

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Гарант Максим Алексеевич
Должность: Ректор
Дата подписания: 10.11.2023 09:50:14
Уникальный программный ключ:
7708e7a47e66a8ee02711b298d7e78bd1e40bf88

Приложение
к рабочей программе дисциплины

**ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ДЛЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ
ПО ДИСЦИПЛИНЕ (МОДУЛЮ)**

Системное программное обеспечение

(наименование дисциплины (модуля))

Направление подготовки / специальность

09.03.01 Информатика и вычислительная техника

(код и наименование)

Направленность (профиль) / специализация

Проектирование АСОИУ на транспорте

(наименование)

Содержание

1. Пояснительная записка.
2. Типовые контрольные задания или иные материалы для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих уровень формирования компетенций.
3. Методические материалы, определяющие процедуру и критерии оценивания формирования компетенций при проведении промежуточной аттестации.

1. Пояснительная записка

Цель промежуточной аттестации – оценивание промежуточных и окончательных результатов обучения по дисциплине, обеспечивающих достижение планируемых результатов освоения образовательной программы.

Формы промежуточной аттестации: 5 семестр зачет; 6 семестр курсовая работа, экзамен.

Перечень компетенций, формируемых в процессе освоения дисциплины

Код и наименование компетенции	Код индикатора достижения компетенции
ПК-2: Способен осуществлять концептуальное, функциональное и логическое проектирование систем среднего и крупного масштаба и сложности	ПК-2.1: Использовать существующие типовые решения и шаблоны проектирования программного обеспечения

Код и наименование индикатора достижения компетенции	Результаты обучения по дисциплине	Оценочные материалы
ПК-2.1: Использовать существующие типовые решения и шаблоны проектирования программного обеспечения	Обучающийся знает: основные модели системных процессов	Вопросы 1–20
	Обучающийся умеет: разрабатывать программы на уровне моделей системных процессов	Задания 1.1–2.1
	Обучающийся владеет: шаблонами проектирования системного программного обеспечения	Задания 2.2–6.2

5 семестр

Промежуточная аттестация (зачет) проводится в одной из следующих форм:

- 1) Собеседование
- 2) выполнение заданий в ЭИОС СамГУПС.

6 семестр

Промежуточная аттестация (курсовая работа) проводится в одной из следующих форм:

- 1) Собеседование
- 2) выполнение заданий в ЭИОС СамГУПС.

6 семестр

Промежуточная аттестация (Экзамен) проводится в одной из следующих форм

- 1) Собеседование;
- 2) выполнение заданий в ЭИОС СамГУПС.

2. Типовые¹ контрольные задания или иные материалы для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих уровень сформированных компетенций

2.1 Типовые вопросы для оценки знаний образовательного результата

Проверяемый образовательный результат:

Код и наименование индикатора достижения компетенции	Образовательный результат
ПК-2.1: Использовать существующие типовые решения и шаблоны проектирования программного обеспечения	Обучающийся знает: основные модели системных процессов
<p><i>Вопросы:</i></p> <ol style="list-style-type: none"> 1. Что включает в себя понятие "программное обеспечение" 2. Назовите и характеризуйте основные категории программного обеспечения. 3. В чем отличие прикладных программ от системных и инструментальных 4. Что входит в системное программное обеспечение 5. Что называется утилитой 6. Для чего предназначены драйвера 7. Какое назначение текстового редактора 8. Для какой цели применяют графические редакторы 9. В чем состоит назначение операционной системы 10. Характеризуйте основные классы операционных систем. 	
<p><i>Вопросы:</i></p> <ol style="list-style-type: none"> 11. Опишите процесс начальной загрузки операционной системы в оперативную память компьютера. 12. Назовите основные разновидности программ-утилит и дайте им краткую характеристику. 13. Какой вид интерфейса удобнее для пользователя - командный или графический 14. Охарактеризуйте основные особенности операционных систем семейства Windows. 15. Назовите функциональные возможности табличного процессора. 16. Дайте определения интегрированного пакета программ. 17. Каково назначение сетевого программного обеспечения 18. Сколько версий операционной системы Windows Вы знаете 19. Что такое окно? Какие существуют разновидности окон в Windows 20. Какие основные элементы окна 	

2.2 Типовые задания для оценки навыков образовательного результата

Проверяемый образовательный результат:

Код и наименование индикатора достижения компетенции	Образовательный результат
ПК-2.1: Использовать существующие типовые решения и шаблоны проектирования программного обеспечения	Обучающийся умеет: разрабатывать программы на уровне моделей системных процессов
<p><i>Задания:</i></p> <p>1.1 Тема «Разбиение на лексемы транслируемой команды» Задание: определить лексический набор команды как упакованный массив или строковый тип. Разделить строку на лексемы по указанной группе разделителей</p> <p>1.2 Тема «Построение бинарного несбалансированного дерева» Задание: определить структуру записи листа дерева с учетом идентификатора лексемы, ее системного признака и полей организации доступа к потомкам; построить дерево и организовать вывод ассоциативной таблицы связей.</p> <p>2.1 Тема «Одномерный поток» Задание: конструктор потока, запуск потока, индикация этапов задачи до потока во время его работы и после завершения потока.</p>	
ПК-2.1: Использовать	Обучающийся владеет: шаблонами проектирования системного программного

¹Приводятся типовые вопросы и задания. Оценочные средства, предназначенные для проведения аттестационного мероприятия, хранятся на кафедре в достаточном для проведения оценочных процедур количестве вариантов. Оценочные средства подлежат актуализации с учетом развития науки, образования, культуры, экономики, техники, технологий и социальной сферы. Ответственность за нераспространение содержания оценочных средств среди обучающихся университета несут заведующий кафедрой и преподаватель – разработчик оценочных средств.

существующие решения и проектирования программного обеспечения	типовые шаблоны	обеспечения
<p><i>Задания:</i></p> <p>2.2 Вывод потоком на консоль собственного и системного идентификатора; вывод имени потока и основных характеристик его системных ресурсов (адрес, приоритет и т.п.)</p> <p><i>Содержание задания:</i></p> <ul style="list-style-type: none"> - <i>подготовить конструкторы асинхронных потоков</i> - <i>передать не менее 2-х параметров в процедуры потоков посредством объекта-структуры</i> - <i>в составе параметров определить имя или идентификатор потока</i> - <i>считать системный идентификатор потока, приоритет потока. Вывести информационное сообщение на консоль.</i> <p>3.1 Организация потоков в массив: регистрация потоков в динамической коллекции с распознаванием идентификатора или имени потока при прохождении по коллекции независимым итератором</p> <p>3.2 Тема «Параллельные асинхронные потоки»</p> <p>Задание: конструктор фиксированного числа потоков, запуск потоков, идентификация асинхронной работы потоков; передача структуры с параметрами потока по нетипированному адресу.</p>		
<p><i>Задания:</i></p> <p>4.1 Тема «Статическая таблица адресации»</p> <p>Задание: Построение статической таблицы адресации методом перевода поискового дерева в индексируемую таблицу лексем со статическими адресами. Перевод осуществлять для элементов-лексем не вызывающих коллизии по определенным условиям</p> <p><i>Содержание задания:</i></p> <ul style="list-style-type: none"> - <i>подготовить несбалансированное дерево;</i> - <i>пойти по дереву и выбрать только идентификаторы словаря</i> - <i>организовать статическую таблицу в виде коллекции записей, содержащих идентификатор и точку входа для списка имен, вызвавших коллизии</i> - <i>организовать переход из дерева поиска в таблицу по индексу идентификатора.</i> <p>4.2 Тема «Динамическая таблица адресации»</p> <p>Задание: Перевести лексем-коллизии в связанные списки, точками входа которых являются лексема совпадающих символов. Организовать поиск команд оператора по всем динамическим структурам транслятора с заключением о синтаксической правильности каждой введенной лексемы</p> <p>5.1 Тема «Параллельные синхронные потоки»</p> <p>Задание: Ознакомиться с принципами организации работы с битовыми семафорами. Объявить семафор и отладить синхронизируемый управляемый запуск потоками. Тест на синхронность по индикации вывода сообщений потоков на консоль</p>		
<p><i>Задания:</i></p> <p>5.2 Использовать семафоры, отличные от битового; организовать собственную схему (отличную от случайного выбора элемента последовательности) диспетчеризации передачи управления потокам</p> <p><i>Содержание задания:</i></p> <ul style="list-style-type: none"> - <i>изучить состояния семафоров, отличных от бинарного</i> - <i>выбрать все константы состояния семафора и составить кейс-оператор вывода соответствующего константам сообщения</i> - <i>составить схему диспетчеризации и показать синхронность управляемых семафором процессов</i> <p>6.1 Регистрация потоков в динамической коллекции с распознаванием идентификатора или имени потока при прохождении по коллекции независимым итератором</p> <p>6.2 Тема «Обмен данными между потоками»</p> <p>Организовать обмен данными между потоками средствами используемого языка программирования. Можно использовать спецсредства библиотек или разработать собственный метод обмена</p>		

2.3. Перечень вопросов для подготовки обучающихся к промежуточной аттестации

Вопросы к зачету:

1. Понятие операционной среды.
2. Понятия вычислительного процесса и ресурса. Основные виды ресурсов.
3. Диаграмма состояний процесса.
4. Процессы и среды. Как они соотносятся?
5. Прерывания.

6. Планирование и диспетчеризация процессов и задач. Стратегии планирования.
7. Дисциплины диспетчеризации. Качество диспетчеризации и гарантии обслуживания.
8. Диспетчеризация задач с использованием динамических приоритетов.
9. Память и отображения, виртуальное адресное пространство.
10. Простое непрерывное распределение и распределение с перекрытием (оверлейные структуры).
11. Распределение статическими и динамическими разделами. Разделы с фиксированными границами. Разделы с подвижными границами.
12. Сегментная, страничная и сегментно-страничная организация памяти. Сегментный способ организации виртуальной памяти. Страничный способ организации виртуальной памяти. Сегментно-страничный способ организации виртуальной памяти.
13. Распределение оперативной памяти в MS-DOS.
14. Распределение оперативной памяти в Microsoft Windows 95/98/Me.
15. Распределение оперативной памяти в Microsoft Windows NT/2000/XP.
16. Реальный и защищенный режимы работы процессора. Адресация в 32-разрядных микропроцессорах i80x86 при работе в защищенном режиме. Поддержка сегментного способа организации виртуальной памяти. Поддержка страничного способа организации виртуальной памяти.
17. Режим виртуальных машин для исполнения приложений реального режима. Защита адресного пространства задач.
18. Языки и цепочки символов. Способы задания языков. Цепочки символов. Операции над цепочками символов. Понятие языка. Формальное определение языка. Способы задания языков.
19. Синтаксис и семантика языка. Особенности языков программирования.
20. Определение грамматики. Форма Бэкуса—Наура. Понятие о грамматике языка. Формальное определение грамматики. Принцип рекурсии в правилах грамматики. Другие способы задания грамматик.
21. Классификация грамматик. Четыре типа грамматик по Хомскому.
22. Классификация языков.
23. Цепочки вывода. Сентенциальная форма грамматики. Язык, заданный грамматикой. Левосторонний и правосторонний вывод. Дерево вывода. Методы построения дерева вывода.
24. Проблемы однозначности и эквивалентности грамматик. Однозначные и неоднозначные грамматики. Эквивалентность и преобразование грамматик. Правила, задающие неоднозначность в грамматиках.
25. Распознаватели. Задача разбора. Общая схема распознавателя. Виды распознавателей. Классификация распознавателей по типам языков. Задача разбора (постановка задачи).
26. Трансляторы, компиляторы и интерпретаторы — общая схема работы. Определение транслятора, компилятора, интерпретатора.
27. Этапы трансляции. Общая схема работы транслятора. Понятие прохода. Многопроходные и однопроходные компиляторы.
28. Интерпретаторы. Особенности построения интерпретаторов.
29. Таблицы идентификаторов. Организация таблиц идентификаторов. Назначение и особенности построения таблиц идентификаторов. Простейшие методы построения таблиц идентификаторов. Построение таблиц идентификаторов по методу бинарного дерева.
30. Хэш-функции и хэш-адресация. Комбинированные способы построения таблиц идентификаторов.
31. Лексические анализаторы (сканеры). Принципы построения сканеров.
32. Назначение лексического анализатора. Принципы построения лексических анализаторов.
33. Построение лексических анализаторов. Автоматизация построения лексических анализаторов.
34. Назначение таблиц идентификаторов
35. Принципы организации таблиц идентификаторов
36. Простейшие методы построения таблиц идентификаторов
37. Построение таблиц идентификаторов по методу бинарного дерева
38. Хэш-функции и хэшадресация
39. Хэш-адресация с рехэшированием
40. Хэш-адресация с использованием метода цепочек
41. Комбинированные способы построения таблиц идентификаторов
42. Синтаксические анализаторы.
43. Синтаксически управляемый перевод. Основные принципы работы синтаксического анализатора.
44. Дерево разбора. Преобразование дерева разбора в дерево операций.
45. Автоматизация построения синтаксических анализаторов.

Вопросы к экзамену:

1. Система прерываний 32-разрядных микропроцессоров i80x86. Работа системы прерываний в реальном режиме работы процессора. Работа системы прерываний в защищенном режиме работы процессора.
2. Режимы управления вводом/выводом. Закрепление устройств, общие устройства ввода/вывода. Основные системные таблицы ввода/вывода. Синхронный и асинхронный ввод/вывод. Кэширование операций ввода/вывода при работе с накопителями на магнитных дисках.
3. Функции файловой системы ОС и иерархия данных. Структура магнитного диска (разбиение дисков на разделы).
4. Файловая система FAT. Таблица размещения файлов. Структура загрузочной записи DOS.
5. Файловые системы VFAT и FAT32.

6. Файловая система HPFS.
7. Файловая система NTFS (NewTechnologyFileSystem). Основные возможности файловой системы NTFS. Структура тома с файловой системой NTFS. Возможности файловой системы NTFS по ограничению доступа к файлам и каталогам.
8. Основные принципы построения операционных систем. Принцип модульности. Принцип функциональной избирательности. Принцип генерируемости ОС. Принцип функциональной избыточности. Принцип виртуализации. Принцип независимости программ от внешних устройств. Принцип совместимости. Принцип открытой и наращиваемой ОС. Принцип мобильности (переносимости). Принцип обеспечения безопасности вычислений.
9. Микроядерные операционные системы. Монолитные операционные системы.
10. Требования, предъявляемые к ОС реального времени.
11. Мультипрограммность и многозадачность.
12. Приоритеты задач (поток). Наследование приоритетов.
13. Синхронизация процессов и задач. Предсказуемость.
14. Принципы построения интерфейсов операционных систем. Интерфейс прикладного программирования.
15. Реализация функций API на уровне ОС. Реализация функций API на уровне системы программирования. Реализация функций API с помощью внешних библиотек.
16. Платформенно-независимый интерфейс POSIX.
17. Семейство операционных систем UNIX. Общая характеристика семейства операционных систем UNIX, особенности архитектуры семейства ОС UNIX. Основные понятия системы UNIX. Функционирование системы UNIX. Файловая система. Межпроцессные коммуникации в UNIX.36. Операционная система Linux.
18. Семейство операционных систем OS/2 Warp компании IBM. Особенности архитектуры и основные возможности OS/2 Warp. Особенности интерфейса OS/2 Warp. Серверная операционная система OS/2 Warp 4.5.
19. Сетевая ОС реального времени QNX. Архитектура системы QNX. Основные механизмы QNX для организации распределенных вычислений.
20. Современные системы программирования. Понятие и структура системы программирования. История возникновения систем программирования. Структура современной системы программирования.
21. Принципы функционирования систем программирования. Функции текстовых редакторов в системах программирования. Компилятор как составная часть системы программирования. Компоновщик. Назначение и функции компоновщика.
22. Загрузчики и отладчики. Функции загрузчика. Библиотеки подпрограмм как составная часть систем программирования. Дополнительные возможности систем программирования. Лексический анализ «на лету». Система подсказок и справок.
23. Разработка программ в архитектуре «клиент-сервер». Разработка программ в трехуровневой архитектуре. Серверы приложений.
24. Системы программирования компании Embarcadero.
25. Системы программирования фирмы Microsoft.
26. Системы программирования под ОС Linux и UNIX.
27. Разработка программного обеспечения для сети Интернет.
28. Проектирование параллельных взаимодействующих вычислительных процессов
29. Независимые и взаимодействующие вычислительные процессы
30. Средства синхронизации и связи при проектировании взаимодействующих вычислительных процессов
31. Использование блокировки памяти при синхронизации параллельных процессов
32. Синхронизация процессов посредством операции «Проверка и установка»
33. Семафорные примитивы Дейкстры
34. Использование семафоров при проектировании взаимодействующих вычислительных процессов
35. Мониторы Хоара
36. Почтовые ящики. Конвейеры и очереди сообщений
37. Конвейеры (программные каналы)
38. Очереди сообщений
39. Примеры создания параллельных взаимодействующих вычислительных процессов
40. Пример создания многозадачного приложения с помощью системы программирования Borland Delphi
41. Пример создания комплекса параллельных взаимодействующих программ, выступающих как самостоятельные вычислительные процессы

2.4. Перечень примерных тем курсовых работ

1. Лексический анализатор программной строки пользователя
2. Компиляция и лексический анализ команды пользователя
3. Поиск, лексический анализ команды пользователя по модели транслятора языков
4. Хэш-функции компилятора при лексическом анализе
5. Статические и динамические данные по идентификаторам программы при лексическом анализе
6. Метод цепочек при организации динамической адресации при лексическом анализе
7. Поиск в двоичных деревьях при организации компилятором лексического анализа

8. Построение таблиц рехэширования в процессе лексического анализа
9. Обработка коллизий пользовательских идентификаторов при лексическом анализе
10. Поиск системы трансляторов при лексическом анализе команд пользователя
11. Оптимизация поиска и метод цепочек по модели языкового транслятора
12. Модель генерации коллизий при лексическом анализе
13. Функции хэш-адресации и рехэширования при лексическом анализе компиляторов
14. Организация таблиц статической и динамической адресации в процессе лексического анализа
15. Компиляция командных строк для систем с промышленными контролями
16. Ветвление и цепочки в основе адресного обеспечения компилируемых данных
17. Фрагментация списков для решения проблемы коллизий с системах трансляции
18. Разделение коллизий в поисковых системах лексических анализаторов
19. Сравнительный анализ операций поиска стадии трансляции
20. Построение и поиск в статической части адресных таблиц
21. Построение и поиск в динамической части адресных таблиц

3. Методические материалы, определяющие процедуру и критерии оценивания сформированных компетенций при проведении промежуточной аттестации

Критерии формирования оценок по результатам выполнения заданий

Критерии формирования оценок по ответам на вопросы, выполнению тестовых заданий

- оценка **«отлично»** выставляется обучающемуся, если количество правильных ответов на вопросы составляет 100 – 90 % от общего объёма заданных вопросов;
- оценка **«хорошо»** выставляется обучающемуся, если количество правильных ответов на вопросы – 89 – 76 % от общего объёма заданных вопросов;
- оценка **«удовлетворительно»** выставляется обучающемуся, если количество правильных ответов на тестовые вопросы – 75–60 % от общего объёма заданных вопросов;
- оценка **«неудовлетворительно»** выставляется обучающемуся, если количество правильных ответов – менее 60 % от общего объёма заданных вопросов.

Критерии формирования оценок по результатам выполнения заданий

«Отлично/зачтено» – ставится за работу, выполненную полностью без ошибок и недочетов.

«Хорошо/зачтено» – ставится за работу, выполненную полностью, но при наличии в ней не более одной негрубой ошибки и одного недочета, не более трех недочетов.

«Удовлетворительно/зачтено» – ставится за работу, если обучающийся правильно выполнил не менее 2/3 всей работы или допустил не более одной грубой ошибки и двух недочетов, не более одной грубой и одной негрубой ошибки, не более трех негрубых ошибок, одной негрубой ошибки и двух недочетов.

«Неудовлетворительно/не зачтено» – ставится за работу, если число ошибок и недочетов превысило норму для оценки «удовлетворительно» или правильно выполнено менее 2/3 всей работы.

Виды ошибок:

- *грубые ошибки: незнание основных понятий, правил, норм; незнание приемов решения задач; ошибки, показывающие неправильное понимание условия предложенного задания.*

- *негрубые ошибки: неточности формулировок, определений; нерациональный выбор хода решения.*

- *недочеты: нерациональные приемы выполнения задания; отдельные погрешности в формулировке выводов; небрежное выполнение задания.*

Критерии формирования оценок по зачету

«Зачтено» - обучающийся демонстрирует знание основных разделов программы изучаемого курса: его базовых понятий и фундаментальных проблем; приобрел необходимые умения и навыки, освоил вопросы практического применения полученных знаний, не допустил фактических ошибок при ответе, достаточно последовательно и логично излагает теоретический материал, допуская лишь незначительные нарушения последовательности изложения и некоторые неточности.

«Не зачтено» - выставляется в том случае, когда обучающийся демонстрирует фрагментарные знания основных разделов программы изучаемого курса: его базовых понятий и фундаментальных проблем. У экзаменуемого слабо выражена способность к самостоятельному аналитическому мышлению, имеются затруднения в изложении материала, отсутствуют необходимые умения и навыки, допущены грубые ошибки и незнание терминологии, отказ отвечать на дополнительные вопросы, знание которых необходимо для получения положительной оценки.

Критерии формирования оценок по написанию и защите курсовой работы

«Отлично» (5 баллов) – получают обучающиеся, оформившие курсовую работу (курсовой проект) в соответствии с предъявляемыми требованиями, в которой отражены все необходимые результаты проведенного анализа, сделаны обобщающие выводы и предложены рекомендации в соответствии с тематикой курсовой работы, а также грамотно и исчерпывающе ответившие на все встречные вопросы преподавателя.

«Хорошо» (4 балла) – получают обучающиеся, оформившие курсовую работу (курсовой проект) в соответствии с предъявляемыми требованиями, в которой отражены все необходимые результаты проведенного анализа, сделаны обобщающие выводы и предложены рекомендации в соответствии с тематикой курсовой работы. При этом при ответах на вопросы преподавателя обучающийся допустил не более двух ошибок.

«Удовлетворительно» (3 балла) – получают обучающиеся, оформившие курсовую работу (курсовой проект) в соответствии с предъявляемыми требованиями. При этом при ответах на вопросы преподавателя обучающийся допустил более трёх ошибок.

«Неудовлетворительно» (0 баллов) – ставится за курсовую работу, если число ошибок и недочетов превысило удовлетворительный уровень компетенции

Критерии формирования оценок по экзамену

«Отлично» (5 баллов) – обучающийся демонстрирует знание всех разделов изучаемой дисциплины: содержание базовых понятий и фундаментальных проблем; умение излагать программный материал с демонстрацией конкретных примеров. Свободное владение материалом должно характеризоваться логической ясностью и четким видением путей применения полученных знаний в практической деятельности, умением связать материал с другими отраслями знания.

«Хорошо» (4 балла) – обучающийся демонстрирует знания всех разделов изучаемой дисциплины: содержание базовых понятий и фундаментальных проблем; приобрел необходимые умения и навыки, освоил вопросы практического применения полученных знаний, не допустил фактических ошибок при ответе, достаточно последовательно и логично излагает теоретический материал, допуская лишь незначительные нарушения последовательности изложения и некоторые неточности. Таким образом данная оценка выставляется за правильный, но недостаточно полный ответ.

«Удовлетворительно» (3 балла) – обучающийся демонстрирует знание основных разделов программы изучаемого курса: его базовых понятий и фундаментальных проблем. Однако знание основных проблем курса не подкрепляется конкретными практическими примерами, не полностью раскрыта сущность вопросов, ответ недостаточно логичен и не всегда последователен, допущены ошибки и неточности.

«Неудовлетворительно» (0 баллов) – выставляется в том случае, когда обучающийся демонстрирует фрагментарные знания основных разделов программы изучаемого курса: его базовых понятий и фундаментальных проблем. У экзаменуемого слабо выражена способность к самостоятельному аналитическому мышлению, имеются затруднения в изложении материала, отсутствуют необходимые умения и навыки, допущены грубые ошибки и незнание терминологии, отказ отвечать на дополнительные вопросы, знание которых необходимо для получения положительной оценки.