

Документ подписан простой электронной подписью

Информация о владельце:

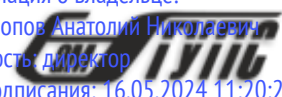
ФИО: Попов Анатолий Николаевич

Должность: директор

Дата подписания: 16.05.2024 11:20:28

Уникальный программный ключ:

1e0c38dccc0aee71c2e1c6c09d1d5875tc7497bc8



МИНИСТЕРСТВО ТРАНСПОРТА РОССИЙСКОЙ ФЕДЕРАЦИИ  
ФЕДЕРАЛЬНОЕ АГЕНТСТВО ЖЕЛЕЗНОДОРОЖНОГО ТРАНСПОРТА  
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ  
САМАРСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ПУТЕЙ СООБЩЕНИЯ

Приложение 2  
к рабочей программе дисциплины

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

### **Системное программное обеспечение**

*(наименование дисциплины(модуля))*

Направление подготовки / специальность

**09.03.03 Прикладная информатика**

*(код и наименование)*

Направленность (профиль)/специализация

**Прикладная информатика на железнодорожном транспорте**

*(наименование)*

## Содержание

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

### Перечень компетенций, формируемых в процессе освоения дисциплины

Код и наименование компетенции
ПК-2 Способен проводить работы по внедрению информационных систем
ПК-2.1 Формирует требования к информационной системе и разрабатывает её концепцию
ПК-4 Способен проводить тестирование компонентов программного обеспечения ИС
ПК-4.1 Проводит модульное тестирование программного обеспечения ИС, интеграционное тестирование
ПК-4.2 Проводит автоматизированное тестирование программного обеспечения с использованием современных библиотек и утилит, фреймворков

### Результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения образовательной программы

Код и наименование индикатора достижения компетенции	Результаты обучения по дисциплине	Оценочные материалы
ПК-2.1 Формирует требования к информационной системе и разрабатывает её концепцию	ПК-2.1.1 Знает: возможности существующей программно-технической архитектуры, современных средств разработки программных продуктов; методологии разработки программного обеспечения и технологии программирования, проектирования и использования баз данных; языки формализации функциональных спецификаций, методы и приемы формализации задач; типовые решения, библиотеки программных модулей, шаблоны, классы объектов, используемые при разработке программного обеспечения;	Задания 1-20
	ПК-2.1.2 Умеет: выбирать средства реализации требований к программному обеспечению; вырабатывать варианты реализации программного обеспечения; использовать существующие типовые решения и шаблоны проектирования программного обеспечения;	Задания 31-40
	ПК-2.1.3 Владеет: способами проектирование структур данных, баз данных, программных интерфейсов	Задания 41-43
ПК-4.1 Проводит модульное тестирование программного обеспечения ИС, интеграционное тестирование	ПК-4.1.1 Обучающийся знает: основные модели системных процессов	Задания 21-30
	ПК-4.1.2 Обучающийся умеет: разрабатывать программы на уровне моделей системных процессов	Задания 44-46
	ПК-4.1.3 Обучающийся владеет: шаблонами проектирования системного программного обеспечения	Задания 47-50
ПК-4.2 Проводит автоматизированное тестирование программного	ПК-4.2.1 Обучающийся знает: Методы функционального проектирования. Технические требования к пользовательскому интерфейсу. Архитектуру целевой аппаратной платформы, для которой разрабатывается	Тест 1-10

обеспечения с использованием современных библиотек и утилит, фреймворков	программное обеспечение.	
	ПК-4.2.2 Обучающийся умеет: планировать проектные работы; выбирать методики разработки требований к системе и шаблоны документов требований к системе; формулировать цели, исходя из анализа проблем, потребностей и возможностей. Разрабатывать графический дизайн интерфейсов.	Задания 51-53
	ПК-4.2.3 Обучающийся владеет: Навыками создания графических документов в программах растровых и векторных изображений. Разработки блок-схемы разрабатываемых систем. Реинжиниринга разработанного программного обеспечения для решения технических задач	Задания 54-56

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

- 1) ответ на билет, состоящий из теоретических вопросов и практических заданий;
- 2) выполнение заданий в ЭИОС СамГУПС.

**2. Типовые<sup>1</sup> контрольные задания или иные материалы для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих уровень сформированности компетенций**

**2.1 Типовые вопросы (тестовые задания) для оценки знаниевого образовательного результата**

Проверяемый образовательный результат

Код и наименование индикатора достижения компетенции	Образовательный результат
ПК-2.1.1	Знает: возможности существующей программно-технической архитектуры, современных средств разработки программных продуктов; методологии разработки программного обеспечения и технологии программирования, проектирования и использования баз данных; языки формализации функциональных спецификаций, методы и приемы формализации задач; типовые решения, библиотеки программных модулей, шаблоны, классы объектов, используемые при разработке программного обеспечения;
<p>Вопрос к заданиям 1 -20. Заполните пропуски в следующих высказываниях:</p> <ol style="list-style-type: none"> <li>1. Принцип Неймана - это принцип _____ программы.</li> <li>2. Процессор - это функциональный блок ЭВМ, предназначенный для обработки информации на основе принципа _____ управления.</li> <li>3. С точки зрения процессора нет принципиальной разницы между _____ и _____ .</li> <li>4. Регистры CS, DS, SS, ES, GS называются _____ регистрами.</li> <li>5. Регистр EAX / AX / AH / AL также носит название _____ .</li> <li>6. Регистр EBX / BX / BH / BL также носит название _____ .</li> <li>7. Регистр ECX / CX / CH / CL также носит название _____ .</li> <li>8. Регистр EDX / DX / DH / DL также носит название _____ .</li> <li>9. Сегментный регистр _____ служит для доступа к сегменту данных.</li> <li>10. Сегментный регистр _____ служит для доступа к сегменту стека.</li> <li>11. Сегментный регистр _____ служит для доступа к сегменту кода.</li> <li>12. Флаг CF называется флагом _____ .</li> <li>13. Флаг ZF называется флагом _____ .</li> <li>14. Флаг SF называется флагом _____ .</li> <li>15. Сегмент может начинаться только с начала _____ .</li> <li>16. Область памяти размером 16 байт с адресом начала кратным 16 называется _____ .</li> <li>17. Максимальный размер сегмента составляет _____ .</li> <li>18. С точки зрения размерности микропроцессор поддерживает следующие типы: байт, _____, _____, _____ .</li> <li>19. _____ представляет некоторую последовательность битов, в которой каждый бит является независимым и может рассматриваться как отдельная переменная.</li> <li>20. _____ представляет собой некоторый непрерывный набор байтов, слов,...максимальной длиной до 4 Гбайт</li> </ol>	
ПК-4.1.1	Обучающийся знает: основные модели системных процессов
<p>Вопрос к заданиям 21 -30. Дан фрагмент программы на ассемблере:</p> <p style="text-align: center;">push BX</p>	

<sup>1</sup>Приводятся 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.

```

push CX
push DX
push AX
mov AX,3508h
add BX,AX
pop AX
pop DX
mov DX,CX
shlBX,1
notAX

```

В каждом задании даны начальные значения регистров AX, BX, CX, DX. Определить значения этих регистров после выполнения данного фрагмента.

21. AX=0000h, BX=0CE5h, CX=0007h, DX=0023h
22. AX=0427h, BX=5ACEh, CX=0000h, DX=000Fh
23. AX=1010h, BX=0000h, CX=0DACh, DX=9990h
24. AX=0DEDh, BX=0444h, CX=1649h, DX=0000h
25. AX=0754h, BX=0431h, CX=0001h, DX=43Ach
26. AX=5325h, BX=0A11h, CX=9999h, DX=0101h
27. AX=0002h, BX=09ABh, CX=3333h, DX=0407h
28. AX=0AAAh, BX=3DEFh, CX=6109h, DX=0C0Ch
29. AX=35ADh, BX=1000h, CX=2100h, DX=0C03h
30. AX=7777h, BX=6666h, CX=5555h, DX=4444h

ПК-4.2.1	Обучающийся знает: Методы функционального проектирования. Технические требования к пользовательскому интерфейсу. Архитектуру целевой аппаратной платформы, для которой разрабатывается программное обеспечение.
<p>1 Легкость применения программного обеспечения это:</p> <ol style="list-style-type: none"> <li>а) характеристики ПО, позволяющие минимизировать усилия пользователя по подготовке исходных данных, применению ПО;</li> <li>б) отношение уровня услуг, предоставляемых ПО пользователю при заданных условиях, к объему используемых ресурсов;</li> <li>в) характеристики ПО, позволяющие минимизировать усилия по внесению изменений для устранения в нем ошибок и по его модификации.</li> </ol> <p>2 Мобильность программного обеспечения это:</p> <ol style="list-style-type: none"> <li>а) способность ПО выполнять набор функций, которые удовлетворяют потребности пользователей;</li> <li>б) способность ПС безотказно выполнять определенные функции при заданных условиях в течение заданного периода времени;</li> <li>в) способность ПО быть перенесенным из одной среды (аппаратного / программного) в другое.</li> </ol> <p>3 Что из приведенного является критериями оценки удобства интерфейсов?</p> <ol style="list-style-type: none"> <li>а) скорость обучения;</li> <li>б) адаптация к стилю работы пользователя;</li> <li>в) все ответы правильные.</li> </ol> <p>4 Артефакт — это</p> <ol style="list-style-type: none"> <li>а) любой продукт деятельности специалистов по разработке программного обеспечения;</li> <li>б) результат ошибок разработчика во входных или проектных спецификациях;</li> <li>в) графическое представление элементов моделирования системы.</li> </ol> <p>5 Укажите правильную последовательность этапов при каскадной модели жизненного цикла:</p> <ol style="list-style-type: none"> <li>а) Определение требований -&gt; Тестирование -&gt; Реализация;</li> <li>б) Проектирование -&gt; Реализация -&gt; Тестирование;</li> <li>в) Проектирование -&gt; Определение требований -&gt; Реализация.</li> </ol>	

6 Устойчивость программного обеспечения — это:

- а) свойство, характеризующее способность ПС завершать автоматически корректное функционирование ПК, несмотря на неправильные (ошибочные) входные данные;
- б) свойство, способная противостоять преднамеренным или непреднамеренным деструктивным действиям пользователя;
- в) свойство, характеризующее способность ПС продолжать корректное функционирование, несмотря на неправильные (ошибочные) входные данные.

7 UML — это:

- а) язык программирования, имеющий синтаксис схож с C ++;
- б) унифицированный язык визуального моделирования, использует нотацию диаграмм;
- в) набор стандартов и спецификаций качества программного обеспечения.

8 При конструировании программного обеспечения на этапе разработки или выбора алгоритма решения реализуется следующее:

- а) архитектурное обработки программы;
- б) выбор языка программирования;
- в) совершенствование программы.

9 Проектирование ПО в основном рассматривается как

- а) архитектурное проектирование;
- б) коммуникационные методы;
- в) детальные методы.

10 На этапе тестирования пользователь выполняет следующее:

- а) синтаксическое отладки;
- б) выбор тестов и метода тестирования;
- в) определение формы выдачи результатов.

## 2.2 Типовые задания для оценки навыкового образовательного результата

Проверяемый образовательный результат

Код и наименование индикатора достижения компетенции	Образовательный результат
ПК-2.1.2	Умеет: выбирать средства реализации требований к программному обеспечению; вырабатывать варианты реализации программного обеспечения; использовать существующие типовые решения и шаблоны проектирования программного обеспечения;

Вопрос к заданиям 31 - 40. Дан фрагмент программы на ассемблере:

```
push  AX
shr   BX,1
add   CX,BX
not   CX
kj1:  add  BX,3
loop  kj1
or    AX,DX
```

В каждом фрагменте даны начальные значения регистров AX, BX, CX, DX. Определить значения этих регистров после выполнения данного фрагмента.

- 31. AX=DACCh, BX=4A1Ch, CX=3542h, DX=7722h
- 32. AX=FFFDh, BX=0003h, CX=55DDh, DX=0985h
- 33. AX=F65bh, BX=1347h, CX=1111h, DX=0886h
- 34. AX=9A13h, BX=CBD5h, CX=0000h, DX=0AECCh

	<p>35. AX=9999h, BX=CCB8h, CX=D0D0h, DX=0011h  36. AX=B000h, BX=9FEBh, CX=3100h, DX=0F0Fh  37. AX=D340h, BX=58B6h, CX=0F00h, DX=CCCCh  38. AX=CCCCh, BX=215Bh, CX=5555h, DX=4321h  39. AX=0E00h, BX=3828h, CX=3251h, DX=000Fh  40. AX=FEDAh, BX=0231h, CX=9876h, DX=4444h</p>
ПК-2.1.3	Владеет: способами проектирование структур данных, баз данных, программных интерфейсов
<p>Задание 41. Провести моделирование объекта автоматизации.  Познакомиться с методологией ARIS;  Задание 42. Построить диаграмму цепочек добавленного качества (VAD диаграммы)  Задание 43. Построить eEPC модели – цепочки процессов, управляемой событиями.</p>	
ПК-4.1.2	Обучающийся умеет: разрабатывать программы на уровне моделей системных процессов
<p>Задание 44. Провести анализ проблемы. На этапе анализа проблемы провести анализ предметной области, для которой разрабатывается ПО.  1) определить границы, или контура, системы;  2) описать объектов автоматизации и/или формализации знаний об этих объектах;  3) выявить или определить потребностей заказчика ПО.  Задание 45. Для выявления потребностей заказчика и описания объектов автоматизации необходимо провести анкетирование заказчиков ПО.  Задание 46. Постановка задачи.  На этапе постановки задачи разработать:  1) перечень заинтересованных лиц;  2) список потребностей заинтересованных лиц в разрабатываемом ПО.</p>	
ПК-4.1.3	Обучающийся владеет: шаблонами проектирования системного программного обеспечения
<p>Задание 47. Разработать модели вариантов использования и их спецификаций.  Задание 48. Оформить технического задания в соответствии с ГОСТ 34.602–89.  Задание 49. Реализовать архитектуру ПО на базе объектно-реляционного отображения.  Задание 50. Разработать простое MDA-приложения.</p>	
ПК-4.2.2	Обучающийся умеет: планировать проектные работы; выбирать методики разработки требований к системе и шаблоны документов требований к системе; формулировать цели, исходя из анализа проблем, потребностей и возможностей. Разрабатывать графический дизайн интерфейсов.
<p>Задания:  51 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  52 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.  53 Тема «Одномерный поток»  Задание: конструктор потока, запуск потока, индикация этапов задачи до потока во время его работы и после завершения потока.</p>	
ПК-4.2.3	Обучающийся владеет: Навыками создания графических документов в программах растровых и векторных изображений. Разработки блок-схемы разрабатываемых систем. Реинжиниринга разработанного программного обеспечения для решения технических задач
<p>Задания:  54 Вывод потоком на консоль собственного и системного идентификатора; вывод имени потока и основных характеристик его системных ресурсов (адрес, приоритет и т.п.)  Содержание задания:</p>	



- подготовить конструкторы асинхронных потоков
- передать не менее 2-х параметров в процедуры потоков посредством объекта-структуры
- в составе параметров определить имя или идентификатор потока
- считать системный идентификатор потока, приоритет потока. Вывести информационное сообщение на консоль.

55 Организация потоков в массив: регистрация потоков в динамической коллекции с распознаванием идентификатора или имени потока при прохождении по коллекции независимым итератором

56 Тема «Параллельные асинхронные потоки»

Задание: конструктор фиксированного числа потоков, запуск потоков, идентификация асинхронной работы потоков; передача структуры с параметрами потока по нетипированному адресу.

### **2.3. Перечень вопросов для подготовки обучающихся к промежуточной аттестации** **Вопросы к зачету:**

1. Определение, структура программного обеспечения.
2. Определение, функции операционной системы (ОС).
3. Определение, основные принципы построения ОС.
4. Понятие вычислительного процесса.
5. Понятие ресурса.
6. Понятие активного процесса. Динамика состояний процесса.
7. Понятие потока, мультипрограммирования.
8. Идентификация процесса.
9. Взаимодействие потоков.
10. Классификация процессов.
11. Классификация ресурсов.
12. Понятие критических секций, основные требования к ним.
13. Понятие тупика, условия его возникновения.
14. Методы борьбы с тупиками. Описание каждого метода.
15. Виды межпроцессных коммуникаций. Очереди сообщений. Сигналы.
16. Виды межпроцессных коммуникаций. Конвейер. Сокеты.
17. Понятие системных часов, таймера.
18. Планирование выполнения процессов в системах реального времени.
19. Отображение пространства имен на физическую память компьютера.
20. Сегментный способ организации виртуальной памяти.
21. Страничный способ организации виртуальной памяти.
22. Сегментно-страничный способ организации виртуальной памяти.
23. Управление памятью вычислительной системы.
24. Особенности файловой системы FAT.
25. Особенности файловой системы NTFS.
26. Понятие ввода/вывода. Основные задачи супервизора ввода/вывода.
27. Режимы ввода/вывода, их характеристика.
28. Процесс управления вводом/выводом.
29. Понятие микроядерной операционной системы.
30. Понятие монолитной операционной системы.
31. Классификация операционных систем.
32. Особенности сетевых и распределенных операционных систем.
33. Понятие прерывания. Механизм обработки прерываний.
34. Синхронные и асинхронные прерывания.
35. Дисциплины диспетчеризации.
36. Понятие утилиты. Виды утилит.
37. Понятие компилятора, интерпретатора, отладчика, компоновщика
38. Виды систем защиты программного обеспечения.
39. Показатели применимости и критерии оценки систем защиты программного обеспечения.

### **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 сформированности компетенций при проведении промежуточной аттестации

#### 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 с оценкой

**«Отлично/зачтено»** – студент приобрел необходимые умения и навыки, продемонстрировал навык практического применения полученных знаний, не допустил логических и фактических ошибок

**«Хорошо/зачтено»** – студент приобрел необходимые умения и навыки, продемонстрировал навык практического применения полученных знаний; допустил незначительные ошибки и неточности.

**«Удовлетворительно/зачтено»** – студент допустил существенные ошибки.

**«Неудовлетворительно/не зачтено»** – студент демонстрирует фрагментарные знания изучаемого курса; отсутствуют необходимые умения и навыки, допущены грубые ошибки.

## 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.

Экспертный лист  
оценочных материалов для проведения промежуточной аттестации по  
дисциплине «Системное программное обеспечение»

Направление подготовки / специальность

**09.03.03 Прикладная информатика**  
(код и наименование)

Направленность (профиль)/специализация

**Прикладная информатика на железнодорожном транспорте**  
(наименование)

**Бакалавр**  
квалификация выпускника

1. Формальное оценивание			
Показатели	Присутствуют	Отсутствуют	
Наличие обязательных структурных элементов:	+		
– титульный лист	+		
– пояснительная записка	+		
– типовые оценочные материалы	+		
– методические материалы, определяющие процедуру и критерии оценивания	+		
Содержательное оценивание			
Показатели	Соответствует	Соответствует частично	Не соответствует
Соответствие требованиям ФГОС ВО к результатам освоения программы	+		
Соответствие требованиям ОПОП ВО к результатам освоения программы	+		
Ориентация на требования к трудовым функциям ПС (при наличии утвержденного ПС)	+		
Соответствует формируемым компетенциям, индикаторам достижения компетенций	+		

Заключение: ФОС рекомендуется/ не рекомендуется к внедрению; обеспечивает/ не обеспечивает объективность и достоверность результатов при проведении оценивания результатов обучения; критерии и показатели оценивания компетенций, шкалы оценивания обеспечивают/ не обеспечивают проведение всесторонней оценки результатов обучения.

Эксперт, должность, ученая степень, ученое звание \_\_\_\_\_ /

(подпись)