МИНОБРНАУКИ РОССИИ

Федеральное государственное автономное образовательное

учреждение высшего образования «Южный федеральный университет»

(ЮЖНЫЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ)

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Операционные системы реального времени и современные технологии кроссплатформенного программного обеспечения

Ростов-на-Дону – Таганрог 2020

Содержание

[I. Цели и задачи освоения дисциплины 4](#_Toc48234156)

[II. Место дисциплины в структуре образовательной программы 4](#_Toc48234157)

[III. Требования к результатам освоения дисциплины 6](#_Toc48234158)

[IV. Содержание и структура дисциплины 7](#_Toc48234159)

[4.1. Содержание дисциплины, структурированное по темам 7](#_Toc48234160)

[4.2. План внеаудиторной самостоятельной работы 8](#_Toc48234161)

[4.3. Содержание учебного материала 9](#_Toc48234162)

[V. Образовательные технологии 10](#_Toc48234163)

[VI. Учебно-методическое обеспечение дисциплины 11](#_Toc48234164)

[6.1. Основная литература 11](#_Toc48234165)

[6.2. Дополнительная литература 11](#_Toc48234166)

[6.3. Периодические издания 11](#_Toc48234167)

[6.4. Перечень ресурсов сети Интернет 12](#_Toc48234168)

[VII. Материально-техническое обеспечение дисциплины 12](#_Toc48234169)

[7.1. Учебно-лабораторное оборудование 12](#_Toc48234170)

[7.2. Программные средства 12](#_Toc48234171)

[VIII. Методические указания для обучающихся по освоению дисциплины 13](#_Toc48234172)

[IX. Учебная карта дисциплины 16](#_Toc48234173)

[X. Фонд оценочных средств 17](#_Toc48234174)

[10.1. Паспорт фонда оценочных средств 17](#_Toc48234175)

[10.2. Практические занятия 17](#_Toc48234176)

[10.3. Темы рефератов 19](#_Toc48234177)

[10.4. Лабораторные работы 22](#_Toc48234178)

[10.5. Экзаменационные вопросы и билеты 24](#_Toc48234179)

# Цели и задачи освоения дисциплины

Целью освоения дисциплины «Операционные системы реального времени (ОС РВ) и современные технологии кроссплатформенного программного обеспечения (ПО)» является удовлетворение потребностей магистранта в получения качественного высшего образования в области информатики и вычислительной техники. В частности, обеспечивается получение студентом знаний, умений и навыков, направленных на освоение современных технологий разработки кросс-платформенного ПО, а также формирование у студентов компетенций в данном направлении.

Задачи освоения дисциплины «ОС РВ и современные технологии кроссплатформенного ПО»:

* приобретение магистрантами теоретических знаний и практических навыков, в области современных технических и программных средств различных современных операционных систем, их архитектуре и функциях, а также знаний о различных ядрах и других системных механизмов ОС;
* получение магистрантами навыков в выборе современных операционных систем в соответствии с целью их использования, инсталляции, настройке, поддержке, а также поиску и инсталляции вспомогательного программного обеспечения;
* освоение магистрантами теоретических, практических и познавательных аспектов современных технологий разработки кросс-платформенного программного обеспечения для различных операционных систем;
* приобретение магистрантами знаний и навыков в области применения технологий кросс-платформенного программирования в операционных системах различного типа;
* освоение кросс-платформенных инструментов разработки программного обеспечения, как то сред разработки, языков программирования, компиляторов и средств виртуализации;
* освоение студентами современных методов разработки кросс-платформенного программного обеспечения с применением актуальных инструментальных и языковых средств, шаблонов проектирования и шаблонов программирования;
* получение обучающимися фундаментальных знаний, достаточных для эффективного применения технологий кросс-платформенного программирования, а также знание принципов многозадачности, синхронизации потоков, псевдопараллельности для имитации работы программ в реальном времени, а также методов кросс-компиляции программных кодов и учета особенностей архитектуры ОС.

# Место дисциплины в структуре образовательной программы

Дисциплина относится к модулю обязательных профессиональных дисциплин образовательной программы, блок Б1 Дисциплины (модули).

Данная дисциплина опирается на базовые знания, умения и навыки, формируемые при получении предшествующего уровня образования:

| Наименование  дисциплины (модуля), практики | Требуемые знания, умения, навыки |
| --- | --- |
| Операционные системы | Знания:  о сущности информации, ее понятии измерении, количестве и качестве информации, единицах измерения, информационных процессах в автоматизированных системах, основных видах обработки данных, функциональной и структурной организация компьютера, типах и структуре данных, файлах данных, файловых структурах, носителях информации и технических средствах для хранения данных, а также основ компьютерной коммуникации, распределением и управлением памятью, процессах, нитях, потоках. |
| Умения:   * применять методы разработки и администрирования программно-аппаратных комплексов в области операционных систем |
| Программирование | Навыки:   * подготовки и отладки программ на современных языках программирования. * установки и настройки современных операционных систем |

Знания, умения и навыки, формируемые данной дисциплиной, потребуются при освоении следующих элементов образовательной программы:

* Архитектура, программирование и проектирование систем на кристалле;
* Архитектура, конфигурирование и проектирование компьютерных сетей;
* Исследовательский проект.

также необходимы при выполнении различных видов практики, научно-исследовательской работы и итоговой аттестации и написании выпускной квалификационной работы.

# Требования к результатам освоения дисциплины

Освоение дисциплины направлено на формирование следующих компетенций в соответствии с образовательным стандартом и образовательной программой:

Перечень планируемых результатов обучения по дисциплине, соотнесённых с индикаторами достижения компетенций

| Компетенция | Индикаторы достижения компетенции | Результаты обучения |
| --- | --- | --- |
| ПК-2. Способен разрабатывать и применять аппаратное и программное обеспечение в области высокопроизводительных и распределенных вычислительных систем | ПК-2.1 Применяет программные средства и платформы при решении профессиональных задач в области высокопроизводительных и распределенных вычислительных систем | Знания:  – состав и назначение программно-аппаратных комплексов, операционных систем реального времени и кросс-платформенного программирования для высокопроизводительных и распределенных вычислительных систем;  – виды, назначение и архитектуру аппаратных средств и платформ в технологиях кросс-платформенного программирования,  – методы разработки программного кода с применением современных шаблонов программирования. |
| Умения:   * использовать знания о видах, назначении и архитектуре аппаратных средств и платформ в технологиях кросс-платформенного программирования; * применять методы разработки и администрирования программно-аппаратных комплексов, операционных систем реального времени и технологий кросс-платформенного программирования. |
| Навыки:   * навыками разработки и администрирования программно-аппаратных средств с применением операционных систем реального времени и технологий кросс-платформенного программирования в области высокопроизводительных и распределенных вычислительных систем |

# Содержание и структура дисциплины

Трудоёмкость дисциплины составляет 7 зачётных единиц, 246 часов,

Форма промежуточной аттестации: экзамен

## Содержание дисциплины, структурированное по темам

| № п/п | Темы дисциплины | Семестр | Виды учебной работы и их трудоёмкость, часы  (в том числе с использованием онлайн-курсов) | | | | Наименования оценочных средств |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Контактная работа | | | Самостоя­тельная работа |
| Лекции | Практические занятия | Лабораторные занятия |
| **Модуль 1. Особенности программирования для различных операционных систем** | | | | | | | |
| 1 | ОС общего назначения. Параметры ОС. Классификация ОС. Архитектура ОС Windows. Архитектура ОС Linux. Архитектура MacOS. Архитектура ОС Android.  Кросс-платформенное программирование. Понятие кросс-компиляции ПО. Кросс-компиляторы и нативное исполнение кода. Понятие виртуальной машины.  Платформа Java. Платформа .NET. Кросс-платформенное программирование на С++. Инструментарий Qt. Кросс-платформенные скриптовые языки программирования Python, Ruby, Perl. Виртуализация JavaScript. | 2 | 8 | 16 | 16 | 66 | * отчеты по лабораторным работам. * устный опрос по проблемно-ориентированной тематике. * защита реферата |
| **Модуль 2. Современные технологии кросс-платформенного программирования** | | | | | | | |
| 2 | Шаблоны проектирования. Шаблоны программирования. Принципы портирования программ на различные платформы. Средство для разработки кросс-платформенных программ Lazarus. Библиотеки визуальных компонент VCL, LCL и технология визуального программирования. Сборка приложений под ОС Windows, Linux, Android. Перенос программного кода между операционными системами. Принципы разработки кросс-платформенных приложений с использованием JavaScript. | 2 | 8 | 18 | 18 | 66 | * отчеты по лабораторным работам. * устный опрос по проблемно-ориентированной тематике * защита реферата |
| Промежуточная аттестация | | 2 | – | – | – | 36 | Экзаменационные вопросы и билеты |
| **Итого часов** | | **2** | **16** | **34** | **34** | **132+36** | **–** |

## План внеаудиторной самостоятельной работы

| № п/п | Темы дисциплины | Семестр | Вид самостоятельной работы | Сроки выполнения (нед.) | Затраты времени (часы) | Учебно-методическое обеспечение |
| --- | --- | --- | --- | --- | --- | --- |
| **Модуль 1. Особенности программирования для различных операционных систем** | | | | | | |
| 1 | Особенности программирования для различных операционных систем. | 2 | * проработка лекционного материала; * подготовка к практическим и лабораторным занятиям. | 1–9 | 28 | см. раздел VI РПД |
| 2 | Особенности программирования для различных операционных систем. | 2 | * написание реферата и подготовка к его защите. | 1–9 | 38 | см. раздел VI РПД |
| **Модуль 2. Современные технологии кросс-платформенного программирования** | | | | | | |
| 3 | Современные технологии кросс-платформенного программирования | 2 | * проработка лекционного материала; * подготовка к практическим и лабораторным занятиям. | 10–18 | 28 | см. раздел VI РПД |
| 4 | Современные технологии кросс-платформенного программирования | 2 | * написание реферата и подготовка к его защите. | 10–18 | 38 | см. раздел VI РПД |
| **Подготовка к экзамену** | | | | | **36** | см. раздел VI РПД |
| **Общая трудоёмкость самостоятельной работы по дисциплине** | | | | | **132+36** | **–** |

## Содержание учебного материала

**Модуль 1. Особенности программирования для различных операционных систем.**

Особенности ОС общего назначения. Параметры ОС и требования, предъявляемые к ним. Стандарты POSIX. Классификация ОС. Архитектура ОС Windows. Архитектура ОС Linux. Архитектура MacOS. Архитектура ОС Android. Понятие кросс-платформенного программирования. Методы кросс-платформенного программирования. Понятие кросс-компиляции ПО. Кросс-компиляторы и нативное исполнение кода. Проблема инициализации и освобождения памяти. Концепция «сборщика мусора». Понятие виртуальной машины. Платформа Java. Платформа .NET. Кросс-платформенное программирование на С++. Инструментарий Qt. Кросс-платформенные скриптовые языки программирования Python, Ruby, Perl. Виртуализация JavaScript. Инструментарий ObjectPascal.

**Модуль 2. Современные технологии кросс-платформенного программирования.**

История создания кросс-платформенных систем программирования. Основные принципы и средства разработки кросс-платформенного программного обеспечения. Стандарт POSIX. Кросс-платформенные средства программирования. Особенности применения технологий ООП в кросс-платформенном программировании. Философия объектной модели. Шаблоны проектирования. Шаблоны программирования. Принципы портирования программ на различные платформы. Портируемость как возможность единожды откомпилировать программный код. Портируемость как свойство программного обеспечения. Методика разработки кросс-платформенных программ. Средство для разработки кросс-платформенных программ Lazarus. Сборка Lazarus под конкретную операционную системы. Библиотеки визуальных компонент VCL, LCL и технология визуального программирования. Компиляция и запуск программ. Сборка приложений под ОС Windows, Linux, Android. Перенос программного кода между операционными системами. Принципы разработки кросс-платформенных приложений с использованием JavaScript для платформ ОС Windows, Linux, MacOS, Android, iOS.

Перечень тем практических занятий

| №  п/п | Тема практического занятия | Количество часов |
| --- | --- | --- |
| **Модуль 1. Особенности программирования для различных операционных систем.** | | |
| 1 | Введение в разработку приложения на языке высокого уровня Java. | 8 |
| 2 | Введение в разработку приложений с графическим интерфейсом | 8 |
| **Модуль 2. Современные технологии кросс-платформенного программирования.** | | |
| 3 | Введение в разработку прикладных кросс-платформенных приложений | 8 |
| 4 | Введение в разработку графических приложения на языке высокого уровня Java. | 10 |
| **Всего часов** | | **34** |

Перечень лабораторных работ

| №  п/п | Название лабораторной работы | Количество часов |
| --- | --- | --- |
| **Модуль 1. Особенности программирования для различных операционных систем.** | | |
| 1 | Установка и настройка системы виртуализации на базе Oracle VirtualBox. | 8 |
| 2 | Средства разработки кросс-платформенного программного обеспечения | 8 |
| **Модуль 2. Современные технологии кросс-платформенного программирования.** | | |
| 3 | Разработка кросс-платформенного программного обеспечения | 8 |
| 4 | Перенос программного кода на различные операционные системы | 10 |
| **Всего часов** | | **34** |

# Образовательные технологии

При реализации различных видов учебной работы (лекционный курс, практические и лабораторные занятия) используется принцип сочетания аудиторных и электронных форм преподавания.

В лекционных занятиях применяются интерактивные формы общения со студентами в виде разбора конкретных задач и алгоритмов, применяется современные информационные технологии в виде презентаций с применением интерактивных досок и проекционной аппаратуры.

С целью формирования и развития требуемых компетенций обучающихся предусмотрены:

* практические занятия, в рамках которых решаются задачи и алгоритмы, обсуждаются вопросы лекций;
* лабораторные работы с элементами мастер-класса;
* круглый стол по современным проблемам высокопроизводительных вычислительных систем и квантовой обработки информации;
* самостоятельная работа студентов, включающая усвоение теоретического материала, выполнение и подготовка к защите рефератов; подготовка к текущему контролю знаний и к промежуточной аттестации;
* консультирование студентов по вопросам учебного материала.

**Наименование тем занятий с использованием активных форм обучения**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| № | Тема занятия/раздела | Вид занятия | Форма / Методы интерактивного обучения | Кол-во часов |
| 1 | Особенности программирования для различных операционных систем. | Лекции,  практич. занятия. | Лекции с использованием презентационных материалов. Обсуждение вопросов лекций, практических занятий, проведение собеседований по лекционному материалу. | 12 |
| 2 | Современные технологии кросс-платформенного программирования. | Лекции,  практич. занятия | Лекции с использованием презентационных материалов. Обсуждение вопросов лекций, индивидуального задания, проведение собеседований по лекционному материалу. | 12 |
| Итого часов | | | | 24 |

В результате выполнения заданий в приведенной серии лабораторных занятий студенты будут обладать знаниями и умениями по реализации квантовых логических схем для различных алгоритмов, а также получат навыки в моделировании узлов вычислительных систем для реализации специализированных алгоритмов цифровой обработки сигналов.

Прикладные задачи выбираются студентами или выдаются преподавателем. Поощряется решение творческих, нетривиальных задач. Предлагается создание групп по 2-3 человека. Каждая группа в конце курса представит проект, реализующий решение прикладной задачи.

Наряду с традиционными образовательными технологиями, для реализации дисциплины могут использоваться технологии электронного обучения и дистанционные образовательные технологий в электронной информационно-образовательной среде Южного федерального университета. Лекционные занятия и другие формы контактной работы обучающихся с преподавателем могут проводиться с использованием платформ Microsoft Teams, Cisco, Moodle (BigBlueButton) и др., что позволяет обеспечить онлайн и офлайн взаимодействие преподавателя с обучающимися в рамках дисциплины.

Основными методами текущего контроля являются электронный учёт и контроль учебных достижений студентов (использование средств сервиса балльно-рейтинговой системы; ведение электронного журнала успеваемости, проведение электронного тестирования и применение других средств контроля с использованием системы электронного обучения

# Учебно-методическое обеспечение дисциплины

## Основная литература

1. Долженко А. И. Технологии командной разработки программного обеспечения информационных систем / А.И. Долженко - 2-е изд., исправ. - Москва: Национальный Открытый Университет «ИНТУИТ», 2016. – 301 с. – Режим доступа: . <http://biblioclub.ru/index.php?page=book&id=428801>
2. Крахоткина Е. В. Технологии разработки Internet-приложений: учебное пособие / Е.В. Крахоткина – Ставрополь: СКФУ, 2016. – 124 с. – Режим доступа: <http://biblioclub.ru/index.php?page=book&id=459070>
3. Технологии разработки Internet-приложений: лабораторный практикум. – Ставрополь: СКФУ, 2016. – 102 с. – Режим доступа:

<http://biblioclub.ru/index.php?page=book&id=459285>

1. Баженова И. Ю. Язык программирования Java / И.Ю. Баженова. –Моск ва: Диалог-МИФИ, 2008. – 254 с. URL: <http://biblioclub.ru/index.php?page=book&id=54745>
2. Создание кросс-платформенных мультимедийных приложений с использованием библиотеки JUCE на простых примерах. – Режим доступа:

[https://docviewer.yandex.ru/view/0/?\*=G%2BC8VrUTcUA1sPr3oKVM99MrHA17InVybCI6Imh0dHA6Ly9zYW1wbGVzLmxlYW5wdWIuY29tL2p1Y2U0eF9ydS1zYW1wbGUucGRmIiwidGl0bGUiOiJqdWNlNHhfcnUtc2FtcGxlLnBkZiIsInVpZCI6IjAiLCJ5](https://docviewer.yandex.ru/view/0/?*=G%2BC8VrUTcUA1sPr3oKVM99MrHA17InVybCI6Imh0dHA6Ly9zYW1wbGVzLmxlYW5wdWIuY29tL2p1Y2U0eF9ydS1zYW1wbGUucGRmIiwidGl0bGUiOiJqdWNlNHhfcnUtc2FtcGxlLnBkZiIsInVpZCI6IjAiLCJ5)

## Дополнительная литература

1. Современные информационные технологии / В.И. Лебедев. – Ставрополь: СКФУ, 2014. – 225 с. – Режим доступа: <http://biblioclub.ru/index.php?page=book&id=457747>
2. Елманова Н. З. Borland С++Builder 3.0. Архитектура "клиент/сервер", многозвенные системы и Internet-приложения / Н.З. Елманова - Москва: Диалог-МИ ФИ, 1998. – 202 с. – Режим доступа: <http://biblioclub.ru/index.php?page=book&id=89074>
3. Свиридов Г. И. Прикладные сервисы в сети Internet / Г.И. Свиридов – Москва: Ла боратория книги, 2012. – 148 с. – Режим доступа:

<http://biblioclub.ru/index.php?page=book&id=141254>

1. Диков А. В. Веб-технологии HTML и CSS / А.В. Диков. – 2-е изд. – Москва: Директ-Медиа, 2012. – 78 с. URL: <http://biblioclub.ru/index.php?page=book&id=96968>

## Периодические издания

1. Журнал «Вестник компьютерных и информационных технологий», <http://www.vkit.ru/>
2. Журнал «Информационные системы и технологии», <http://oreluniver.ru/science/journal/isit>
3. Журнал «Информационные технологии», <http://novtex.ru/IT/>
4. Журнал «Известия ЮФУ. Технические науки», <http://izv-tn.tti.sfedu.ru/>
5. Журнал «Информатизация и связь», <http://www.infsv.ru/>

## Перечень ресурсов сети Интернет

1. Научная электронная библиотека. <http://elibrary.ru/>
2. Центр аналитической информации. <http://citforum.ru/>
3. Сайт кафедры Вычислительной техники <http://dce.ictis.sfedu.ru/>
4. Материалы по кросс-платформенному программированию

<http://www.crossplatform.ru/node?page=1>.

1. Учебные материалы Microsoft. <https://www.microsoft.com/ru-ru/learning/training.aspx>.

# Материально-техническое обеспечение дисциплины

При реализации дисциплины используются следующие помещения, оборудование и программное обеспечение:

|  |  |  |  |
| --- | --- | --- | --- |
| №№ п/п | Модуль дисциплины | Адрес (местоположение) учебных кабинетов, наименование оборудованных учебных кабинетов, объектов для проведения практических занятий с перечнем основного оборудования | Оснащенность учебного кабинета (технические средства, наборы демонстрационного оборудования, лабораторное оборудование и т.п.) |
| 1 | **Модуль 1.**  Особенности программирования для различных операционных систем | г. Таганрог, ул. Энгельса 1, Корпус Г  Г-410, Г-412, Г-413, Д-509. Электронная доска и проектор Smart Board 6801. Презентации по курсу. | Полная |
| 2 | **Модуль 2.**  Современные технологии кросс-платформенного программирования | г. Таганрог, ул. Энгельса 1, Корпус Г  Г-410, Г-412, Г-413, Д-509. Электронная доска и проектор Smart Board 6801. Презентации по курсу. | Полная |

## Учебно-лабораторное оборудование

Для проведения лабораторных и практических занятий по дисциплине «Операционные системы реального времени и современные технологии кроссплатформенного программного обеспечения» используются лаборатории Г-412, Г-413, Д-509 кафедры ВТ, рабочие станции, а также лицензионное, свободное и разработанное на кафедре ВТ программное обеспечение.

## Программные средства

1. Операционная система Microsoft Windows 10. Лицензионное программное обеспечение по программе Microsoft DreamSpark Premium (URL: <http://ictis.sfedu.ru/msdn/>) предоставляется профессорско-преподавательскому составу, студентам и аспирантам Института компьютерных технологий и информационной безопасности в целях обучения, для выполнения учебных проектов и некоммерческих исследований.
2. Операционная система Ubuntu Linux.
3. Комплект офисных программ MicrosoftOffice. Лицензионное программное обеспечение по программе Microsoft DreamSpark Premium (URL: <http://ictis.sfedu.ru/msdn/>) предоставляется профессорско-преподавательскому составу, студентам и аспирантам Института компьютерных технологий и информационной безопасности в целях обучения, для выполнения учебных проектов и некоммерческих исследований.
4. Кросс-платформенное ПО для разработки программ на языке С++: QtCreator, QtDesigner Cвободное программное обеспечение. Лицензия GNU GPL (URL: <http://www.gnu.org/licenses/>).
5. Кросс-платформенное ПО для разработки программ на языке ObjectPascal: Lazarus Cвободное программное обеспечение. Лицензия GNU GPL (URL: <http://www.gnu.org/licenses/>).
6. Кросс-платформенное ПО для разработки программ на языке Java: NetBeans. Свободное программное обеспечение. Лицензия GNU Lesser General Public License версии 2.1 (LGPL)/GNU General Public License версии 2 (URL: <https://netbeans.org/about/legal/product-licences_ru.html>).
7. Кросс-платформенное ПО для разработки программ на различных языках программирования Visual Studio Code. Лицензия GNU Lesser General Public License версии 2.1 (LGPL)/GNU General Public License версии 2 (URL: <https://netbeans.org/about/legal/product-licences_ru.html>).
8. Microsoft Office (Microsoft Teams), актуальные версии браузеров Chrome, Firefox, Edge, Safari с поддержкой протокола WebRTC

# Методические указания для обучающихся по освоению дисциплины

Дисциплина «ОС РВ и современные технологии кроссплатформенного ПО» читается во 2-ом семестре.

Учебный процесс обучения по дисциплине включает в себя аудиторные занятия (лекции, практические и лабораторные занятия) и самостоятельную работу. Итоговый контроль по дисциплине – экзамен. Лекторы и преподаватели, ведущие практические занятия, контролируют посещение всех видов аудиторных занятий.

Система университетского обучения основывается на рациональном сочетании нескольких видов аудиторных учебных занятий (лекций, практических и лабораторных занятий) и самостоятельной внеаудиторной и аудиторной работы, каждая из которых обладает определенной спецификой.

Если учебные занятия и/или промежуточная аттестация проводятся с использованием ЭО и ДОТ, то при их организации и проведении необходимо руководствоваться Методическими рекомендациями, утвержденными Приказом ректора ЮФУ №394 от 17 марта 2020 г., и инструкциями в разделе «Электронное обучение и дистанционные образовательные технологии» по адресу <http://ictis.sfedu.ru/dot/>

**Подготовка к лекциям.** Знакомство с дисциплиной происходит уже на первой лекции. От студента требуется не просто внимание, но и самостоятельное оформление конспекта. При работе с конспектом лекций необходимо учитывать тот фактор, что одни лекции дают ответы на конкретные вопросы темы, другие – лишь выявляют взаимосвязи между явлениями, помогая студенту понять глубинные процессы развития изучаемого предмета, как в истории, так и в настоящее время.

Конспектирование лекций – сложный вид вузовской аудиторной работы, предполагающий интенсивную умственную деятельность студента. Конспект является полезным тогда, когда записано самое существенное и сделано это самим обучающимся. Не надо стремиться записать дословно всю лекцию. Такое «конспектирование» приносит больше вреда, чем пользы. Целесообразно вначале понять основную мысль, излагаемую лектором, а затем записать ее. Желательно запись осуществлять на одной странице листа или оставляя поля, на которых позднее, при самостоятельной работе с конспектом, можно сделать дополнительные записи, отметить непонятные места.

Конспект лекции лучше подразделять на пункты, соблюдая красную строку. Этому в большой степени будут способствовать вопросы плана лекции, предложенные преподавателям. Следует обращать внимание на акценты, выводы, которые делает лектор, отмечая наиболее важные моменты в лекционном материале замечаниями «важно», «хорошо запомнить» и т.п. Можно делать это и с помощью разноцветных маркеров или ручек, подчеркивая термины и определения.

Целесообразно разработать собственную систему сокращений, аббревиатур и символов. Однако при дальнейшей работе с конспектом символы лучше заменить обычными словами для быстрого зрительного восприятия текста.

Работая над конспектом лекций, всегда необходимо использовать не только учебник, но и ту литературу, которую дополнительно рекомендовал лектор. Именно такая серьезная, кропотливая работа с лекционным материалом позволит глубоко овладеть теоретическим материалом.

**Подготовка к практическим занятиям.** Основная цель проведения практических занятий – формирование у студентов аналитического, творческого мышления путем приобретения практических навыков. Подготовку к каждому практическому занятию студент должен начать с ознакомления с учебной картой практического занятия, которая отражает содержание предложенной темы. Тщательное продумывание и изучение вопросов учебной карты основывается на проработке текущего материала лекции, а затем изучения обязательной и дополнительной литературы, рекомендованной к данной теме. Все новые понятия по изучаемой теме необходимо выучить наизусть и внести в глоссарий, который целесообразно вести с самого начала изучения курса.

Важнейшей составляющей любой формы практических занятий являются упражнения (задания). Основа в упражнении – пример, который разбирается с позиций теории, развитой в лекции. Как правило, основное внимание уделяется формированию конкретных умений, навыков, что и определяет содержание деятельности студентов – решение задач, расчетные работы, уточнение категорий и понятий науки, являющихся предпосылкой правильного мышления и речи.

Результат такой работы должен проявиться в способности студента свободно ответить на теоретические вопросы практикума, участвовать в коллективном обсуждении вопросов изучаемой темы, правильно выполнить практические задания и примеры, контрольные работы.

В процессе подготовки к практическим занятиям, студентам необходимо обратить особое внимание на самостоятельное изучение рекомендованной литературы. При всей полноте конспектирования лекции в ней невозможно изложить весь материал из-за лимита аудиторных часов. Поэтому самостоятельная работа с учебниками, учебными пособиями, научной, справочной литературой, материалами периодических изданий и Интернета является наиболее эффективным методом получения дополнительных знаний, позволяет значительно активизировать процесс овладения информацией, способствует более глубокому усвоению изучаемого материала, формирует у студентов свое отношение к конкретной проблеме.

**Подготовка к лабораторным занятиям.** Основная цель проведения лабораторных занятий – формирование у студентов практических навыков. Подготовку к каждому лабораторному занятию студент должен начать с ознакомления с методическими рекомендациями по выполнению лабораторной работы. Затем студент должен проанализировать предложенный ему вариант задания лабораторной работы и составить для себя план его выполнения. Непосредственно на занятии самостоятельно или в режиме мастер-класса студент выполняет задание лабораторной работы. По итогам его выполнения студент составляет отчет о выполненной лабораторной работе. Требования по оформлению отчета изложены в методических рекомендациях по выполнению лабораторных работ по дисциплине.

Защита лабораторной работы начинается с предъявления преподавателю результата выполнения работы и отчета, в случае удовлетворительного качества предъявленного материала, завершается собеседованием по теме работы.

**Рекомендации по работе с литературой.** Работу с литературой целесообразно начать с изучения общих работ по теме, а также учебников и учебных пособий. Далее рекомендуется перейти к анализу монографий и статей, рассматривающих отдельные аспекты проблем, изучаемых в рамках курса, а также официальных материалов и неопубликованных документов (научно-исследовательские работы, диссертации), в которых могут содержаться основные вопросы изучаемой проблемы. Работу с источниками надо начинать с ознакомительного чтения, т.е. просмотреть текст, выделяя его структурные единицы. При ознакомительном чтении закладками отмечаются те страницы, которые требуют более внимательного изучения.

В зависимости от результатов ознакомительного чтения выбирается дальнейший способ работы с источником. Если для разрешения поставленной задачи требуется изучение некоторых фрагментов текста, то используется метод выборочного чтения. Избранные фрагменты или весь текст (если он целиком имеет отношение к теме) требуют вдумчивого, неторопливого чтения с «мысленной проработкой» материала. Такое чтение предполагает выделение: 1) главного в тексте; 2) основных аргументов; 3) выводов. Особое внимание следует обратить на то, вытекает тезис из аргументов или нет.

Необходимо также проанализировать, какие из утверждений автора носят проблематичный, гипотетический характер и уловить скрытые вопросы. Понятно, что умение таким образом работать с текстом приходит далеко не сразу. Наилучший способ научиться выделять главное в тексте, улавливать проблематичный характер утверждений, давать оценку авторской позиции – это сравнительное чтение, в ходе которого студент знакомится с различными мнениями по одному и тому же вопросу, сравнивает весомость и доказательность аргументов сторон и делает вывод о наибольшей убедительности той или иной позиции.

Если в литературе встречаются разные точки зрения по тому или иному вопросу, нельзя их отвергать, не разобравшись. При наличии расхождений между авторами необходимо найти рациональное зерно у каждого из них, что позволит глубже усвоить предмет изучения и более критично оценивать изучаемые вопросы.

Следующим этапом работы с литературными источниками является создание конспектов, фиксирующих основные тезисы и аргументы. Можно делать записи на отдельных листах, которые потом легко систематизировать по отдельным темам изучаемого курса. Другой способ – это ведение тематических тетрадей-конспектов по одной какой-либо теме. Большие специальные работы монографического характера целесообразно конспектировать отдельно.

Таким образом, при работе с источниками и литературой важно уметь:

* сопоставлять, сравнивать, классифицировать, группировать, систематизировать информацию в соответствии с определенной учебной задачей;
* обобщать полученную информацию, оценивать прослушанное и прочитанное;
* фиксировать основное содержание сообщений; формулировать, устно и письменно, основную идею сообщения; составлять план, формулировать тезисы;
* работать в разных режимах (индивидуально, в паре, в группе), взаимодействуя друг с другом;
* пользоваться реферативными и справочными материалами;
* контролировать свои действия и действия своих товарищей, объективно оценивать свои действия;
* обращаться за помощью, дополнительными разъяснениями к преподавателю, другим студентам.
* пользоваться лингвистической или контекстуальной догадкой, словарями различного характера, различного рода подсказками, опорами в тексте (ключевые слова, структура текста, предваряющая информация и др.).

**Подготовка к промежуточной аттестации.** Студенты, которые по уважительной причине не смогли набрать необходимое число баллов по текущему контролю модуля, могут по согласованию с преподавателем ликвидировать задолженности до начала промежуточной аттестации (экзамена). Основным ориентиром при подготовке к экзамену служат вопросы для экзамена, приведенные в фонде оценочных средств. Изучая материал, относящийся к конкретному вопросу, следует внимательно прочитать рекомендованную литературу, выделить и рассмотреть различные подходы к его решению, проанализировать их сходство и различие, возможные преимущества и недостатки. При подготовке к экзамену рекомендуется составить план ответа на вопрос и привести примеры использования рассматриваемых теоретических положений на практике.

# Учебная карта дисциплины

**Курс 1 семестр 2, очная форма обучения**

|  |  |  |  |
| --- | --- | --- | --- |
| №  п/п | Виды контрольных мероприятий  (наименования оценочных средств) | Количество баллов | |
| Текущий контроль | Рубежный контроль |
| **Модуль 1. Особенности программирования для различных операционных систем** | | | |
| 1 | Лабораторные работы №№ 1, 2 (выполнение, подготовка отчёта, защита отчётов) | 10 (2 работы × 5 баллов) | – |
| 2 | Работа на практических занятиях | 5 | – |
| 3 | Реферат, собеседование | – | 15 |
| **Модуль 2. Современные технологии кросс-платформенного программирования** | | | |
| 4 | Лабораторные работы №№ 3, 4 (выполнение, подготовка отчёта, защита отчётов) | 10 (2 работы × 5 баллов) | – |
| 5 | Работа на практических занятиях | 5 | – |
| 6 | Реферат, собеседование | – | 15 |
| **Всего** | | **30** | **30** |
| Бонусные баллы | | **до 10 баллов**  (Начисляются за активную работу на практических и лабораторных занятиях) | |
| **Промежуточная аттестация  в форме экзамена** | | **40 баллов**  Экзамен считается сданным при получении не менее 22 баллов, для допуска к экзамену необходимо набрать не менее 38 баллов по сумме текущего и рубежного контроля.  Оценка по дисциплине выставляется по сумме баллов за текущий контроль, рубежный контроль и экзамен:   * 85–100 баллов – оценка «отлично»; * 71–84 балла – оценка «хорошо»; * 60–70 баллов – оценка «удовлетворительно»; * менее 60 баллов – оценка «неудовлетворительно». | |

# Фонд оценочных средств

## Паспорт фонда оценочных средств

| № п/п | Индикатор достижения компетенции | Наименование оценочного средства |
| --- | --- | --- |
| 1 | ПК-2.1 Применяет программные средства и платформы при решении профессиональных задач в области высокопроизводительных и распределенных вычислительных систем | * лабораторные работы №№ 1-4 * реферат; * экзаменационные вопросы и билеты |

## Практические занятия

**Цель проведения:**

Практические занятия по дисциплине «Операционные системы реального времени и современные технологии кроссплатформенного программного обеспечения» необходимы для формирования у обучающихся навыков применения средств и инструментов разработки клиентского программного обеспечения на основе современных методов и средств кросс-платформенного программирования.

В результате выполнения студентами практических заданий и изучения лекционного материала на практике, обучающиеся должны обладать знаниями и умениями быстрого создания простых кросс-платформенных приложений и эффективного прототипирования сложных программных комплексов, которые могут запускаться и работать под управлением современных операционных систем различных типов (например, ОС Windows, ОС Linux и Mac OS)

Для проведения практических занятий в основном используется открытое бесплатное программное обеспечения типа Eclipse, NetBeans, Lazarus IDE.

Так же студентам предоставляется доступ к ресурсам сети Интернет, а именно к форумам, справочной информации и электронным учебникам.

Поощряется творческое выполнение практических заданий, решение нетривиальных задач, использование современных технологий сопровождения проектов типа досок заданий (например Trello, Asana) или репозиториев для хранения программного кода (типа GitHub).

**Требования к содержанию и оформлению отчетов:**

Отчет по каждому практическому заданию должен содержать титульный лист с указанием названия задания, фамилий выполнивших студентов, номера группы (подгруппы), фамилию преподавателя; цель практического задания; ход выполнения задания с представлением копий изображений экранов монитора, демонстрирующих основные этапы проведенной работы. Оценивание каждого практического задания проводится во время демонстрации и пояснения полученных результатов.

**Практические задания к модулю 1.**

**Часть 1.**

Тема работы: Введение в разработку приложения на языке высокого уровня Java.

Задание к работе:

- Выполнить обзор среды разработки приложения NetBeans.

- Установка среды разработки и первичная настройка.

- Создание нового проекта.

- Основные типы данных.

- Сложные типы данных.

- Управляющие конструкции.

- Объектная модель.

- Ввод и вывод данных.

**Часть 2.**

Тема работы: Введение в разработку приложений с графическим интерфейсом.

Задание к работе:

- Графический интерфейс пользователя.

- Контейнеры и компоненты Java.

- Библиотека AWT, SWING библиотека.

- Использование визуального редактора интерфейсов.

- Обработка событий.

- Модель слушателя и модель источника события.

- Работа с файлами.

- Чтение и запись файлов.

**Практические задания к модулю 2.**

**Часть 1.**

Тема работы: Введение в разработку прикладных кросс-платформенных приложений.

Задание к работе:

- Разработка простого приложения:

\*\*\* Калькулятор

\*\*\* Учет доходов/расходов.

\*\*\* Кредитный калькулятор.

\*\*\* Список дел.

\*\*\* Список покупок.

- Синтез алгоритма работы приложения.

- Прототипирование графического интерфейса приложения.

- Создание графического интерфейса приложения.

- Разработка объектной модели Калькулятора.

- Программирование объектной модели.

- Связывание объектной модели и графического интерфейса.

- Завершение программного кода.

- Отладка и запуск приложения в ОС Windows.

- Настройка среды программирования в ОС Linux.

- Запуск приложения в ОС Linux.

**Часть 2.**

Тема работы: Введение в разработку графических приложения на языке высокого уровня Java.

Задание к работе:

- Разработка простой игры:

\*\*\* Игра Змейка

\*\*\* Игра Тетрис

\*\*\* Игра 2048

\*\*\* Игра Шарики

\*\*\* Игра крестики/нолики

\*\*\* Игра Башенки

- Синтез алгоритма работы игры.

- Прототипирование графического интерфейса игры.

- Разработка объектной модели игры.

- Проработка игрового цикла.

- Программирование объектной модели.

- Связывание объектной модели и графического интерфейса.

- Завершение программного кода.

- Отладка и запуск Игры в ОС Windows.

- Настройка среды программирования Java в ОС Linux.

- Запуск игры в ОС Linux.

**Критерии оценивания**

5 - 4 балла – обучающийся в течение модуля активно принимает участие в устных опросах, даваемые им ответы верны и позволяют высоко оценить уровень владения материалом, обучающийся решил все поставленные перед ним практические задания творчески и в полном объеме;

3 - 2 балла – обучающийся принимает участие в устных опросах, даваемые им ответы, как правило, верны и позволяют оценить владение материалом на достаточном уровне, практические задания решены в достаточном объеме, но только в рамках предъявленных требований и в них присутствуют ошибки;

1 балл – обучающийся не проявляет инициативы для участия в устных опросах, а даваемые им ответы фрагментарны и верны лишь частично, что не позволяет оценить владение материалом на достаточном уровне;

0 баллов – обучающийся не принимает участия в устных опросах, либо даваемые им ответы принципиально неверны, также не решены практические задания.

## Темы рефератов

***Тематика рефератов по модулям и темам***

**Модуль 1. Особенности программирования для различных операционных систем**

1. Операционные системы как ОС реального времени.
2. Современные операционные системы.
3. Операционные системы реального времени. Основные архитектуры. Сравнение и анализ.
4. Основные понятия и программное обеспечение систем реального времени.
5. Ядра операционные систем.
6. Ядра операционные систем реального времени.
7. Основные механизмы управления процессами в операционных системах реального времени.
8. Виды ОС РВ, основные стандарты и требования к ним.
9. Описание минимального ядра операционной системы реального времени.
10. Операционные системы реального времени для встраиваемых систем.
11. История развития операционных систем.
12. История развития операционных систем реального времени.
13. Модели управления памятью в ОС.
14. Модели управления памятью в ОСРВ.
15. Методы управления аппаратными средствами ЭВМ под управлением ОСРВ.
16. Методология разработки прикладного программного обеспечения ОС.
17. Методология разработки программного обеспечения ОС реального времени.
18. Методы управления задачами в ОС.
19. Методы управления задачами в ОСРВ.
20. Методы управления аппаратными средствами ЭВМ под управлением ОС.
21. Операционная система QNX.
22. Операционная система LinuxRT.
23. Мобильные операционные системы

**Модуль 2. Современные технологии кросс-платформенного программирования**

1. Основные принципы кросс-платформенного программирования.
2. Основные технологии, используемые при кросс-платформенном программировании.
3. Основные инструменты кросс-платформенного программирования.
4. История развития кросс-платформенного программирования.
5. Технология кросс-платформенного программирования на языке Java.
6. Технология кросс-платформенного программирования на языке С++.
7. Технология кросс-платформенного программирования на языке ObjectPascal.
8. Технология кросс-платформенного программирования на скриптовых языках.
9. Современные гибридные технологии JavaScript.
10. Виртуализация программного обеспечения.
11. Особенности реализации POSIX-систем.
12. Виртуальные машины и эмуляция платформ.
13. Основные принципы и технологии портируемости приложений.
14. Базовые методы кросскомпиляции приложений.

**Методические рекомендации по написанию, требования к оформлению**

***Общие положения***

Реферат это одна из форм устной итоговой аттестации. Реферат – это самостоятельная исследовательская работа, в которой автор раскрывает суть исследуемой проблемы, приводит различные точки зрения, а также собственные взгляды не нее. Содержание реферата должно быть логичным, изложение материала носит проблемно-тематический характер.

***Цель***

Реферат как форма промежуточной (итоговой) аттестации стимулирует раскрытие исследовательского потенциала студента, способность к творческому поиску, сотрудничеству, самораскрытию и проявлению возможностей.

***Общие требования к реферату:***

1. Объем – 15-20 страниц.
2. Материалы, которые используются в реферате, не должны быть старше 3 лет.
3. В реферате должно быть содержание.
4. В реферате должен быть список используемых источников.
5. По тексту реферата должны быть ссылки на источники, откуда был получен материал.
6. Должна быть электронная копия реферата и те материалы, которые использовались при подготовке реферата в виде файлов с расширением doc, docx, pdf, html и т. д.
7. Исходные коды программ и другие материалы должны присутствовать в полном объеме в виде Приложения к реферату или на электронном носителе.
8. Допускается индивидуальная тема, согласованная с преподавателем.

***Требования к содержимому реферата***

Автор реферата должен продемонстрировать достижение им уровня профессиональной компетенции, т.е. продемонстрировать знания по теме исследования, существующих междисциплинарных связях, умение проявлять оценочные знания, изучать теоретические работы, использовать различные методы исследования, применять различные приемы творческой деятельности.

1. Необходимо правильно сформулировать тему, отобрать по ней необходимый материал.  
2. Использовать только тот материал, который отражает сущность темы.  
3. Во введении к реферату необходимо обосновать выбор темы.  
4. После цитаты необходимо делать ссылку на автора, например [№ произведения по списку, стр.].  
5. Изложение должно быть последовательным. Недопустимы нечеткие формулировки, речевые и орфографические ошибки.  
6. В подготовке реферата необходимо использовать материалы современных изданий не старше 3 лет.  
7. Оформление реферата (в том числе титульный лист, литература) должно быть грамотным.  
8. Список литературы оформляется с указанием автора, названия источника, места издания, года издания, названия издательства, использованных страниц.

9. Программные коды, разработанные студентами в ходе работы над рефератом, должны быть представлены в полном объеме в виде Приложения к тексту и на электронном носителе.

***Требования к оформлению реферата***

- Изложение текста и оформление реферата выполняют в соответствии с требованиями ГОСТ 7.32 – 2001, ГОСТ 2.105 – 95 и ГОСТ 6.38 – 90. Страницы текстовой части и включенные в нее иллюстрации и таблицы должны соответствовать формату А4 по ГОСТ 9327-60.

- Реферат должен быть выполнен на одной стороне бумаги формата А4 через полтора интервала. Цвет шрифта должен быть черным, высота букв, цифр и других знаков не менее 1.8 (шрифт Times New Roman, 14 пт.).

- Текст следует печатать, соблюдая следующие размеры полей: верхнее и нижнее – 20 мм, левое – 30 мм, правое – 10 мм. Абзацный отступ должен быть одинаковым по всему тексту и составлять 1,25 см.

- Выравнивание текста по ширине.

- Разрешается использовать компьютерные возможности акцентирования внимания на определенных терминах, формулах, применяя выделение жирным шрифтом, курсив, подчеркивание.

- Перенос слов недопустим.

- Точку в конце заголовка не ставят. Если заголовок состоит из двух предложений, их разделяют точкой.

- Подчеркивать заголовки не допускается.

- Расстояние между заголовками раздела, подраздела и последующим текстом так же, как и расстояние между заголовками и предыдущим текстом, должно быть равно 15мм (2 пробела).

- Название каждой главы и параграфа в тексте работы можно писать более крупным шрифтом, жирным шрифтом, чем весь остальной текст. Каждая глава начинается с новой страницы, параграфы (подразделы) располагаются друг за другом.

- В тексте реферат рекомендуется чаще применять красную строку, выделяя законченную мысль в самостоятельный абзац.

- Перечисления, встречающиеся в тексте реферата, должны быть оформлены в виде маркированного или нумерованного списка.

*Пример:*

*Цель работы:  
1). Научиться организовывать свою работу;  
2). Поставить достижимые цели;  
3). Составить реальный план;  
4). Выполнить его и оценить его результаты*

- Все страницы обязательно должны быть пронумерованы. Нумерация листов должна быть сквозной. Номер листа проставляется арабскими цифрами.

- Нумерация листов начинается с третьего листа (после содержания) и заканчивается последним. На третьем листе ставится номер «3».

- Номер страницы на титульном листе не проставляется.

- Номера страниц проставляются в центре нижней части листа без точки. Список использованной литературы и приложения включаются в общую нумерацию листов.

- Рисунки и таблицы, расположенные на отдельных листах, включают в общую нумерацию листов и помещают по возможности следом за листами, на которых приведены ссылки на эти таблицы или иллюстрации. Таблицы и иллюстрации нумеруются последовательно арабскими цифрами сквозной нумерацией. Допускается нумеровать рисунки и таблицы в пределах раздела. В этом случае номер таблицы (рисунка) состоит из номера раздела и порядкового номера таблицы, разделенных точкой.

*Оформление литературы:*  
Каждый источник должен содержать следующие обязательные реквизиты:  
- фамилия и инициалы автора;  
- наименование;  
- издательство;  
- место издания;  
- год издания.  
Все источники, включенные в библиографию, должны быть последовательно пронумерованы и расположены в следующем порядке:  
- законодательные акты;  
- постановления Правительства;  
- нормативные документы;  
- статистические материалы;  
- научные и литературные источники – в алфавитном порядке по первой букве фамилии автора.

В конце работы размещаются приложения. В тексте на все приложения должны быть даны ссылки. Каждое приложение следует начинать с новой страницы с указанием наверху посередине страницы слова «Приложение» и его номера. Приложение должно иметь заголовок, который записывают симметрично относительно текста с прописной буквы отдельной строкой.

**Критерии оценки:**

* оценка (**15 - 13 баллов**) выставляется студенту, если при подготовке темы реферата выполнены все требования, предъявляемые к рефератам, общее количество источников превышает 10, включая источники на иностранном языке, при собеседовании студент отлично ориентируется в теме реферата.
* оценка (**12 - 11 баллов**) – если при подготовке темы реферата требования, предъявляемые к рефератам, выполнены частично: объем менее 15 страниц, или большое количество работ старше 3-х лет, или список использованных источников оформлен не по ГОСТ, или отсутствуют ссылки, при собеседовании студент ориентируется в теме реферата.
* оценка (**10 - 9 баллов**) – если при подготовке темы реферата требования, предъявляемые к рефератам, выполнены с большими отступлениями или полностью нарушены: объем менее 15 страниц, практически все работы старше 3-х лет, список использованных источников оформлен не по ГОСТ, отсутствуют ссылки, при собеседовании студент слабо ориентируется в теме реферата.
* оценка (**менее 9 баллов**) – если при подготовке темы реферата требования, предъявляемые к рефератам, выполнены с большими отступлениями или полностью нарушены: объем менее 15 страниц, приведенная информация не выходит за рамки практических занятий, практически все работы старше 3-х лет, список использованных источников оформлен не по ГОСТ, отсутствуют ссылки, при собеседовании студент не ориентируется в теме реферата, не может ответить ни на один вопрос по теме реферата.

## Лабораторные работы

Целью проведения лабораторных работ по дисциплине «ОС РВ и современные технологии кроссплатформенного ПО» является формирование у обучающихся следующих практических знаний и навыков: установка и настройка различных операционных систем для запуска и тестирования кросс-платформенного ПО, настройка и использование средств виртуализации, установка и настройка средств разработки ПО, разработка структуры программы, синтез программного кода, отладка и запуск, поиск и устранение ошибок программного обеспечения, запуск и поверка кода на различных операционных системах.

*Лабораторная работа 1. Установка и настройка системы виртуализации на базе Oracle VirtualBox.*

Лабораторная работа 1 посвящена изучению методов виртуализации операционных систем различного типа средствами ПЭВМ на примере программного обеспечения Oracle VirtualBox. В процессе работы изучаются и применяются на практике инструменты настройки виртуальных машин, средства поддержки виртуализации, выделения ресурсов, а также процедура создания виртуальных машин, ориентированных на выполнение различных задач. В завершении работы, студент должен свободно устанавливать и настраивать виртуальную машину любой конфигурации согласно задания, а также создавать и изменять параметры виртуальных контейнеров для операционных систем. В лабораторной работе также предусматривается подготовка виртуального окружения для операционных систем Linux и Windows, а также установка необходимого программного обеспечения.

*Лабораторная работа 2. Средства разработки кросс-платформенного программного обеспечения*

Лабораторная работа 2 посвящена изучению инструментов разработки кросс-платформенного программного обеспечения. В процессе работы студенты изучают методы сборки среды разработки Lazarus IDE под различные операционные системы, разрядности и типы процессоров, а также методы разработки, отладки и компиляции кросс-платформенных программ. В завершении работы студент должен обладать достаточными знаниями для разработки кросс-платформенных приложений, проектирования графического интерфейса пользователя, написания программного кода, подключения библиотек и компиляции программ, а также получения исполняемого файла и запуска его на различных операционных системах.

*Лабораторная работа 3. Разработка кросс-платформенного программного обеспечения*

Лабораторная работа 3 посвящена изучению принципов разработки программ для различных операционных систем. В рамках работы студентам предлагается написать программный код, который реализует определенные задачи, выбранные студентами из предложенных вариантов. В завершении работы студент должен иметь четкое представление о методах разработки кросс-платформенных программ согласно методологии проектирования и изученному на лекционных занятиях шаблону программирования.

*Лабораторная работа 4. Перенос программного кода на различные операционные системы*

Лабораторная работа 4 посвящена изучению принципов переноса программного кода в различных операционных системах. В рамках работы студентам предлагается перенести разработанный программный код с одной операционной системы на другую, выполнить его верификацию и тестирование. В завершении работы студент должен иметь четкое представление о методах переноса кросс-платформенных программ на другую платформу.

**Методические рекомендации по выполнению лабораторных работ**

Лабораторные работы проводятся в аудитории, оборудованной в соответствии с требованиями данной программы дисциплины.

Основанием для допуска к лабораторной работе являются:

1. знания теоретического материала и методических указаний, которые должна продемонстрировать бригада студентов в начале занятия.
2. отсутствие задолженностей по предыдущим лабораторным работам, если таковые проводились.

Процесс выполнения лабораторной работы документируется с помощью текстового редактора MS Word, полученные сведения служат основой для формирования отчета о выполнении лабораторной работы.

Лабораторные работы, связанные с виртуализацией, выполняются с использованием инструментов типа Oracle VirtualBox с расширениями.

Лабораторные работы, связанные с программирование, выполняются с применением средств разработки кросс-платформенного программного обеспечения типа IDE Lazarus для языка ObjectPascal или NetBeans для языка Java.

Защита отчета о выполнении лабораторной работы сопровождается демонстрацией полученных результатов, теоретических знаний и ответов на дополнительные вопросы преподавателя по теме занятия.

В процессе подготовки и выполнения лабораторных работ студент руководствуется учебной и методической литературой, указанной в Рабочей программе дисциплины.

**Критерии оценивания**

**5 баллов** выставляется студенту, если он своевременно выполнил все задачи, предусмотренные в лабораторной работе, подготовил отчет в соответствии с требованиями преподавателя и в процессе защиты, продемонстрировал полноту теоретических знаний в объеме содержания учебной дисциплины, относящейся к лабораторной работе. Сумел ответить на дополнительные вопросы, связанные не только с процессом выполнения лабораторной работы, но и с пониманием совершенных действий и решенных задач.

**4 балла** выставляется студенту, если он выполнил все задачи, предусмотренные в лабораторной работе, подготовил отчет в соответствии с требованиями преподавателя и в процессе защиты продемонстрировал наличие достаточных теоретических знаний в объеме содержания учебной дисциплины, относящейся к лабораторной работе. Сумел ответить на вопросы, связанные с процессом выполнения лабораторной работы.

**3 балла** выставляется студенту, если он более чем на половину выполнил поставленные в лабораторной работе задачи, способен ответить на вопросы, касающиеся теоретической составляющей в объеме содержания учебной дисциплины, относящейся к лабораторной работе.

**0 баллов** выставляется студенту, при невыполнении требований, предусмотренных в случае удовлетворительной оценки или, если он не приступил к занятиям в течение семестра.

## Экзаменационные вопросы и билеты

Перечень вопросов к экзамену:

1. Операционные системы. Примеры систем.
2. Операционные системы. Классификация. Принципы работы.
3. Основные характеристики ОС различных типов.
4. Особенности архитектуры операционных систем.
5. Основные особенности монолитной архитектуры ОС.
6. Основные особенности слоеной архитектуры ОС.
7. Основные особенности архитектуры типа клиент-сервер.
8. Функции ядра ОС.
9. Прикладное программное обеспечение для ОС и его разработка.
10. Диспетчирование ОС.
11. Управление памятью и ресурсами ОС.
12. Определение масштаба времени для ОС. Модель временной шкалы.
13. Управление процессами. Организация многозадачности на однозадачной системе. Прерывания.
14. Управление ресурсами, приоритеты нити, разделение ресурсов.
15. Управление процессами. Тупики. Синхронизация.
16. Управление процессорами. Планирование периодических задач.
17. Базовые алгоритмы планирования задач ОС: приоритетные (с переключением) и неприоритетные (без переключения).
18. Алгоритмы диспетчиризации задач в ОС.
19. Способы межпроцессного взаимодействия. Система сообщений. Сигналы.
20. Управление памятью в однозадачных и многозадачных системах.
21. Управление внешней памятью.
22. Обработка запросов внешних устройств в ОС.
23. Взаимодействие с устройствами. Управление внешними устройствами. Правила управления. Драйверы.
24. Особенности ОС UNIX/LINUX.
25. Особенности ОС Windows.
26. Особенности MacOS.
27. Базовые концепции кросс-платформенного программирования.
28. Современные кросс-платформанные библиотеки и фреймворки.
29. Кросскомпиляция.
30. Виртуализация ПО.
31. История создания кросс-платформенных систем программирования.
32. Основные принципы кросс-платформенного программирования.
33. Стандарт POSIX.
34. Использование ООП при кросс-платформенном программировании.
35. Основные инструменты кросс-платформенного программирования.
36. Современные технологии кросс-платформенного программирования.
37. Философия объектой модели.
38. Кросс-платформенность на уровне выполнения.
39. Современные эмуляторы платформ.
40. Принципы кросскомпиляции.
41. Методы портирования приложений на различные платформы.
42. Виртуализация приложений.
43. Средства верификации и тестирования кросс-платформенного ПО.
44. Методы переноса программ.
45. Методы адаптации программ для различных ОС.
46. Шаблоны проектирования кросс-платформенного ПО.
47. Шаблоны программирования.
48. Портируемость как возможность единожды откомпилировать программный код. Портируемость как свойство программного обеспечения.
49. Библиотеки визуальных компонент VCL, LCL и технология визуального программирования.
50. Компиляция и запуск кросс-платформенных программ.
51. Кросс-платформенность языка Java.
52. Виртуальная машина JVM.
53. Принципы JIT-компиляции.
54. Кросс-платформенность языка С/С++.
55. Кросс-платформенность скриптовых языков на примере Python, Ruby, Perl.
56. Кросс-платформенное программирование на JavaScript.
57. Кросс-платформенная экосистема Qt/C++.
58. Кросс-платформенное программирование на языке ObjectPascal.
59. Средство для разработки кросс-платформенных программ Lazarus.
60. Средство для разработки кросс-платформенных программ NetBeans.
61. Средство для разработки кросс-платформенных программ Visual Studio Code.

**Критерии оценивания:**

* **35-40 баллов** – выставляются на экзамене при полном самостоятельном верном ответе на вопросы билета: даны определения, проанализированы различные точки зрения, концептуальные основы рассматриваемой проблемы, приведены примеры. Или если при ответе были допущены неточности, но студент отвечает на любой вопрос, предложенный экзаменатором. Материал излагается логично и последовательно. Студент свободно владеет понятийным аппаратом, умеет его использовать.
* **29-34 баллов** – выставляются на экзамене при достаточно полном ответе на вопросы билета. В ответе проанализированы не все точки зрения на рассматриваемую проблему. Студент владеет понятийным аппаратом, но при его использовании допускает неточности, испытывает небольшие затруднения при обобщении теоретического материала и формулировке выводов. Допускаются некоторые неточности при изложении фактического материала. Во время беседы задавались наводящие вопросы, на которые студент давал незамедлительные ответы.
* **22-28 баллов** – выставляются на экзамене при неполном ответе на вопросы билета. Студент в основном знает содержание понятий, но допускает ошибки в их использовании, испытывает значительные трудности при обобщении теоретического материала и в формулировке выводов, не умеет доказательно обосновать свои суждения. Предпочитает отвечать на вопросы воспроизводящего характера и путается при ответах на видоизмененные вопросы. Материал в значительной степени излагается бессистемно и с нарушением логических связей.
* **менее 22 баллов** ставится если студент имеет разрозненные, бессистемные знания, не умеет выделять главное и второстепенное, допускает ошибки в определении понятий, искажает их смысл, беспорядочно и неуверенно излагает материал, не может применять знания для решения практических задач. Или же демонстрирует полное незнание и непонимание учебного материала или отказывается отвечать.
* **0 баллов** ставится, если студент не явился на экзамен.

**Форма экзаменационного билета**

 МИНОБРНАУКИ РОССИИ

Федеральное государственное автономное образовательное

учреждение высшего образования

«ЮЖНЫЙ ФЕДЕРАЛЬНЫЙ УНИВЕРСИТЕТ»

**ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ №** \_\_\_

По дисциплинеОС РВ и современные технологии кроссплатформенного ПО

Структурное подразделение Институт компьютерных технологий и информационной безопасности

Направление 09.04.01 Информатика и вычислительная техника

Магистерская программа: «Высокопроизводительные вычислительные системы и квантовая обработка информации»

1 Вопрос……………………………………………………………………………

2 Вопрос……………………………………………………………………………