

**УТВЕРЖДЕНА**

Решением Ученого совета  
АНО ВО «Центральный университет»  
«07» марта 2024 г.  
Протокол №1

**Рабочая программа дисциплины (модуля)  
«Архитектура операционных систем»**

**Направление подготовки:** 02.03.01 Математика и компьютерные науки

**Направленность (профиль) подготовки:** Искусственный интеллект

**Квалификация (степень) выпускника:** бакалавр

**Форма обучения:** очная

**Срок освоения программы:** 4 года

**Год набора:** 2024

**Москва  
2024**

## Содержание

<b>1. Краткая характеристика дисциплины (модуля)</b> .....	3
<b>2. Перечень планируемых результатов обучения</b> .....	4
<b>3. Тематический план</b> .....	6
<b>4. Содержание дисциплины (модуля)</b> .....	6
<b>5. Учебно-методическое обеспечение</b> .....	8
<b>6. Материально-техническое обеспечение</b> .....	8
<b>7. Методические и оценочные материалы</b> .....	10

## **1. Краткая характеристика дисциплины (модуля)**

Рабочая программа дисциплины (модуля) «Архитектура операционных систем» составлена в соответствии с федеральным государственным образовательным стандартом высшего образования – бакалавриат по специальности 02.03.01 Математика и компьютерные науки, профиль Искусственный интеллект, утвержденный приказом Министерства науки и высшего образования Российской Федерации № 807 от 23.08.2017 года.

Изучение дисциплины (модуля) «Архитектура операционных систем» важно для понимания принципов взаимодействия аппаратного и программного обеспечения, что позволяет эффективно разрабатывать и оптимизировать системы. Это знание способствует созданию надежных, производительных и безопасных вычислительных сред, необходимых в современных технологиях.

### **Место дисциплины (модуля) в структуре образовательной программы**

Настоящая дисциплина (модуль) включена в учебный план по программе подготовки бакалавриата по направлению 02.03.01 Математика и компьютерные науки, профиль Искусственный интеллект и входит в вариативную часть Блока 1, формируемую участниками образовательных отношений.

Дисциплина (модуль) является выборной и доступна для изучения на 3 или 4 курсе в 5, 6, 7, 8 семестрах на выбор.

**Цель изучения дисциплины (модуля):** формирование глубокого понимания внутреннего устройства и принципов работы операционных систем для эффективного проектирования и управления вычислительными ресурсами.

**Задачи изучения дисциплины (модуля)** направлены на формирование у студентов следующий знаний, умений и навыков:

- знание взаимодействия операционной системы с оборудованием;
- знание алгоритмов планирования задач;
- знание принципов устройства файловых систем;
- умение вести разработку на эмуляторе компьютера без ОС;
- навык реализации простой операционной системы с базовым функционалом.

## 2. Перечень планируемых результатов обучения

Компетенции, формируемые в результате освоения дисциплины (модуля) при проведении учебных занятий в форме контактной работы обучающихся с педагогическими работниками Университета и в форме самостоятельной работы обучающихся:

Компетенция	Содержание компетенции	Индикатор компетенции	Перечень планируемых результатов обучения по дисциплине (модулю)
УК-1.	Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач	УК-1.1.	Знает методы поиска и анализа информации в области искусственного интеллекта, основные принципы критической оценки источников информации и их релевантности
		УК-1.2.	Умеет критически оценивать источники информации и синтезировать данные из различных источников для решения задач, применять системный подход к анализу и решению комплексных проблем
		УК-1.3.	Имеет практический опыт работы с современными инструментами и технологиями для обработки информации, формулировании и структурировании задач на основе полученной информации
ОПК-4.	Способен находить, анализировать, реализовывать программно и использовать на практике математические алгоритмы, в том числе с применением современных вычислительных систем	ОПК-4.1.	Знает базовые основы современного математического аппарата, связанного с проектированием, разработкой, реализацией и оценкой качества программных продуктов и программных комплексов в различных областях человеческой деятельности
		ОПК-4.2.	Умеет использовать этот математический аппарат в профессиональной деятельности
		ОПК-4.3.	Имеет практический опыт применения современного математического аппарата, связанного с проектированием, разработкой, реализацией и оценкой качества программных продуктов и программных комплексов в различных областях человеческой деятельности
ОПК-5.	Способен понимать принципы работы современных информационных технологий и использовать их для решения	ОПК-5.1.	Знает технологии, необходимые для прикладного программирования, включая современные функциональные

	задач профессиональной деятельности		языки программирования, а также основные принципы и понятия, применяемыми при использовании компьютерных сетей
		ОПК-5.2.	Умеет пользоваться технологиями прикладного программирования, включая среды высокоуровневого программирования
		ОПК-5.3.	Имеет практический опыт использования технологий прикладного программирования
ПК-3.	Способен применять методы математического и алгоритмического моделирования для решения как теоретических, так и практических задач в рамках профессиональной деятельности	ПК-3.1.	Знает основные методы математического и алгоритмического моделирования, а также их применение для решения теоретических и прикладных задач
		ПК-3.2.	Умеет применять методы математического и алгоритмического моделирования для анализа и решения различных задач в области математики и компьютерных наук
		ПК-3.3.	Имеет опыт использования методов математического и алгоритмического моделирования при решении теоретических и прикладных задач в профессиональной деятельности

### 3. Тематический план

№п/п	Наименование раздела дисциплины (модуля)	Трудоемкость, академические часы				ТКУ (текущий контроль успеваемости)
		<i>Очная форма</i>				
		Контактная работа		Контроль	Самостоятельная работа	
		Лекции	Семинары (практические занятия)			
1	Инструменты для разработки ядра ОС. Процесс загрузки ОС	4	4		18	Домашние задания, Тест
2	Прерывания процессора	4	4		18	Домашние задания, Тест
3	Реализация виртуальной памяти	5	4		20	Домашние задания, Тест
4	Планирование процессов	4	4		18	Домашние задания, Тест
5	Работа с внешними носителями данных	5	4		19	Домашние задания, Тест
6	Базовые принципы организации файловых систем	4	4		19	Домашние задания, Тест
7	Запуск пользовательских процессов	4	4		18	Домашние задания, Тест
	<i>Зачет с оценкой</i>			2		Проект
	<b>Итого:</b>	<b>30</b>	<b>28</b>	<b>2</b>	<b>130</b>	
	<b>Объем дисциплины (модуля) (в ак. ч.)</b>	<b>190</b>				
	<b>Объем дисциплины (модуля) (в зач. ед.)</b>	<b>5</b>				

### 4. Содержание дисциплины (модуля)

№п/п	Наименование раздела дисциплины (модуля)	Содержание дисциплины (модуля) по темам
1	Инструменты для разработки ядра ОС. Процесс загрузки ОС	Использование ассемблера, C и специализированных средств для разработки ядра. Средства отладки и симуляторы для тестирования компонентов ядра. Этапы загрузки: BIOS/UEFI, загрузчик, инициализация ядра. Загрузка и распаковка образа ядра в память. Инициализация аппаратных ресурсов и переход к пользовательскому режиму.
2	Прерывания процессора	Аппаратные и программные прерывания как механизм обработки событий. Вектора прерываний и таблица прерываний (IDT). Обработчики прерываний и их роль в ОС. Маскирование и приоритеты прерываний. Взаимодействие прерываний с планировщиком процессов.
3	Реализация виртуальной памяти	Механизм отображения виртуальных адресов на физические. Использование таблиц страниц и страничной организации памяти. Страничные ошибки и обработка page fault. Свопинг и управление памятью при нехватке ресурсов. Защита памяти и изоляция процессов.

4	Планирование процессов	Механизм отображения виртуальных адресов на физические. Использование таблиц страниц и страничной организации памяти. Страничные ошибки и обработка page fault. Свопинг и управление памятью при нехватке ресурсов. Защита памяти и изоляция процессов.
5	Работа с внешними носителями данных	Интерфейсы и протоколы взаимодействия с устройствами хранения. Буферизация и кэширование данных. Обработка ошибок и восстановление данных. Асинхронный ввод-вывод и прерывания. Управление доступом и синхронизация при работе с носителями.
6	Базовые принципы организации файловых систем	Структура файловой системы: каталоги, файлы, метаданные. Алгоритмы размещения данных: последовательное, индексное, смешанное. Управление свободным пространством (списки, битовые карты). Журналирование и защита целостности данных. Поддержка прав доступа и атрибутов файлов.
7	Запуск пользовательских процессов	Создание процесса: выделение ресурсов и копирование контекста. Загрузка исполняемого файла в память. Инициализация стека и параметров запуска. Передача управления процессу и переход в пользовательский режим. Мониторинг и управление жизненным циклом процесса.

## 5. Учебно-методическое обеспечение

Университет располагает полным набором лицензионного и свободно распространяемого программного обеспечения, включая продукты отечественного производства.

Каждый студент в течение всего периода обучения получает индивидуальный неограниченный доступ к электронно-библиотечной системе и электронной информационно-образовательной среде университета. Эти системы предоставляют возможность доступа к ресурсам из любой точки, где есть подключение к сети Интернет, как на территории университета, так и за его пределами.

Студентам обеспечен удаленный доступ к современным профессиональным базам данных и информационным справочным системам.

### *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>.

### *Дополнительная литература:*

1. Новожилов О. П. Архитектура ЭВМ и вычислительных систем : учебник для вузов / О. П. Новожилов. — 2-е изд., испр. и доп. — Москва : Издательство Юрайт, 2025. — 505 с. — (Высшее образование). — ISBN 978-5-534-20365-3. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/568920>.

2. Толстобров А. П. Архитектура ЭВМ : учебник для вузов / А. П. Толстобров. — 3-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2025. — 162 с. — (Высшее образование). — ISBN 978-5-534-16839-6. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/566711>.

## 6. Материально-техническое обеспечение

Университет располагает материально-технической базой, соответствующей действующим противопожарным правилам и нормам и обеспечивающей проведение всех видов дисциплинарной и междисциплинарной подготовки, практической и научно-исследовательской работ обучающихся, предусмотренных учебным планом.

Помещения, которые представляют собой учебные аудитории для проведения занятий лекционного типа, занятий семинарского (практического) типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, а также помещения для самостоятельной работы и помещения для хранения и профилактического обслуживания учебного оборудования. Помещения укомплектованы специализированной мебелью и техническими средствами обучения, служащими для представления учебной информации большой аудитории.

Изучение дисциплины (модуля) обеспечивается в учебных аудиториях, оснащенных:

- столами и стульями;
- компьютерной техникой;
- специализированным оборудованием, включая демонстрационное оборудование.

Помещения для самостоятельной работы обучающихся, в том числе приспособленные для использования инвалидами и лицами с ограниченными возможностями здоровья, оснащены компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа к в электронную информационно-образовательную среду Университета.

Обучающимся предоставляется доступ (в том числе удаленный) к ресурсам информационно-телекоммуникационной сети «Интернет», электронным ресурсам (в том

числе электронным библиотечным системам, современным профессиональным базам данных и информационным справочным системам):

№	Наименование портала (издания, курса, документа)	Ссылка
1.	Научная электронная библиотека eLibrary.ru библиотека	<a href="https://elibrary.ru/defaultx.asp">https://elibrary.ru/defaultx.asp</a>
2.	База данных для IT-специалистов	<a href="https://habr.com">https://habr.com</a>
3.	База данных ScienceDirect	<a href="https://www.sciencedirect.com">https://www.sciencedirect.com</a>
4.	Официальный сайт Министерства науки и высшего образования Российской Федерации	<a href="https://minobrnauki.gov.ru/">https://minobrnauki.gov.ru/</a>
5.	Федеральный портал «Российское образование»	<a href="https://www.edu.ru/">https://www.edu.ru/</a>
6.	Информационная система "Единое окно доступа к образовательным ресурсам"	<a href="http://window.edu.ru/">http://window.edu.ru/</a>
7.	Единая коллекция цифровых образовательных ресурсов	<a href="http://school-collection.edu.ru/">http://school-collection.edu.ru/</a>
8.	Федеральный центр информационно - образовательных ресурсов	<a href="http://fcior.edu.ru/">http://fcior.edu.ru/</a>

Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине (модулю), в том числе комплект лицензионного программного обеспечения, современные профессиональные базы данных и информационные справочные системы:

Наименование ПО	Производство	Лицензионное / свободно распространяемое
<b>Операционные системы:</b>		
Microsoft Imagine (Windows Client, Server)	зарубежное	лицензионное
<b>Браузеры:</b>		
Яндекс.Браузер	отечественное	свободно распространяемое
Google Chrome	зарубежное	свободно распространяемое
<b>Офисные приложения:</b>		
Microsoft Imagine (Visio, OneNote)	зарубежное	лицензионное
TeXstudio	зарубежное	свободно распространяемое
Adobe Acrobat Reader	зарубежное	свободно распространяемое
<b>Программное обеспечение для планирования и учета времени:</b>		
Toggle app	зарубежное	свободно распространяемое
<b>Системы управления проектами:</b>		
Microsoft Imagine (Project)	зарубежное	лицензионное
<b>Системы управления базами данных:</b>		
Microsoft Imagine (SQL Server)	зарубежное	лицензионное
<b>Системы резервного копирования (backup):</b>		
Acronis Backup Advanced for HyperV	зарубежное	лицензионное
<b>Справочно-правовые системы:</b>		
КонсультантПлюс: справочно-правовая система	отечественное	лицензионное
<b>Средства антивирусной защиты:</b>		
Kaspersky Endpoint Security для бизнеса Стандартный Russian Edition	отечественное	лицензионное
<b>Среды разработки:</b>		
Visual Studio Code	зарубежное	свободно распространяемое
Bash (Unix shell)	зарубежное	свободно распространяемое
Anaconda	зарубежное	свободно распространяемое
Robotic Operating System	зарубежное	свободно распространяемое

CopelliaSim	зарубежное	свободно распространяемое
Google Colaboratory	зарубежное	свободно распространяемое
<b>Пакеты программных средств и библиотек:</b>		
AutoPsy	зарубежное	свободно распространяемое
Interactive Disassembler (IDA)	зарубежное	свободно распространяемое
<b>Системы управления библиографической информацией:</b>		
Zotero	зарубежное	свободно распространяемое
<b>Сервисы и службы:</b>		
Bind	зарубежное	свободно распространяемое
Docker	зарубежное	свободно распространяемое

## 7. Методические и оценочные материалы

### Методические указания для обучающихся по освоению дисциплины (модуля)

В процессе изучения дисциплины (модуля) «Архитектура операционных систем» в рамках текущего контроля успеваемости используются такие виды учебной работы, как лекции, семинары, тесты, домашние задания, проект, а также различные виды самостоятельной работы обучающихся по заданию преподавателя, направленные на развитие навыков профессиональной лексики, закрепление практических профессиональных компетенций, поощрение инициатив.

*Лекция* – систематическое, последовательное, монологическое изложение преподавателем учебного материала, как правило, теоретического характера.

В процессе лекций рекомендуется вести конспект лекций: кратко и схематично фиксировать основные идеи, выводы и обобщения лекции; выделять важные мысли, ключевые слова и термины. Необходимо отметить вопросы или материалы, которые вызывают затруднения, и попытаться найти ответы в рекомендованной литературе. Если разобраться в материале не удастся, следует сформулировать вопрос и задать его преподавателю на консультации или во время семинарского (практического) занятия.

*Участие в семинаре* – активная работа студента на семинаре, его ответы на вопросы преподавателя и участие в дискуссии.

Для успешного участия в семинаре студентам рекомендуется заранее ознакомиться с темой обсуждения, прочитать необходимые материалы и подготовить вопросы. Важно активно слушать и вовлекаться в дискуссию, высказывая свои мнения и аргументируя их. При ответах на вопросы преподавателя стоит быть уверенным, четким и логичным, опираясь на изученный материал. Также полезно поддерживать диалог с однокурсниками, чтобы обогатить обсуждение и расширить свои знания.

*Домашнее задание* – набор задач по темам недели.

При работе над домашними заданиями важно внимательно ознакомиться с требованиями и сроками выполнения. Рекомендуется разбивать задания на этапы, чтобы избежать перегрузки и лучше усвоить материал. Использовать различные источники информации, включая учебники и онлайн-ресурсы, для более глубокого понимания темы.

*Тест* – особая форма проверки знаний. Проводится после освоения одной или нескольких тем и свидетельствует о качестве понимания основных понятий изучаемого материала. Тестовые задания составлены к ключевым понятиям, основным разделам, важным терминологическим категориям изучаемой дисциплины (модуля).

Для подготовки к тесту необходимо знать терминологический аппарат дисциплины (модуля), понимать смысл научных категорий и уметь их использовать в профессиональной лексике. Владение понятийным аппаратом, включённым в тестовые задания, позволяет преподавателю быстро проверить уровень понимания студентами важных

методологических категорий.

*Проект* – исследовательская работа по курсу и презентация результатов.

Для успешной подготовки к проекту: четко определите цели и задачи проекта, распределите роли и обязанности между участниками, а также установите сроки выполнения каждой части работы. Регулярно проводите встречи для обсуждения прогресса и решения возникающих вопросов.

*Самостоятельная работа* – работа студентов, направленная на углубленное изучение отдельных тем и вопросов учебной дисциплины (модуля).

В процессе самостоятельной работы студенты взаимодействуют с рекомендованными материалами при минимальном участии преподавателя. Задачи студента включают работу с конспектами лекций (обработка текста), повторное изучение учебных материалов, планов и тезисов ответов, изучение дополнительных тем, выполнение учебно-исследовательских заданий и другое.

### **Система оценивания результатов обучения по дисциплине (модулю)**

#### **Критерии получения уровня и оценивания сформированности компетенций по дисциплине (модулю) «Архитектура операционных систем»**

Оценивание уровня учебных достижений, обучающихся по дисциплине (модулю), осуществляется в виде текущего контроля успеваемости и промежуточной аттестации.

**Промежуточная аттестация** по дисциплине (модулю) осуществляется в форме *зачета с оценкой*, при этом проводится оценка компетенций, сформированных по дисциплине.

Для оценивания текущего контроля успеваемости и промежуточной аттестации используется десятибалльная шкала оценивания, которая соотносится с традиционной пятибалльной шкалой следующим образом:

<b>Десятибалльная оценка</b>	<b>Пятибалльная оценка</b>	<b>Оценка за зачет</b>	<b>Общая характеристика результата обучения по дисциплине (модулю)</b>
10	Отлично	Зачтено	Студент полностью владеет знаниями, изложенными в рабочей программе, и глубоко осмысляет дисциплину. Он самостоятельно и логически последовательно отвечает на все вопросы, акцентируя внимание на наиболее важном. Умеет анализировать, сравнивать, классифицировать, обобщать, конкретизировать и систематизировать изученный материал, выделяя ключевые моменты и устанавливая причинно-следственные связи. Четко формулирует ответы, уверенно интерпретирует результаты анализов и других исследований, а также решает сложные задачи. Студент хорошо знаком с методами исследования, необходимыми для практической деятельности, и умеет связывать теоретические аспекты
9	Отлично	Зачтено	
8	Отлично	Зачтено	

Десятибалльная оценка	Пятибалльная оценка	Оценка за зачет	Общая характеристика результата обучения по дисциплине (модулю)
			дисциплины (модуля) с практическими задачами.
7	Хорошо	Зачтено	Студент обладает знаниями предмета почти в полном объеме рабочей программы и самостоятельно, логически последовательно и всесторонне отвечает на все вопросы, акцентируя внимание на наиболее значимых моментах. Он умеет анализировать, сравнивать, классифицировать, обобщать, конкретизировать и систематизировать изученный материал, выделяя его ключевые аспекты и устанавливая причинно-следственные связи. Формулирует свои ответы, уверенно интерпретирует результаты анализов и других исследований, а также решает сложные ситуационные задачи. Студент хорошо знаком с методами исследования, необходимыми для практической деятельности, и умеет связывать теоретические аспекты предмета с практическими задачами.
6	Хорошо	Зачтено	
5	Удовлетворительно	Зачтено	Студент обладает базовыми знаниями по дисциплине (модулю), но испытывает трудности при самостоятельных ответах и использует неточные формулировки. В ходе ответов он допускает ошибки, касающиеся сути вопросов. Студент способен решать только самые простые задачи и владеет лишь минимальным набором методов исследования.
4	Удовлетворительно	Зачтено	
3	Не сдан	Не зачтено	Студент не овладел обязательным минимумом знаний по предмету и не может ответить на вопросы, даже если преподаватель задает дополнительные наводящие вопросы.
2	Не сдан	Не зачтено	
1	Не сдан	Не зачтено	

Дисциплина (модуль) «Архитектура операционных систем» оценивается следующим образом:

Активность	Вес	Количество	Описание
Домашние задания	35%	14	Набор задач по темам недели
Тесты	25%	7	Письменная работа с набором задач, которые нужно решить за ограниченное время
Зачет с оценкой	40%	1	Защита итогового проекта

**Формула расчёта итоговой оценки по дисциплине (модулю) «Архитектура операционных систем»:**  $\langle 0,35 \times \text{среднее за домашние задания} + 0,25 \times \text{среднее за тесты} + 0,4 \times \text{зачет с оценкой} \rangle$ .

## **Текущий контроль успеваемости обучающихся по дисциплине (модулю)**

### **Примерные домашние задания**

#### **Домашнее задание по теме «Инструменты для разработки ядра ОС. Процесс загрузки ОС»**

**Задание 1.** Изучите синтаксис базовых команд ассемблера x86 и напишите простую программу, которая выводит сообщение на экран в режиме реального времени (например, с помощью BIOS прерывания).

**Задание 2.** Создайте минимальный загрузчик на ассемблере, который загружает и передает управление простому коду на C, скомпилированному в виде отдельного файла.

**Задание 3.** Настройте и используйте эмулятор QEMU для запуска собственного загрузочного сектора и ядра ОС, отлаживая процесс с помощью GDB.

**Задание 4.** Опишите по шагам процесс загрузки ОС от включения питания до запуска пользовательского режима, выделяя роли BIOS/UEFI, загрузчика и ядра.

**Задание 5.** Исследуйте и напишите краткое описание одного из современных загрузчиков (например, GRUB), объясняя, как он загружает ядро и передает ему управление.

#### **Домашнее задание по теме «Прерывания процессора»**

**Задание 1.** Напишите программу на C с использованием inline-ассемблера, которая вызывает программное прерывание (например, INT 0x80) и обработайте его в простом обработчике.

**Задание 2.** Изучите структуру таблицы прерываний (IDT) и составьте схему, показывающую, как аппаратные прерывания связываются с обработчиками.

**Задание 3.** Реализуйте простой обработчик аппаратного прерывания (например, таймера) в эмуляторе QEMU, который увеличивает счетчик каждую секунду.

**Задание 4.** Изучите механизм маскировки прерываний (например, с помощью PIC или APIC) и опишите, как приоритеты прерываний влияют на порядок их обработки.

**Задание 5.** Опишите взаимодействие прерываний с планировщиком процессов: как прерывания могут инициировать переключение контекста.

#### **Домашнее задание по теме «Реализация виртуальной памяти»**

**Задание 1.** Напишите программу, которая демонстрирует трансляцию виртуального адреса в физический с помощью таблицы страниц (модельно, без реального аппаратного взаимодействия).

**Задание 2.** Изучите структуру таблиц страниц в архитектуре x86-64 и опишите роль каждого уровня (PML4, PDPT, PD, PT).

**Задание 3.** Смоделируйте обработку page fault: опишите, что происходит в ОС при возникновении страничной ошибки.

**Задание 4.** Исследуйте механизм свопинга: напишите краткий отчет о том, как ОС освобождает память при нехватке ресурсов.

**Задание 5.** Изучите методы защиты памяти (например, бит прав доступа) и опишите, как ОС изолирует процессы друг от друга.

#### **Домашнее задание по теме «Планирование процессов»**

**Задание 1.** Реализуйте на псевдокоде алгоритм планирования Round Robin с квантованием времени и переключением контекста.

**Задание 2.** Опишите жизненный цикл процесса и составьте диаграмму состояний (готов, выполняется, ожидание и т.д.).

**Задание 3.** Исследуйте и сравните алгоритмы планирования FIFO, приоритетного и многопоточного, выделив их преимущества и недостатки.

**Задание 4.** Напишите отчет о том, как планировщик учитывает приоритеты и время отклика при выборе следующего процесса для выполнения.

**Задание 5.** Опишите механизм переключения контекста: что сохраняется и восстанавливается, и почему это важно для многозадачности.

### Примерные задания по тестам

#### Тест № 1

1. **Что такое загрузчик (bootloader)?**

- a) Программа, управляющая вводом-выводом
- b) Программа, загружающая ОС в память и передающая ей управление
- c) Модуль ядра, отвечающий за планирование процессов
- d) Программа для отладки кода

**Ответ: b**

2. **В каком режиме работает BIOS при старте компьютера?**

- a) Защищённом режиме
- b) Виртуальном режиме
- c) Пользовательском режиме
- d) Реальном режиме

**Ответ: d**

3. **Что происходит после выполнения BIOS POST (Power-On Self Test)?**

- a) Загружается ядро ОС напрямую
- b) Запускается планировщик процессов
- c) Выполняется загрузчик с загрузочного устройства
- d) Инициализируется виртуальная память

**Ответ: c**

4. **Какой из перечисленных загрузчиков является наиболее популярным для Linux?**

- a) LILO
- b) GRUB
- c) NTLDR
- d) BOOTMGR

**Ответ: b**

5. **Что такое сектор загрузки (boot sector)?**

- a) Первый сектор на диске, содержащий загрузочный код
- b) Область памяти для хранения пользовательских данных
- c) Файл с ядром ОС
- d) Таблица разделов диска

**Ответ: a**

6. **В каком формате обычно записывается код загрузчика?**

- a) Машинный код
- b) Скомпилированный C-код
- c) Ассемблерный код, преобразованный в машинный код
- d) Скриптовый язык

**Ответ: c**

7. **Что такое GDB?**

- a) Эмулятор процессора
- b) Отладчик программного кода
- c) Загрузчик ОС
- d) Планировщик процессов

**Ответ: b**

**8. Что такое UEFI?**

- a) Язык программирования
- b) Тип процессора
- c) Формат файла ядра ОС
- d) Новый стандарт BIOS с расширенными возможностями

**Ответ: d**

**9. Какую роль выполняет загрузчик второго этапа?**

- a) Загружает ядро ОС в память и передаёт управление
- b) Выполняет диагностику оборудования
- c) Инициализирует драйверы устройств
- d) Управляет пользовательским интерфейсом

**Ответ: a**

**10. Что означает термин "передача управления" в процессе загрузки?**

- a) Копирование данных из памяти на диск
- b) Инициализация аппаратных прерываний
- c) Переход исполнения кода от загрузчика к ядру ОС
- d) Настройка сетевого соединения

**Ответ: c**

**Тест № 2**

**1. Что такое прерывание?**

- a) Сбой в работе процессора
- b) Команда для завершения программы
- c) Постоянный цикл выполнения программы
- d) Событие, заставляющее процессор временно прервать текущую работу для обработки другого события

**Ответ: d**

**2. Что такое IDT (Interrupt Descriptor Table)?**

- a) Таблица с настройками памяти
- b) Таблица, содержащая адреса обработчиков прерываний
- c) Список процессов ОС
- d) Таблица файловой системы

**Ответ: b**

**3. Какой тип прерывания инициируется аппаратным устройством?**

- a) Аппаратное прерывание
- b) Программное прерывание
- c) Исключение
- d) Системный вызов

**Ответ: a**

**4. Что такое программное прерывание?**

- a) Прерывание, вызванное аппаратным устройством
- b) Прерывание из-за ошибки памяти
- c) Прерывание, вызванное инструкцией процессора (например, INT)
- d) Прерывание при выключении питания

**Ответ: c**

**5. Что делает обработчик прерывания?**

- a) Отключает процессор
- b) Выполняет специальный код для обработки события, вызвавшего прерывание
- c) Удаляет процесс из памяти
- d) Запускает новый процесс

**Ответ: b**

**6. Что такое маскирование прерываний?**

- a) Процесс временного отключения определённых прерываний

- b) Полное удаление прерываний из системы
- c) Смена адреса обработчика прерывания
- d) Запуск планировщика процессов

**Ответ: a**

**7. Какое устройство обычно отвечает за управление аппаратными прерываниями в старых системах?**

- a) CPU
- b) RAM
- c) PIC (Programmable Interrupt Controller)
- d) SSD

**Ответ: c**

**8. Что происходит при аппаратном прерывании таймера?**

- a) Процессор выключается
- b) Загружается загрузчик ОС
- c) Останавливается выполнение всех программ
- d) Планировщик процессов может инициировать переключение контекста

**Ответ: d**

**9. Какая инструкция вызывает программное прерывание в x86?**

- a) INT
- b) JMP
- c) CALL
- d) RET

**Ответ: a**

**10. Почему прерывания важны для многозадачности?**

- a) Они позволяют процессору выполнять только одну задачу
- b) Они позволяют ОС прерывать текущий процесс и переключаться на другой
- c) Они ускоряют работу процессора
- d) Они уменьшают потребление энергии

**Ответ: b**

### **Тест № 3**

**1. Что такое виртуальная память?**

- a) Физическая память компьютера
- b) Абстракция, предоставляющая процессам собственное адресное пространство
- c) Постоянное хранилище данных
- d) Кэш процессора

**Ответ: b**

**2. Как называется таблица, которая преобразует виртуальные адреса в физические?**

- a) Таблица файлов
- b) Таблица страниц (Page Table)
- c) Таблица процессов
- d) Таблица прерываний

**Ответ: b**

**3. Что такое page fault?**

- a) Ошибка процессора при выполнении инструкции
- b) Исключение, возникающее при обращении к неотображённой странице памяти
- c) Команда для переключения контекста
- d) Прерывание от устройства ввода

**Ответ: b**

**4. Что происходит при возникновении page fault?**

- a) ОС загружает нужную страницу из диска в память
- b) Процессор выключается

- c) Процесс завершается без сохранения данных
- d) Запускается загрузчик ОС

**Ответ: a**

**5. Что такое свопинг (swapping)?**

- a) Переключение между процессами
- b) Перемещение страниц памяти между ОЗУ и диском для освобождения памяти
- c) Ошибка памяти
- d) Планирование задач

**Ответ: b**

**6. Какие уровни таблиц страниц существуют в x86-64 архитектуре?**

- a) Только одна таблица
- b) PML4, PDPT, PD, PT
- c) BIOS, UEFI, Kernel, User
- d) IDT, GDT, LDT

**Ответ: b**

**7. Что обеспечивает защита памяти в виртуальной памяти?**

- a) Изоляцию процессов и предотвращение несанкционированного доступа
- b) Ускорение работы процессора
- c) Автоматическое обновление программ
- d) Сжатие данных в памяти

**Ответ: a**

**8. Что такое бит прав доступа в таблице страниц?**

- a) Флаг, указывающий разрешённые операции с страницей (чтение, запись, исполнение)
- b) Адрес страницы в памяти
- c) Размер страницы
- d) Тип процессора

**Ответ: a**

**9. Какая страница называется "страницей ядра"?**

- a) Страница, используемая только ядром ОС
- b) Страница, используемая пользователем
- c) Страница с данными пользователя
- d) Страница файла

**Ответ: a**

**10. Почему виртуальная память увеличивает безопасность ОС?**

- a) Потому что снижает скорость работы программ
- b) Потому что изолирует адресные пространства процессов друг от друга
- c) Потому что уменьшает потребление энергии
- d) Потому что ускоряет загрузку ОС

**Ответ: b**

**Примерное описание и критерии оценивания к итоговому проекту**

**Описание проекта:**

Итоговый проект направлен на комплексную проверку и закрепление знаний, полученных в ходе изучения ключевых аспектов архитектуры операционных систем. Студентам предлагается разработать прототип или модель компонентов операционной системы, отражающих основные механизмы и принципы, изученные в дисциплине. Проект должен продемонстрировать понимание и умение применять технологии разработки ядра, обработку прерываний, управление виртуальной памятью, планирование процессов, работу с внешними носителями данных, основы организации файловых систем и запуск пользовательских процессов.

В рамках проекта студентам необходимо:

- Создать минимальную среду или модуль, реализующий процесс загрузки ОС с

использованием загрузчика и инициализации ядра.

- Реализовать обработку аппаратных и программных прерываний с использованием таблицы векторов прерываний (IDT).
- Организовать механизм виртуальной памяти с таблицами страниц, обработкой page fault и базовым управлением памятью.
- Внедрить простой планировщик процессов, обеспечивающий переключение контекста и управление состояниями процессов.
- Обеспечить взаимодействие с внешним устройством хранения данных, реализовав базовые операции ввода-вывода с учетом асинхронности и прерываний.
- Спроектировать структуру файловой системы с каталогами, файлами и метаданными, реализовать алгоритмы размещения и управление свободным пространством.
- Обеспечить запуск пользовательских процессов с корректной инициализацией и мониторингом.

Проект может быть выполнен в форме программного модуля, эмулятора, расширения существующей ОС или учебного симулятора. Важно продемонстрировать интеграцию изученных тем и понимание взаимодействия компонентов ОС.

#### Критерии оценивания:

1. Понимание и применение концепций загрузки ОС
2. Обработка прерываний
3. Механизмы виртуальной памяти
4. Планирование процессов
5. Работа с внешними носителями данных
6. Организация файловой системы
7. Запуск и управление пользовательскими процессами
8. Качество реализации и интеграция компонентов
9. Демонстрация и защита проекта

#### Задания для промежуточной аттестации по дисциплине (модулю)

№ п/п	Задание	Ответ	Компетенция
1.	Какие языки программирования чаще всего используются для разработки ядра операционной системы? а) С и ассемблер б) Python и Java в) JavaScript и Ruby г) SQL и HTML	а	ОПК-4
2.	Что из перечисленного НЕ входит в этапы загрузки ОС? а) BIOS/UEFI б) Загрузчик (bootloader) в) Инициализация ядра г) Выполнение пользовательских приложений до загрузки ядра	д	УК-1
3.	Структура, которая связывает номера прерываний с обработчиками – это...	Таблица векторов прерываний / IDT	ПК-3
4.	Какой механизм используется для обработки ситуации, когда процесс обращается к странице памяти, отсутствующей в физической памяти? а) Маскирование прерываний б) Page fault (страничная ошибка) и обработка исключения в) Планирование процессов	б	ПК-3

	d) Буферизация данных		
5.	Какая из перечисленных задач НЕ относится к планировщику процессов? a) Переключение контекста между процессами b) Управление очередями процессов c) Загрузка исполняемых файлов в память d) Управление состояниями процессов	c	ОПК-5
6.	Что из перечисленного является примером асинхронного ввода-вывода? a) Ожидание завершения операции чтения с диска в основном потоке b) Синхронное копирование данных из буфера c) Последовательное чтение файла блокирующим вызовом d) Использование прерываний для уведомления о завершении операции	d	ОПК-4
7.	Какой алгоритм размещения данных в файловой системе использует индексные блоки?	Индексное размещение	ПК-3
8.	Что происходит при создании пользовательского процесса? a) Стираются все данные на диске b) Удаляется ядро ОС c) Выделяются ресурсы, копируется контекст, загружается исполняемый файл d) Открывается таблица векторов прерываний	c	УК-1
9.	Как называется таблица, связывающая номера прерываний с обработчиками?	IDT (Interrupt Descriptor Table)	ПК-3
10.	Какой тип ошибки возникает при обращении к отсутствующей странице памяти?	Page fault	ПК-3
11.	Как называется процесс переключения между разными процессами в ОС?	Контекстное переключение	ОПК-5
12.	Какой протокол или механизм используется для уведомления о завершении асинхронной операции ввода-вывода?	Прерывание	ОПК-4