УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«УНИВЕРСИТЕТ УПРАВЛЕНИЯ «ТИСБИ»

Кафедра информационных технологий

Утверждаю
Зав. кафедрой
О.В.Федорова

Протокол заседания
кафедры № 10
от 06.04.2026

Рабочая программа дисциплины

Наименование дисциплины	Основы операционных систем
Направление подготовки	09.03.04 Программная инженерия
Профиль подготовки	Информационные технологии
Год набора	2026

Составители:

Ст.преп., Е.А. Потапова

Казань

Содержание

1. Цели и задачи учебной дисциплины 3

2. Место дисциплины в структуре ОПОП 5

3. Требования к результатам освоения содержания дисциплины 5

4. Структура и содержание дисциплины 7

4.1 Модульно-тематический план и пояснительная записка с указанием этапов формирования компетенций 7

4.2 Содержание дисциплины по темам (разделам) 11

4.3 Планы практических и семинарских занятий 15

4.4 Планы практической подготовки/лабораторных занятий ……..18

5. Учебно-методическое обеспечение самостоятельной работы студентов 17

6. Учебно-методическое и информационное обеспечение дисциплины 17

7. Материально-техническое обеспечение дисциплины 19

8. Оценка компетенций по изучаемой дисциплине 20

Приложение 1. Методические указания для обучающихся по освоению дисциплины

Приложение 2. Фонд оценочных средств для проведения текущей и промежуточной аттестации по дисциплине

1. **Цели и задачи учебной дисциплины**

Дисциплина является одной из основных по направлению подготовки **09.03.04 — Программная инженерия**.

Целью дисциплины является изучение основных принципов функционирования современных универсальных многозадачных операционных систем. Рассматриваются такие важнейшие механизмы как управление процессами и потоками, управление основной памятью, управление внешними устройствами и файлами. Отдельно рассматриваются особенности разработки событийно-управляемых программ в условиях многозадачной работы.

Практическая часть курса состоит из двух частей. Первоначально студенты должны освоить технологию разработки событийно-управляемых программ с использованием одного из современных пакетов разработки оконных приложений (Delphi, Lazarus, Visual Studio, Java NetBeans и др.). Во второй части необходимо реализовать несколько небольших программ, имитирующих работу важнейших подсистем многозадачных ОС.

Задачи:

Изучив курс, студенты должны иметь представление:

- о современных операционных системах,
- особенностях и тенденциях их развития;

Знать:

- операционные системы и оболочки

Уметь:

- моделировать основные функции ОС и анализировать их работу.

Владеть:

- навыками работы с современными операционными системами и средствами разработки программного интерфейса
- основными принципами, связанными с управлением памятью, внешними устройствами и файлами, процессами и потоками

2. Место дисциплины в структуре ОПОП

Данная дисциплина относится к обязательной части Блока 1 учебного плана. До начала изучения дисциплины «Основы операционных систем» у студента должны быть сформированы компоненты компетенций, полученных в результате изучения дисциплин Методы программирования и Архитектура вычислительных систем. Дисциплина находится во взаимосвязи с дисциплинами согласно схеме:

Обеспечивающие учебные дисциплины



Методы программирования

Архитектура вычислительных систем

Основы операционных систем

Обеспечиваемые учебные дисциплины



Объектно-ориентированное программирование

3. Требования к результатам освоения содержания дисциплины

Дисциплина «Основы операционных систем» участвует в формировании следующих компетенций в соответствии с ФГОС ВО по направлению «Программная инженерия»:

ОПК-6 «Способен разрабатывать алгоритмы и программы, пригодные для практического использования, применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов»

ОПК-7 «Способен применять в практической деятельности основные концепции, принципы, теории и факты, связанные с информатикой»

После освоения дисциплины студент должен получить следующие образовательные результаты соотнесённые с индикаторами достижения компетенций.

Декомпозиция компетенций

№ п/п	Название компетенции	Индикатор	Результаты обучения по дисциплине
1	**ОПК-6.** Способен разрабатывать алгоритмы и программы, пригодные для практического использования, применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов	**ОПК-6.1.** Применяет языки программирования и работы с базами данных, современные программные среды разработки информационных систем и технологий для автоматизации бизнес-процессов, решения прикладных задач различных классов, ведения баз данных	**ОПК-6.1. З.2.** Знает операционные системы и оболочки **ОПК-6.1. У.2.** Умеет моделировать основные функции ОС и анализировать их работу
		ОПК-6.2. Программирует, отлаживает и тестирует программные продукты	**ОПК-6.2. В.2.** Владеет навыками работы с современными операционными системами и средствами разработки программного интерфейса
2	**ОПК-7.** Способен применять в практической деятельности основные концепции, принципы, теории и факты, связанные с информатикой	**ОПК-7.2.** Использует в практической деятельности основные концепции, принципы, теории и факты, связанные с информатикой	**ОПК-7.2. В.2.** Владеет основными принципами, связанными с управлением памятью, внешними устройствами и файлами, процессами и потоками

4. Структура и содержание дисциплины

4.1. Модульно-тематический план и пояснительная записка с указанием этапов формирования компетенций

Общая трудоемкость дисциплины составляет 4 зачетных единиц (144 академических часов)

	Модульная разбивка учебной дисциплины
	Направление «Программная инженерия» Дисциплина «Основы операционных систем»

Наименование модулей	Количество ауд. часов			Самостоятельная работа *Очное*	Всего часов	Индикаторы компетенции
	Лекции *Очное*	Практики *Очное*	Практическая подготовка/ Лабораторные занятия *Очное*			
Модуль 1: «Основные понятия»						
Тема 1: Назначение, место и особенности ОС	2		-	4	6	ОПК-6.1
Тема 2: История развития ОС	2		-	4	6	
Тема 3: Структура и основные функции многозадачных ОС	2		-	4	6	
Модуль 2: «Управление процессами и потоками»						
Тема 1. Понятие о процессах и потоках	2		-	4	6	ОПК-6.1 ОПК-7.2
Тема 2. *[1] Состояние потоков, диаграмма переходов, задача планирования порядка выполнения потоков	2*	6	-	4	12	
Тема 3. Взаимодействие и синхронизация потоков	2		-	4	6	
Модуль 3: «Управление основной памятью»						
Тема 1: Основы использования памяти. Адресные пространства	2		-	4	6	ОПК-6.1 ОПК-7.2
Тема 2: * Страничная организация памяти	2*	6	-	4	12	
Тема 3. Сегментная организация памяти	2		-	4	6	

[*] Данная тема изучается с элементами интерактивных методов обучения

Тема 4. Комбинированные способы организации памяти	2		-	4	6	
Модуль 4: «Управление внешними устройствами и файлами»						
Тема 1: Структура подсистемы управления устройствами. Механизм прерываний	2		-	4	6	ОПК-6.1 ОПК-7.2
Тема 2: * Задача управления внешней памятью и способы ее решения	2*	6	-	4	12	
Тема 3: Алгоритмы реализации основных операций с файлами	2		-	6	8	
Модуль 5: Основы разработки событийно-управляемых программ						
Тема 1. Взаимодействие прикладных программ с ядром системы	2		-	6	8	ОПК-6.1 ОПК-6.2
Тема 2. Основы реализации графического интерфейса	2	12	-	6	20	
Подготовка к зачету				18	18	
Всего	**30**	**30**		**84**	**144**	

Пояснительная записка

с этапами формирования компетенций

Данный курс разбит на пять логически завершенных и взаимосвязанных между собой модуля, которые охватывают весь материал дисциплины, обеспечивают приобретение образовательных результатов в соответствии с федеральными государственными образовательными стандартами. Порядок освоения модулей выстраивает траекторию и этапы формирования заявленных компетенций (или их составляющих).

Модуль 1 «Основные понятия» включает в себя три учебные темы.

В результате прохождения первого модуля студент должен:

- знать назначение и особенности операционных систем;
- знать структуру и основные функции многозадачных ОС.

Уровень освоения полученных знаний проверяется тестовым опросом и ответами на теоретические вопросы.

Модуль 2 «Управление процессами и потоками» включает в себя три учебные темы.

В результате прохождения второго модуля студент должен:

- знать такие понятия как процесс и поток, понимать их назначение и отличия, знать порядок создания процессов и потоков и используемые при этом структуры данных, знать алгоритмы работы планировщика потоков;
- уметь разрабатывать простейшие программы, имитирующие работу планировщиков потоков;
- владеть основными принципами управления процессами и потоками;

Уровень освоения полученных знаний и умений проверяется тестовым опросом, написанными программами и ответами на теоретические вопросы.

Тема 2: *Изучение алгоритмов работы планировщика потоков.

Вид проведения занятия: работа с компьютерной тренажерной программой.

Время: на лекции 20 мин., самостоятельно — в зависимости от индивидуальных способностей, на практике — 30 минут.

Сценарий использования тренажерной программы:

После объяснения соответствующего материала студентам на лекции демонстрируется работа тренажерной программы, после чего данная программа предоставляется в распоряжение студентов для изучения во внеучебное время. На практическом занятии студенты под контролем преподавателя работают с тренажером в тестирующем режиме, и по результатам этого тестирования выставляется балльная оценка.

Модуль 3 «Управление основной памятью» включает в себя 4 учебные темы.

В результате прохождения третьего модуля студент должен:

- знать такие понятия как виртуальное и реальное адресное пространство, иметь представление о страничной организации памяти и таблицах страниц, знать алгоритм преобразования виртуальных адресов в физические, иметь представление о сегментной организации памяти и возможных комбинациях страничного и сегментного способов управления памятью;

- уметь разрабатывать простейшие программы, имитирующие работу диспетчера страничной памяти;

- владеть основными принципами управления памятью.

Тема 2: *Страничная организация памяти.

Вид проведения занятия: работа с компьютерной тренажерной программой.

Время: на лекции 20 мин., самостоятельно — в зависимости от индивидуальных способностей, на практике — 30 минут.

Сценарий использования тренажерной программы:

После объяснения соответствующего материала студентам на лекции демонстрируется работа тренажерной программы, после чего данная программа предоставляется в распоряжение студентов для изучения во внеучебное время. На практическом занятии студенты под контролем преподавателя работают с тренажером в тестирующем режиме, и по результатам этого тестирования выставляется балльная оценка.

Модуль 4 «Управление внешними устройствами и файлами» включает в себя 3 учебных темы.

В результате прохождения модуля студент должен:

- знать структуру подсистемы управления устройствами, понимать назначение и реализацию механизма прерываний, знать способы организации внешней памяти и методы распределения этой памяти между файлами, знать алгоритмы реализации основных операций с файлами;

- уметь разрабатывать простые программы, моделирующие работу алгоритмов распределения внешней памяти между файлами;

-владеть основными принципами управления внешними устройствами и файлами;

Тема 2: *Распределение внешней памяти.

Вид проведения занятия: работа с компьютерной тренажерной программой.

Время: на лекции 20 мин., самостоятельно — в зависимости от индивидуальных способностей, на практике — 30 минут.

Сценарий использования тренажерной программы:

После объяснения соответствующего материала студентам на лекции демонстрируется работа тренажерной программы, после чего данная программа предоставляется в распоряжение студентов для изучения во внеучебное время. На практическом занятии студенты под контролем преподавателя работают с тренажером в тестирующем режиме, и по результатам этого тестирования выставляется балльная оценка.

Модуль 5 «Основы разработки событийно-управляемых программ» включает в себя две учебные темы.

В результате прохождения пятого модуля студент должен:

- знать механизм взаимодействия прикладной программы с ядром системы;
- уметь разрабатывать программы с оконным пользовательским интерфейсом на основе современных программных пакетов;
- владеть навыками работы с программами быстрой разработки приложений.

Уровень освоения полученных знаний, умений и навыков проверяется тестовым опросом, выполнением типовых практических заданий и ответами на теоретические и практические вопросы.

Данное деление дисциплины на модули активизирует самостоятельную работу студентов, повышает интенсивность и системность учебной работы, регулирует контроль учебной деятельности студентов в течении семестров, усиливает мотивацию студентов к изучению учебного материала.

4.2. Содержание дисциплины по темам (разделам)

Тема 1. Назначение, место и особенности операционных систем

Понятие об операционных системах и их место среди аппаратно-технических средств. Особенности ОС по сравнению с другими типами программ. Системное и прикладное программное обеспечение.

Тема 2. История развития ОС

Предпосылки создания операционных систем. Семейство ОС от IBM OS/360/370 и реализованные в них механизмы. Семейство ОС Unix. Простейшие системы для персональных компьютеров MS DOS. Развитие семейства ОС Windows. Системы Linux. Операционные системы для мобильных устройств.

Тема 3. Структура и основные функции многозадачных ОС.

Основные функции операционных систем. Физические и логические ресурсы. Ядро и надстройка. Режим ядра и режим пользователя.

Возможные подходы к организации ядра. Многоуровневая организация ядра. Аппаратная поддержка важнейших функций ОС. Аппаратно-зависимый слой ядра. Аппаратно-независимые и управляющие модули. Уровень системных вызовов.

Тема 4. Понятие о процессах и потоках

Понятие о процессах. Внешняя многозадачность. Понятие о потоках. Внутренняя многозадачность. Различия между процессами и потоками. Преимущества и недостатки многопоточных программ. Основные задачи подсистемы управления процессами и потоками.

Основные этапы создания процессов. Дескрипторы процессов. Создание потоков, дескрипторы потоков. Контекст потока. Основные причины завершения потоков.

Тема 5. Состояние потоков, диаграмма переходов, задача планирования порядка выполнения потоков.

Возможные состояния потоков: готовность к выполнению, выполнение, ожидание (блокировка). Причины изменения состояния потоков. Диаграмма переходов между состояниями.

Задача планирования порядка выполнения потоков. Вытесняющая многозадачность. Квантование времени ЦП. Проблема выбора величины кванта. Приоритеты потоков и различные схемы их использования. Статические и динамические приоритеты. Абсолютные приоритеты. Необходимые структуры данных.

Тема 6. Взаимодействие и синхронизация потоков

Обмен данными между потоками. Сигнальный и канальный способы обмена. Общие области памяти. Проблема синхронизации потоков. Задача согласованной обработки разными потоками общей структуры данных. Понятие о семафорах. Двоичные семафоры: мьютексы и критические

секции. Проблема взаимной блокировки потоков и способы ее решения.

Тема 7. Основы использования памяти. Адресные пространства.

Основы использования памяти. Адресация команд и данных. Абсолютные и перемещаемые программы. Виртуальные адреса. Виртуальное и физическое адресное пространство. Задача преобразования виртуальных адресов. Механизм виртуальной памяти. Основные задачи подсистемы управления памятью.

Тема 8. Страничная организация памяти.

Понятие о страницах памяти. Виртуальные и физические страницы. Отображение виртуальных страниц на физические страницы. Таблица страниц процесса. Понятие о дескрипторе страницы. Алгоритм преобразования адресов. Страничное прерывание. Правила освобождения физических страниц. Задача отслеживания свободных и занятых физических страниц. Преимущества и недостатки страничной организации.

Тема 9. Сегментная организация памяти.

Понятие сегмента. Программа как набор виртуальных сегментов. Структура виртуального адреса. Таблица сегментов процесса. Дескриптор сегмента. Алгоритм преобразования адресов. Задача отслеживания свободной и занятой памяти. Фрагментация памяти и способы ее устранения. Преимущества и недостатки сегментной модели.

Тема 10. Комбинированные способы организации памяти.

Сегментно-страничная организация памяти. Виртуальные сегменты. Структура виртуального адреса. Два этапа трансляции адресов. Промежуточный виртуальный адрес и его трансляция в физический адрес. Необходимые таблицы. Плоская модель памяти.

Тема 11. Структура подсистемы управления устройствами. Механизм прерываний.

Общая организация подсистемы ввода/вывода. Аппаратный и программный уровни. Контроллеры и их регистры. Специализация

регистров. Понятие порта ввода/вывода. Низкоуровневые команды ввода/вывода. Драйверы устройств и их взаимодействие с контроллерами. Общая структура драйверов.

Прерывания как способ синхронизации работы внешних устройств. Классификация прерываний. Приоритеты внешних прерываний. Аппаратно-программная обработка прерываний. Контроллер прерываний. Таблица векторов прерываний и обработчики прерываний. Внутренние и программные прерывания.

Тема 12. Задача управления внешней памятью и способы ее решения

Назначение файловой системы. Основные типы файловых систем. Физическая организация хранения информации на жестких дисках. Особенности организации внешней памяти. Виртуальный адрес сектора. Блоки или кластеры, причины их использования. Способы распределения внешней памяти между файлами. Непрерывное распределение и области его применения. Табличное распределение. Индексный способ распределения внешней памяти. Индексный узел и его структура. Использование индексных узлов для хранения больших файлов. Отслеживание свободных и занятых блоков внешней памяти.

Тема 13. Алгоритмы реализации основных операций с файлами

Основные файловые операции: создание/уничтожение, открытие/закрытие, поиск, чтение/запись. Типовая структура каталога. Алгоритм поиска файла по его имени в иерархической древовидной структуре. Алгоритмы создания и уничтожения файлов. Алгоритм открытия файлов и таблица открытых файлов. Алгоритмы чтения и записи файлов.

Тема 14. Взаимодействие прикладных программ с ядром системы

Централизованная обработка событий. Взаимодействие приложения с ядром системы. Сообщения и их основные типы. Программирование с

использованием системных API-вызовов. Базовая структура приложения. Описание оконного класса и создание окон. Правила оформления и структура оконной функции. Дочерние и всплывающие окна, иерархия окон.

Тема 15. Основы реализации графического интерфейса

Основные управляющие элементы, их создание и использование. Взаимодействие с управляющими элементами. Программирование систем меню. Динамическое управление командами меню. Командные сообщения и их обработка в оконной функции. Взаимодействие с клавиатурой. Основные клавиатурные события. Взаимодействие с мышью. Создание и использование таймеров.

4.3. Планы практических и семинарских занятий

Тема 1: Моделирование работы планировщика потоков

Цель: Освоение алгоритмов планирования потоков в многозадачных операционных системах.

Основные этапы выполнения:

1. Практическое изучение работы приоритетного планировщика потоков с помощью тренажерной программы с прохождением контрольного тестирования
2. Разработка программы для моделирования работы простейшего бесприоритетного циклического планировщика потоков
3. Разработка программы для моделирования работы приоритетного планировщика

Тема 2: Моделирование работы диспетчера памяти

Цель: освоение способов организации основной памяти в условиях многозадачности и алгоритмов преобразования адресов

Основные этапы выполнения:

1. Практическое изучение работы страничного механизма управления памятью с помощью тренажерной программы с прохождением

контрольного тестирования

2. Разработка программы для моделирования страничного механизма управления памятью

Тема 3: Моделирование файловых систем

Цель: освоение основных методов управления файлами, алгоритмов распределения дискового пространства и реализации основных файловых операций

Основные этапы выполнения:

1. Практическое изучение работы файловых систем с помощью тренажерной программы с прохождением контрольного тестирования

2. Разработка программы для моделирования файловой системы

Тема 4: Основы разработки событийно-управляемых программ

Цель: Приобретение практических навыков по разработке программ с графическим оконным интерфейсом

Основные этапы выполнения:

1. Изучение одного из пакетов быстрой разработки приложений

2. Написание различных программ в соответствии с заданиями:

- Реализовать простейшую программу, которая создает пустое главное окно со стандартным поведением. Выполнить программу несколько раз для различных типов курсоров, заголовков, фонов заполнения окна.

-Добавить в простейшую программу вывод в главном окне одного дочернего окна, в котором, в свою очередь, выводятся два своих дочерних окна. Исследовать различия в поведении этих окон при перемещении и сворачивании.

- Добавить в простейшую программу следующую возможность: при закрытии главного окна должно выводиться информационное сообщение с подтверждением пользователем факта завершения приложения.

- Программное создание меню.

- Простейший вывод графических примитивов.

4.4 Планы практической подготовки/лабораторных занятий

Не предусмотрено учебным планом

5. Учебно-методическое обеспечение самостоятельной работы студентов

Самостоятельная работа является важным элементом освоения основной программы курса. В соответствии со спецификой предмета, самостоятельная работа включает в себя:

- самостоятельное изучение материала по заданным преподавателем темам;
- подготовку к практическим занятиям по темам;
- работу с компьютерными тренажерными программами
- работу с литературой и учебно-методическими пособиями

Самостоятельная подготовка к практическим занятиям выполняется студентом в свободное от занятий время и включает в себя:

- проработку лекционного материала по указанной теме;
- подготовку алгоритмов решения задач по теме;
- написание программ в соответствии с составленным алгоритмом;
- проверку работоспособности программ, как в домашних условиях, так и в компьютерных классах института;
- предоставление результатов самостоятельной работы преподавателю.

6. Учебно-методическое и информационное обеспечение дисциплины

Основная литература

1. Гостев, И. М. Операционные системы: учебник и практикум для вузов / И. М. Гостев. — 2-е изд., испр. и доп. — Москва: Издательство Юрайт, 2025. — 164 с. — (Высшее образование). — ISBN 978-5-534-04520-

8. — Текст: электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/561557

2. Силаков, Д. В. Операционная система linux. Дистрибьюция программного обеспечения : учебник для вузов / Д. В. Силаков. — Москва : Издательство Юрайт, 2025. — 92 с. — (Высшее образование). — ISBN 978-5-534-21813-8. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/582204

3. Моргунов, А. В. Операционные системы : учебное пособие / А. В. Моргунов. — Новосибирск : Сибирский государственный университет телекоммуникаций и информатики, 2024. — 72 с. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: https://www.iprbookshop.ru/149525.html — Режим доступа: для авторизир. пользователей

Дополнительная литература:

1. Назаров, С. В. Современные операционные системы: учебное пособие / С. В. Назаров, А. И. Широков. — 4-е изд. — Москва: Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2024. — 351 с. — ISBN 978-5-4497-2458-8. — Текст: электронный // Цифровой образовательный ресурс IPR SMART: [сайт]. — URL: https://www.iprbookshop.ru/133980.html

2. Коньков, К. А. Основы операционных систем: учебник / К. А. Коньков, В. Е. Карпов. — 4-е изд. — Москва: Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2025. — 346 с. — ISBN 978-5-4497-0889-2. — Текст: электронный // Цифровой образовательный ресурс IPR SMART: [сайт]. — URL: https://www.iprbookshop.ru/146366.html

3. Операционные системы реального времени и технологии разработки кроссплатформенного программного обеспечения. Ч.4: учебное пособие / Д. А. Беспалов, С. М. Гушанский, Н. М. Коробейникова, В. Е.

Буглов. — Ростов-на-Дону, Таганрог: Издательство Южного федерального университета, 2023. — 115 с. — ISBN 978-5-9275-3366-4, 978-5-9275-4523-0 (ч.4). — Текст: электронный // Цифровой образовательный ресурс IPR SMART: [сайт]. — URL: https://www.iprbookshop.ru/138018.html

Интернет-ресурсы и перечень ежегодно обновляемых современных профессиональных баз данных и информационных справочных систем:

1. «IPRSmart» www.iprbookshop.ru
2. Образовательная платформа «ЮРАЙТ» https://urait.ru
3. http://citforum.ru/
4. https://habr.com/ru/
5. https://stackoverflow.com/
6. https://social.microsoft.com//Forums/ru-RU/home

7. Материально-техническое обеспечение дисциплины

В процессе изучение данной дисциплины используется учебная аудитория, кабинет для самостоятельной работы студентов, читальный зал, видеопроекционное оборудование, компьютер, оснащенный типовым пакетом системного и офисного ПО, в соответствии с Реестром материально-технического обеспечения аудиторного фонда Университета управления «ТИСБИ».

Комплект презентационного оборудования (компьютер с выходом в интернет, проектор, экран).

Персональный компьютер с выходом в интернет в компьютерных классах Университета для каждого студента на практических занятиях.

Пакет лицензионного системного и офисного ПО:

- Microsoft Windows 10.

- Microsoft Office 2013 Professional

Программное обеспечение, входящее в пакет, получает обновление в автоматическом, установленном разработчиком (компанией Microsoft) порядке, посредством сети Интернет.

Visual Studio 2019 (Community) – бесплатная комплексная среда IDE для разработчиков

Lazarus – бесплатная среда по разработке программного обеспечения (https://lazarus-rus.ru/)

Delphi 10.1 Berlin. Среда разработки программного обеспечения. Акт предоставления прав № Tr044490 от 14.09.2016

Собственное программное обеспечение (тренажерные программы)

8. Оценка компетенций по изучаемой дисциплине

Для оценки компетентности рекомендуется использовать рейтинговую оценку знаний, умений и навыков студента по окончанию изучения каждого Модуля в соответствии с Положением о модульно-рейтинговой системе организации образовательного процесс. Итоговая оценка (в баллах) складывается из баллов, набранных по каждому Модулю (семестровая оценка) и баллов, набранных, непосредственно на Зачете (зачете).

Расчет набранных баллов по дисциплине осуществляется в следующей последовательности:

$C = \frac{M_1 + M_2 + ...M_n}{n} \times 0{,}6$, где М – количество баллов по модулю; n – количество модулей

$З = K \times 0{,}4$, где К - количество баллов на Зачете (зачете);

$И = C + З + П$, где П – поощрительные баллы (от 1 до 5).

Уровень сформированности компетенций и их основные признаки оцениваются по следующим таблицам:

Оценка уровня сформированности компетенций ОПК-6

«Способен разрабатывать алгоритмы и программы, пригодные для практического использования, применять основы информатики и программирования к проектированию, конструированию и тестированию программных продуктов»

в части дисциплины «Основы операционных систем»

№ п/п	Уровни сформированности компетенции	Основные признаки уровня	Инструменты оценки сформированности уровня
1	2	3	4
1	**Пороговый уровень (как минимально допустимый)** (обязательный для всех студентов-выпускников вуза по завершении освоения ОПОП ВО) (от 60 до 70 баллов)	- знает назначение, структуру и основные функции ОС; - умеет разрабатывать простые событийно-управляемые программы с использованием одного из пакетов быстрой разработки приложений; - владеет навыками работы в одной из современных операционных систем;	Практические задания Тестирование с использованием тренажерной программы Зачет
2	**Базовый уровень (относительно порогового уровня)** **(**От 71 до 85 баллов**)**	- знает назначение, структуру, основные функции ОС и алгоритмы реализации некоторых из этих функций; - умеет разрабатывать достаточно сложные оконные приложения; - владеет навыками программного моделирования некоторых функций ОС;	Практические задания Тестирование с использованием тренажерной программы Зачет

3	**Повышенный уровень (относительно порогового уровня)** **(**От 86 до 100 баллов**)**	- знает назначение, структуру, основные функции ОС и алгоритмы реализации всех функций; - умеет разрабатывать достаточно сложные оконные приложения; - владеет навыками программного моделирования всех изучаемых в курсе функций ОС;	Практические задания Тестирование с использованием тренажерной программы Зачет

Оценка уровня сформированности компетенции ОПК-7

«Способен применять в практической деятельности основные концепции, принципы, теории и факты, связанные с информатикой»

в части дисциплины «Основы операционных систем»

№ п/п	Уровни сформированности компетенции	Основные признаки уровня	Инструменты оценки сформированности уровня
1	2	3	4
1	**Пороговый уровень (как минимально допустимый)** (обязательный для всех студентов-выпускников вуза по завершении освоения ОПОП ВО) (от 60 до 70 баллов)	- владеет одним из основных принципов, связанными с управлением памятью, или внешними устройствами и файлами, или процессами и потоками;	Практические задания Тестирование с использованием тренажерной программы Зачет

2	**Базовый уровень (относительно порогового уровня)** **(**От 71 до 85 баллов**)**	- владеет некоторыми основными принципами, связанными с управлением памятью, внешними устройствами и файлами, процессами и потоками	Практические задания Тестирование с использованием тренажерной программы Зачет
3	**Повышенный уровень (относительно порогового уровня)** **(От 86 до 100 баллов)**	- владеет всеми основными принципами, связанными с управлением памятью, внешними устройствами и файлами, процессами и потоками	Практические задания Тестирование с использованием тренажерной программы Зачет

Приложение 1

Методические указания для обучающихся по освоению дисциплины

Студентам на первом занятии необходимо ознакомиться с Рабочей программой дисциплины, где прописаны цели, задачи и трудоемкость дисциплины. Перед началом изучения дисциплины необходимо повторить учебный материал обеспечивающих учебных дисциплин предшествующих курсов.

Затем необходимо ознакомиться с порядком изучения дисциплины, т.е. модульно-тематическим планом и пояснительной запиской с указанием этапов формирования заявленных компетенций.

И, наконец, ознакомиться с порядком оценивания результатов обучения, для чего необходимо изучить следующие документы: Положение о модульно-рейтинговой системе оценивания и Принципы оценки уровня знаний, умений и навыков (характеристика ответа).

Студент должен внимательно изучить перечень основной (дополнительной) литературы и взять необходимые учебники в библиотеке.

При изучении данного курса используются интерактивные методы обучения, что помогает эффективнее сформировать заявленные компетенции. Основным средством здесь являются компьютерные тренажерные программы, которые в наглядной форме на конкретных примерах демонстрируют работу изучаемых алгоритмов. Эти программы предоставляются студентам в свободное использование в течение 1-2 недель, после чего на практических занятиях организуется контрольное тестирование. По результатам тестирования преподаватель делает вывод об успешности освоения данного материала. При необходимости контрольное тестирование может быть повторено