

Документ подписан простой электронной подписью  
Информация о владельце:  
ФИО: Таскаев Сергей Валерьевич  
Должность: Ректор  
Дата подписания: 05.09.2025 10:59:19  
Уникальный программный ключ:  
04c19ed8bfb98f3b6cb77a486b9a3788b832324



МИНОБРНАУКИ РОССИИ  
Федеральное государственное бюджетное  
образовательное учреждение высшего образования  
«Челябинский государственный университет» (ФГБОУ ВО «ЧелГУ»)

Фонд оценочных средств для промежуточной аттестации по дисциплине (модулю) «Операционные системы»  
по направлению подготовки 09.03.03 «Прикладная информатика» направленности «ИТ-решения и  
технологии обработки данных в экономике» ФГБОУ ВО «ЧелГУ»

стр. 1

Фонд оценочных средств для промежуточной аттестации по дисциплине (модулю)  
**«Операционные системы»**

Направление подготовки (специальность)  
**09.03.03 «Прикладная информатика»**

Направленность (профиль)  
**«ИТ-решения и технологии обработки данных в экономике»**

Присваиваемая квалификация  
**Бакалавр**

Форма обучения  
**Очная**

Год набора  
**2025**

Челябинск, 2025 г.

**09.03.03 Прикладная информатика, ИТ-решения и технологии обработки данных в экономике, бакалавр, *Операционные системы*, 2025, очная**

**Фонд оценочных средств дисциплины (модуля) одобрен и рекомендован**  
Проректор по учебной работе утверждено 24.02.2025 А.А. Саламатов

Ученым советом института информационных технологий

Протокол заседания № 6 от 20.02.2025

Председатель Ученого совета  
института информационных  
технологий

согласовано

Ю. В. Петриченко

**Заседанием кафедры информационных технологий и экономической информатики**

Протокол заседания № 6 от 20.02.2025

И. о. заведующего кафедрой

согласовано

С.А. Скрипов

Автор (составитель)

С.А. Скрипов

**Структура рабочей программы соответствует приказу ректора ФГБОУ ВО «ЧелГУ» от «13» апреля 2021 г. № 247-1**



## Содержание

1. Паспорт фонда оценочных средств .....	3
2. Перечень формируемых компетенций .....	4
3. Содержание оценочных средств по дисциплине .....	6
3.1. Виды оценочных средств .....	6
3.2. Содержание оценочных средств .....	8
4. Порядок проведения и критерии оценивания промежуточной аттестации .....	19
4.1. Порядок проведения промежуточной аттестации .....	19
4.2. Критерии оценивания промежуточной аттестации по видам оценочных средств .....	19
4.3. Результаты промежуточной аттестации и уровни сформированности компетенций.....	19



## **1. Паспорт фонда оценочных средств**

Направление подготовки: 09.03.03 Прикладная информатика

Направленность: ИТ-решения и технологии обработки данных в экономике

Дисциплина: Операционные системы.

Семестры: 3

Форма промежуточной аттестации: экзамен

Для оценивания результатов обучения используется балльно-рейтинговая система.



## 2. Перечень формируемых компетенций

Изучение дисциплины «Операционные системы» направлено на формирование компетенций, приведённых в 1.

Таблица 1. Результаты обучения по дисциплине.

Коды компетенции и согласно ФГОС (ОПОП ВО)	Содержание компетенций согласно ФГОС (ОПОП ВО)	Индикаторы достижения компетенции согласно ОПОП	Перечень планируемых результатов обучения по дисциплине
1	2	3	4
ОПК-2	Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности;	ОПК-2.1. Демонстрирует знание инструментальных средств, готового программного обеспечения и библиотек; знаком с содержанием Единого реестра российских программ ОПК-2.2. Умеет выбирать и использовать инструментальные средства, готовое программное обеспечение и библиотеки ОПК-2.3. Имеет практический опыт решения задач анализа, интеграции различных типов программного обеспечения и сетевых коммуникаций	Знать: методы использования инструментальных средств для автоматизации работы с операционной оболочкой; программное обеспечение и библиотеки для системного программирования Уметь: выбирать и использовать инструментальные средства, готовое программное обеспечение и библиотеки для работы с оболочкой и системного программирования Владеть: навыками решения задач анализа, интеграции программного обеспечения для различных операционных систем
ОПК-5	Способен устанавливать программное и аппаратное обеспечение для информационных и автоматизированных систем;	ОПК-5.1. Обладает базовыми знаниями основ устройства и администрирования программного и аппаратного обеспечения информационных систем и баз данных ОПК-5.2. Демонстрирует умения устанавливать программное обеспечение информационных систем и баз данных ОПК-5.3. Имеет практический опыт установки и сопровождения программного и аппаратного обеспечения информационных систем и баз данных	Знать: основы устройства и администрирования операционных систем Уметь: устанавливать программное обеспечение с использованием возможностей операционной оболочки Владеть: навыками установки программного обеспечения из репозитория



ПК-4	Способен решать прикладные задачи сбора, обработки и анализа данных предприятия с применением технологий баз данных, современных языков программирования, сетевых технологий, технологий бизнес-аналитики и машинного обучения	ПК-4.1. Демонстрирует знание сетевых технологий, языков программирования, баз данных и технологий бизнес-аналитики и машинного обучения ПК-4.2. Демонстрирует умение осуществлять выбор прикладного программного обеспечения, баз данных, сетевых технологий и технологий бизнес-аналитики и машинного обучения ПК-4.3. Имеет практический опыт использования современных языков программирования, баз данных и технологий бизнес-аналитики и машинного обучения	Знать: основы операционных систем и сетевых технологий Уметь: решать прикладные задачи с использованием современных операционных систем Владеть: навыками использования операционных систем
------	--	--	---



### 3. Содержание оценочных средств по дисциплине

#### 3.1. Виды оценочных средств

Таблица 2. Виды оценочных средств.

№ п/п	Код компетенции/ планируемые результаты обучения	Контролируемые темы/ разделы	Наименование оценочного средства для текущего контроля	Наименование оценочного средства на промежуточной аттестации/№ задания
1	ОПК-2.1. Демонстрирует знание инструментальных средств, готового программного обеспечения и библиотек; знаком с содержанием Единого реестра российских программ Знать: методы использования инструментальных средств для автоматизации работы с операционной оболочкой; программное обеспечение и библиотеки для системного программирования	Понятие, история классификация ОС. Управление памятью. Файловые системы. Мультипрограммирование Операционные оболочки	Практическая работа	Задания теста № 12-23, 27-52, 63-69
2	ОПК-2.2. Умеет выбирать и использовать инструментальные средства, готовое программное обеспечение и библиотеки Уметь: выбирать и использовать инструментальные средства, готовое программное обеспечение и библиотеки для работы с оболочкой и системного программирования	Понятие, история классификация ОС. Управление памятью. Файловые системы. Мультипрограммирование Операционные оболочки	Практическая работа	Задания теста № 1-11, 24-26, 63-69
3	ОПК-2.3. Имеет практический опыт решения задач анализа, интеграции различных типов программного обеспечения и сетевых коммуникаций Владеть: навыками решения задач анализа, интеграции программного обеспечения для различных операционных систем	Понятие, история классификация ОС. Управление памятью. Файловые системы. Мультипрограммирование Операционные оболочки	Практическая работа	Задания теста № 1-11, 53-62
4	ОПК-5.1. Обладает базовыми знаниями основ устройства и администрирования	Понятие, история классификация ОС. Управление памятью. Файловые системы.	Практическая работа	Задания теста № 12-23, 27-52, 63-69



	программного и аппаратного обеспечения информационных систем и баз данных Знать: основы устройства и администрирования операционных систем	Мультипрограммирование Операционные оболочки		
5	ОПК-5.2. Демонстрирует умения устанавливать программное обеспечение информационных систем и баз данных Уметь: устанавливать программное обеспечение с использованием возможностей операционной оболочки	Понятие, история классификация ОС. Управление памятью. Файловые системы. Мультипрограммирование Операционные оболочки	Практическая работа	Задания теста № 1-11, 24-26, 63-69
6	ОПК-5.3. Имеет практический опыт установки и сопровождения программного и аппаратного обеспечения информационных систем и баз данных Владеть: навыками установки программного обеспечения из репозитория	Понятие, история классификация ОС. Управление памятью. Файловые системы. Мультипрограммирование Операционные оболочки	Практическая работа	Задания теста № 1-11, 53-62
7	ПК-4.1. Демонстрирует знание сетевых технологий, языков программирования, баз данных и технологий бизнес-аналитики и машинного обучения Знать: основы операционных систем и сетевых технологий	Понятие, история классификация ОС. Управление памятью. Файловые системы. Мультипрограммирование Операционные оболочки	Практическая работа	Задания теста № 12-23, 27-52, 63-69
8	ПК-4.2. Демонстрирует умение осуществлять выбор прикладного программного обеспечения, баз данных, сетевых технологий и технологий бизнес-аналитики и машинного обучения Уметь: решать прикладные задачи с использованием современных операционных систем	Понятие, история классификация ОС. Управление памятью. Файловые системы. Мультипрограммирование Операционные оболочки	Практическая работа	Задания теста № 1-11, 24-26, 63-69
9	ПК-4.3. Имеет практический опыт использования современных языков программирования, баз данных и технологий бизнес-аналитики и машинного обучения Владеть: навыками использования операционных систем	Понятие, история классификация ОС. Управление памятью. Файловые системы. Мультипрограммирование Операционные оболочки	Практическая работа	Задания теста № 1-11, 53-62



Типовые задания, критерии и показатели оценивания в рамках текущего контроля представлены в рабочей программе дисциплины (модуля). Полные комплекты оценочных средств и контрольно-измерительных материалов хранятся на кафедре.

### 3.2. Содержание оценочных средств

#### База тестовых вопросов

№ п/п	Формулировка вопроса	Варианты ответов (полужирным шрифтом – верные варианты)
1.	Команда echo off	<b>a. Отключит вывод команд на экран во время выполнения</b> b. Выключит режим раннего связывания для переменных окружения c. Позволяет выводить данные не на экран, а в файл
2.	В командной строке Windows команда dir позволяет:	<b>a. Получить список файлов и подкаталогов</b> b. Изменить атрибуты файлов и каталогов c. Переименовать файлы и каталоги d. Переименовать только файлы (но не каталоги)
3.	В командной строке Windows команда md позволяет:	a. Получить список файлов и подкаталогов b. Изменить атрибуты файлов и каталогов c. Удалить файл <b>d. Создать каталог</b>
4.	Для получения первого параметра, переданного в командный файл Windows (.bat) необходимо использовать следующее выражение:	<b>a. %1</b> b. %0 c. %~x1 d. \$PARAM[1] e. \$PARAM[0]
5.	В командном файле (.bat) оператор > используется:	<b>a. Для перенаправления вывода в файл с перезаписью указанного файла</b> b. Для перенаправления вывода в файл с записью в конец указанного файла c. Для извлечения параметров, переданных в командной строке
6.	В командном файле (.bat) оператор >> используется:	a. Для перенаправления вывода в файл с перезаписью указанного



		файла <b>b. Для перенаправления вывода в файл с записью в конец указанного файла</b> c. Для извлечения параметров, переданных в командной строке
7.	Для использования связывания времени выполнения (то есть чтобы в составных выражениях подставлялись актуальные значения переменных окружения) в командном файле Windows необходимо использовать следующий синтаксис:	<b>a. !k! для переменной k</b> b. %k% для переменной k c. &k& для переменной k d. ?k? для переменной k
8.	Следующий сценарий: SET /a k=0 FOR /R %%i IN (*) do set /a k=k+1 echo %k%	<b>a. Подсчитывает количество файлов в текущем каталоге, включая файлы в подкаталогах</b> b. Выведет на экран все имена файлов в текущем каталоге, не включая подкаталоги c. Выведет на экран все имена файлов в текущем каталоге, включая подкаталоги d. Выведет на экран все файлы, в именах которых есть "i"
9.	Следующая команда в сценарии: dir /B *.doc >> 1.txt	<b>a. Добавит список файлов с расширением "doc" в файл "1.txt"</b> b. Изменит расширения файлов с "doc" на "txt" c. Содержит ошибку d. Переименует файл "*.doc" в "1.txt"
10.	Следующая команда в сценарии: attrib +R %1	<b>a. Добавит атрибут "Только чтение" для файла, имя которого передано в первом параметре</b> b. Добавит атрибут "Только чтение" для всех файлов, имена которых содержат "1" c. Содержит ошибку d. Запросит у пользователя имя одного файла и атрибут для установки
11.	Следующая команда в сценарии (.bat): echo %0	<b>a. Выведет на экран имя сценария</b> b. Выведет на экран %0 c. Выведет на экран 0 d. Выведет на экран первый переданный сценарию



		параметр
12.	Microsoft Windows использует:	<b>a. Страничное распределение памяти</b> b. Сегментное распределение памяти c. Распределение памяти перемещаемыми разделами
13.	Виртуальная память позволяет:	<b>a. Защитить процессы друг от друга</b> b. Обеспечить предсказуемость времени доступа к памяти c. Обойтись без аппаратной поддержки при управлении памятью
14.	Виртуальная страница при страничном распределении памяти:	<b>a. Располагается либо в физической памяти, либо во внешней памяти, либо отсутствует</b> b. Может располагаться исключительно во внешней памяти c. Не может отсутствовать, все адреса виртуального адресного пространства всегда доступны
15.	Для преобразования виртуального адреса в физический при страничном распределении нет необходимости учитывать:	<b>a. Тип содержимого страницы, например исполняемый код или данные</b> b. Размер страницы c. Данные из таблицы страниц процесса d. Смещение внутри виртуальной страницы
16.	Модули оперативной памяти на материнской плате, это:	<b>a. Физическая память</b> b. Виртуальная память c. Внешняя память d. Регистры процессора
17.	Недостатком виртуальной памяти является:	<b>a. Непредсказуемое время доступа к памяти</b> b. Процесс, требующий памяти больше, чем реально есть в системе загрузить нельзя c. Фрагментация памяти
18.	Недостатком оверлеев является:	<b>a. Необходимость для прикладной программы самостоятельно</b>



		<b>реализовывать управление памятью</b> b. Необходимость целиком загружать и выгружать образ процесса c. Процесс, требующий памяти больше, чем реально есть в системе, загрузить нельзя
19.	Почему размер страницы при страничном распределении выбирается кратным степени двойки?	<b>a. Так из виртуального адреса проще выделить номер страницы и смещение</b> b. Это просто традиция, которая на практике не дает никаких преимуществ c. Размер страницы выбирается по размеру кластера на диске
20.	Преимуществом виртуальной памяти является:	<b>a. Возможность скрыть от прикладной программы реальную организацию памяти</b> b. Простота реализации c. Нет необходимости аппаратной поддержки
21.	При использовании виртуальной памяти:	<b>a. Программа может использовать памяти больше, чем объем физической (оперативной) памяти</b> b. Доступны средства для создания виртуальной реальности c. Размер выделенной памяти фиксируется в момент запуска программы и в дальнейшем не может быть изменен
22.	При страничном распределении памяти:	<b>a. Программист работает с одним линейным адресным пространством</b> b. Исполняемый код и данные могут храниться в отдельных адресных пространствах c. Адресное пространство приложения не может превышать размеры физической памяти
23.	При страничном распределении памяти:	<b>a. Память делится на страницы одинакового</b>



		<b>размера</b> b. Память делится на страницы произвольного размера c. Размер любой страницы может измениться в процессе работы приложения
24.	В системе используется страничное распределение памяти. Процесс имеет 8-битное адресное пространство. Размер страницы - 4 байта. Сколько всего страниц у процесса?	<b>a. 64</b> b. 16 c. 256 d. 4 e. 32
25.	В системе используется страничное распределение памяти. Процесс имеет 8-битное адресное пространство. Размер страницы - 8 байт. Сколько всего страниц у процесса?	a. 64 b. 16 c. 256 d. 4 <b>e. 32</b>
26.	В системе используется страничное распределение памяти. Процесс имеет 8-битное адресное пространство. Размер страницы - 32 байта. Сколько всего страниц у процесса?	a. 64 b. 16 c. 256 <b>d. 8</b> e. 32
27.	Свопинг предполагает, что:	<b>a. Образы процессов выгружаются на диск и возвращаются в оперативную память целиком</b> b. Внешняя память не используется c. Процесс самостоятельно управляет распределением памяти
28.	Страничное распределение памяти предполагает:	<b>a. Использование внешней памяти</b> b. Возможность управлять физической памятью для каждого процесса c. Деление адресного пространства на части произвольного размера
29.	Алгоритмы планирования потоков, где решения о смене и выборе активного потока принимаются операционной системой называются:	<b>a. Вытесняющими</b> b. Невытесняющими c. Системными
30.	Алгоритмы планирования потоков, где потоку позволяет выполняться, пока он сам, по собственной инициативе не отдаст управление операционной системе, называются:	a. Вытесняющими <b>b. Невытесняющими</b> c. Системными
31.	Критическая секция это:	<b>a. Участок исполняемого</b>



		<b>кода</b> b. Данные c. Конструктивная часть процессора
32.	Недостатком систем разделения времени является:	<b>a. Накладные расходы при переключении между задачами</b> b. Отсутствие интерактивности c. Невозможность распараллелить вычисления и ввод-вывод
33.	Особенностью систем реального времени является:	<b>a. При проектировании закладывается запас вычислительной мощности</b> b. Предназначены для решения вычислительных задач c. Основной критерий при проектировании - удобство пользователя
34.	Преимуществом систем пакетной обработки является:	<b>a. Хорошая пропускная способность (которая выражается в количестве задач, выполняемых за единицу времени)</b> b. Интерактивность c. Пригодность для систем реального времени
35.	Выберите правильное утверждение:	<b>a. Несколько потоков могут принадлежать одному процессу</b> b. В настоящее время процесс и поток - это одно и то же c. Процесс не может иметь более одного потока
36.	Возможность параллельного выполнения вычислений и ввода-вывода позволяет:	<b>a. Уменьшить суммарное время выполнения нескольких задач</b> b. Уменьшить время выполнения каждой задачи c. Использовать центральный процессор для прямого управления вводом-выводом
37.	В операционной системе Windows каталоги (например при использовании файловой системы FAT):	<b>a. Образуют древовидную структуру</b> b. Образуют сетевую



		структуру с. Изолированы, один каталог не может находиться внутри другого
38.	Особенностью использования файловых систем в ОС Windows является:	<b>а. Файловые системы используются независимо (автономно)</b> б. Файловые системы монтируются в единое дерево с. Может использоваться ровно одна файловая система
39.	В файловой системе FAT в записи каталога о файле хранится:	<b>а. Номер первого кластера</b> б. Список всех кластеров с. Смещение начала файла в байтах относительно начала области данных
40.	Если кластер в файловой системе FAT не последний в файле, в соответствующем ему индексном указателе хранится:	<b>а. Номер следующего кластера</b> б. Номер предыдущего кластера с. Специальное значение "кластер занят"
41.	Журналирование файловой системы позволяет:	<b>а. Сохранить целостность файловой системы</b> б. Узнать, кто и когда создавал, удалял и редактировал файлы с. Сохранить данные, находящиеся в кэше при отключении питания
42.	Монтирование предполагает:	<b>а. Объединение нескольких файловых систем в единое дерево</b> б. Удаление всех файлов и создание новой файловой системы с. Создание специальных файлов, ассоциированных с устройствами ввода-вывода
43.	Особенностью файловой системы FAT является:	<b>а. Корневой каталог занимает фиксированную область</b> б. Файл всегда располагается в смежных кластерах с. Размер кластера не фиксирован и выбирается динамически, в процессе



		записи файла на диск.
44.	Особенностью файловой системы FAT является:	<b>а. Файл всегда занимает целое число кластеров, поэтому дисковое пространство обычно используется неоптимально</b> б. В одном кластере может храниться два и более файлов с. Кластеры используются только для хранения каталогов
45.	Особенностью жесткой ссылки является:	<b>а. Файл не будет удален, пока имеется хотя бы одна жесткая ссылка</b> б. Различают первичную жесткую ссылку и второстепенные с. Файл полностью удаляется, если удалить любую жесткую ссылку
46.	Особенностью ярлыка Windows является:	<b>а. С точки зрения файловой системы ярлык является обычным файлом</b> б. При удалении ярлыка файл также будет удален с. Файл не будет удален, пока есть хотя бы один ярлык, указывающий на него
47.	Отличие FAT16 от FAT32 состоит в:	<b>а. Разрядности индексных указателей</b> б. Разрядности операционной системы, под которой используется данная файловая система с. Размёре кластера: 16 или 32 байта
48.	Для получения списка кластеров файла в FAT необходимо:	<b>а. Восстановить цепочку кластеров с помощью таблицы индексных указателей (FAT-таблицы)</b> б. Восстановить цепочку кластеров, прочитав по очереди их содержимое с. Прочитать запись в каталоге для данного файла
49.	При появлении испорченной области на жестком диске,	<b>а. Соответствующие</b>



	использующем файловую систему FAT:	<b>кластеры помечаются в таблице индексных указателей специальным значением</b> b. В особую область добавляется список плохих кластеров c. Невозможно далее использовать диск
50.	Примером корневого каталога в Windows в файловой системе FAT является:	a. C:\ b. C:\WINDOWS c. C:\SYSTEM
51.	Фрагментация в файловой системе может означать, что:	a. <b>Соседние участки файла располагаются в несмежных участках диска</b> b. Файл хранится в двух и более смежных кластерах c. Файл занимает меньше одного кластера
52.	Что такое каталоги?	a. <b>это системные файлы, обеспечивающие поддержку структуры файловой системы</b> b. это фиктивные файлы, ассоциированные с устройствами ввода-вывода c. циклические буферы, позволяющие выходной файл одной программы соединить со входным файлом другой программы d. это обычные файлы, отображенные на адресное пространство процесса по указанному виртуальному адресу
53.	В JScript сценарии имеется конструкция:for(i in B) S+=B[i];B данном случае:	a. <b>К переменной S будет добавлена сумма всех элементов массива B</b> b. К переменной S будет добавлено значение первого элемента массива B c. Используется неверный синтаксис для цикла for
54.	Главным объектом WSH, позволяющим создавать другие объекты, является:	a. <b>WScript</b> b. WMain c. Функция main()
55.	Для вывода значения переменной K в JScript можно использовать конструкцию:	a. <b>WScript.Echo(K);</b> b. Echo K; c. printf("%s",K);



56.	Для запуска приложения из WSH сценария можно использовать объект:	<b>a. WScript.Shell</b> b. Scripting.FileSystemObject c. WScript.StdOut
57.	Для получения имени пользователя в WSH сценарии можно использовать объект:	<b>a. Wscript.Network</b> b. Scripting.FileSystemObject c. WScript.StdOut
58.	Для создания объекта в WSH используется функция:	<b>a. WScript.CreateObject</b> b. WScript.Echo c. Main.Create
59.	Для запуска WSH сценария C:\test.js в консольном режиме, в командной строке необходимо набрать:	<b>a. cscript C:\test.js</b> b. wscript C:\test.js c. C:\test.js
60.	Особенностью переменных JScript является:	<b>a. Тип присваивается при записи значения</b> b. Необходимо указать тип при объявлении c. В переменных нельзя хранить текст
61.	Преимуществом WSH по сравнению с пакетными файлами Windows является:	<b>a. Наличие полноценных циклов и условий</b> b. Возможность выполнить несколько консольных команд из одного сценария c. Возможность редактировать сценарий в текстовом редакторе
62.	Для того, чтобы в WSH запросить у пользователя данные в консоли, можно использовать функцию:	<b>a. WScript.StdIn.ReadLine</b> b. scanf c. Main.ReadFromConsole
63.	Выполнение какого требования к операционной системе позволяет пользователю выполнить свои приложения в новой операционной системе?	<b>a. совместимости</b> b. безопасности c. предсказуемости d. масштабируемости
64.	Микроядерная архитектура операционной системы предполагает, что:	<b>a. В режиме ядра работает минимально необходимая часть операционной системы</b> b. Возможности операционной системы сильно ограничены c. Отсутствует аппаратная поддержка операционной системы
65.	Операционные системы с многослойной архитектурой, по сравнению с монолитной, имеют следующее преимущество:	<b>a. Проще отлаживать, дорабатывать, модифицировать</b>



		<p>b. Имеют большую производительность c. Проще проектировать</p>
66.	Операционная оболочка (в отличие от операционной среды и операционной системы) определяет:	<p><b>a. Интерфейс пользователя</b> b. Особенности управления ресурсами системы c. Интерфейс прикладного программирования</p>
67.	Операционная система (в отличие от операционной среды и операционной оболочки) определяет:	<p><b>a. Особенности управления ресурсами системы</b> b. Интерфейс пользователя c. Интерфейс прикладного программирования</p>
68.	Операционная среда определяет:	<p><b>a. Интерфейс прикладного программирования</b> b. Интерфейс пользователя c. Особенности управления ресурсами системы</p>
69.	Операционная система выполняет следующую функцию:	<p><b>a. Все перечисленное</b> b. Управление памятью c. Управление процессами d. Управление внешними устройствами</p>



#### 4. Порядок проведения и критерии оценивания промежуточной аттестации

##### 4.1. Порядок проведения промежуточной аттестации

Экзамен проводится в виде тестирования. Студент должен ответить на вопросы закрытого типа, которые предполагают выбор вариантов ответа. Всего 20 тестовых вопросов. Продолжительность теста – 35 минут.

##### 4.2. Критерии оценивания промежуточной аттестации по видам оценочных средств

Тест формируется в системе электронного обучения MOODLE.  
Максимальный балл за тест — 100 баллов.

Оценка	Отлично/ Зачтено	Хорошо/ зачтено	Удовлетворитель но/зачтено	Неудовлетворительно/ незачтено
Баллы	100-90 баллов	89-75 баллов	74-60 баллов	59-0 баллов
Уровень освоения проверяемых компетенций	высокий	средний	базовый	недостаточный

##### 4.3. Результаты промежуточной аттестации и уровни сформированности компетенций

Итоговый балл рассчитывается по формуле:

$$S = ((P * 50) / n) + T / 1.5$$

Здесь:

P - Сумма баллов за практические работы

n - Количество практических работ

T - Баллы за итоговый тест

0-59 баллов – неудовлетворительно/незачтено;

60-74 баллов – удовлетворительно/зачтено;

75-89 баллов – хорошо/зачтено;

90-100 баллов – отлично/зачтено;

Особенности проведения процедуры оценивания результатов обучения инвалидов и лиц с ограниченными возможностями здоровья обозначены в рабочей программе дисциплины (модуля).

Уровни сформированности компетенций определяется следующим образом:

1. Высокий уровень сформированности компетенций соответствует оценке отлично:  
- предполагает формирование компетенций на высоком уровне;



- знание теоретических разделов изучаемой дисциплины на уровне не ниже оценки отлично;
  - студент умеет применять на практике знания, полученные в рамках изучения дисциплины
  - формируются навыки использования теоретических и практических разделов дисциплины для решения задач профессиональной деятельности;
2. Средний уровень соответствует оценке хорошо:
- предполагает формирование компетенций на среднем уровне;
  - знание теоретических разделов изучаемой дисциплины на уровне не ниже оценки хорошо;
  - студент умеет применять знания, полученные в рамках изучения дисциплины, для решения задач профессиональной деятельности;
3. Базовый уровень соответствует оценке удовлетворительно:
- предполагает формирование компетенций на базовом уровне;
  - знание теоретических разделов изучаемой дисциплины на уровне не ниже оценки удовлетворительно;
4. Недостаточный уровень соответствует оценке неудовлетворительно.