+ 15 (на кафедре)



## 11. ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ УЧЕБНОГО ПРОЦЕССА

11.1. Ресурсы электронной библиотеки, информационно-справочные системы и профессиональные базы данных:

№ пп	Электронные учебные издания, электронные образовательные ресурсы
1.	ЭБС «Лань» <a href="http://www.e.lanbook.com/">http://www.e.lanbook.com/</a>
2.	«Znaniium.com» научно-издательского центра «Инфра-М» <a href="http://znaniium.com/">http://znaniium.com/</a>
3.	Электронные издания «РГУ им. А.Н. Косыгина» на платформе ЭБС «Znaniium.com» <a href="http://znaniium.com/">http://znaniium.com/</a>
Профессиональные базы данных, информационные справочные системы	
1.	Web of Science <a href="http://webofknowledge.com/">http://webofknowledge.com/</a> (обширная международная универсальная реферативная база данных)
2.	Scopus <a href="https://www.scopus.com">https://www.scopus.com</a> (международная универсальная реферативная база данных, индексирующая более 21 тыс. наименований научно-технических, гуманитарных и медицинских журналов, материалов конференций примерно 5000 международных издательств)
3.	Научная электронная библиотека eLIBRARY.RU <a href="https://elibrary.ru">https://elibrary.ru</a> (крупнейший российский информационный портал в области науки, технологии, медицины и образования)
4.	ООО «Национальная электронная библиотека» (НЭБ) <a href="http://нэб.рф/">http://нэб.рф/</a> (объединенные фонды публичных библиотек России федерального, регионального, муниципального уровня, библиотек научных и образовательных учреждений)

11.2. Перечень программного обеспечения

№п/п	Программное обеспечение	Реквизиты подтверждающего документа/ Свободно распространяемое
1.	Windows 10 Pro, MS Office 2019	контракт 85-ЭА-44-20 от 28.12.2020
2.	Университетский сервер FreeBSD Host:195.133.232.91	Свободно распространяемое программное обеспечение
3.	Среда виртуализации Oracle VirtualBox	Свободно распространяемое программное обеспечение

**ЛИСТ УЧЕТА ОБНОВЛЕНИЙ РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ**

В рабочую программу учебной дисциплины внесены изменения/обновления и утверждены на заседании кафедры:

<b>№ пп</b>	<b>год обновления РПД</b>	<b>характер изменений/обновлений с указанием раздела</b>	<b>номер протокола и дата заседания кафедры</b>