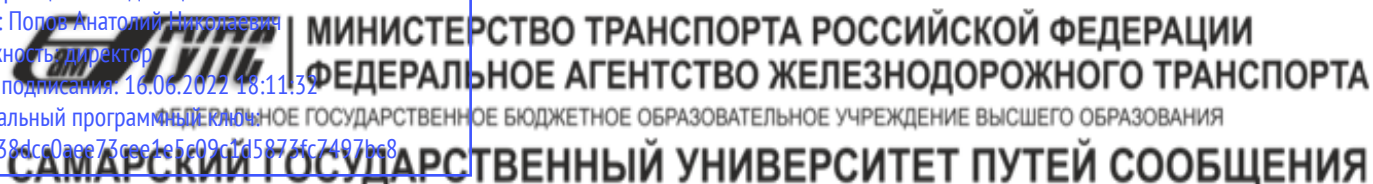


Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Попов Анатолий Николаевич
Должность: директор
Дата подписания: 16.06.2022 18:11:32
Уникальный программный ключ:
1e0c38dca0ae73cee1e5e09c1d5873fc7497ba8



Приложение 2
к рабочей программе дисциплины

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ДЛЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ПО ДИСЦИПЛИНЕ (МОДУЛЮ)

Основы и методологии программирования *(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	Код индикатора достижения компетенции
ОПК-8 Способен разрабатывать алгоритмы и программы, пригодные для практического применения	Знать: – понятие алгоритма и классификацию алгоритмических языков и систем программирования; – язык программирования высокого уровня - Си;
	Уметь: – применять на практике современные технологии разработки алгоритмов и программ, языки программирования, методы тестирования, отладки и решения задач на ЭВМ; – Разрабатывать программы в соответствии с выбранной моделью жизненного цикла;
	Владеть: – навыками алгоритмизации и программной реализации на языке высокого уровня решений практических задач; – навыками разработки программ в водопадной модели жизненного цикла

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

Код и наименование индикатора достижения компетенции	Результаты обучения по дисциплине	Оценочные материалы
ОПК-8 Способен разрабатывать алгоритмы и программы, пригодные для практического применения деятельности	Знает: – понятие алгоритма и классификацию алгоритмических языков и систем программирования; – язык программирования высокого уровня - Си;	
	Умеет: – применять на практике современные технологии разработки алгоритмов и программ, языки программирования, методы тестирования, отладки и решения задач на ЭВМ; – Разрабатывать программы в соответствии с выбранной моделью жизненного цикла;	
	Владеет: – навыками алгоритмизации и программной реализации на языке высокого уровня решений практических задач; – навыками разработки программ в водопадной модели жизненного цикла	

Промежуточная аттестация (зачет) 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 сформированности компетенций

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

Проверяемый образовательный результат:

Код и наименование индикатора достижения компетенции	Образовательный результат
ОПК-8 Способен разрабатывать алгоритмы и программы, пригодные для практического применения деятельности	Обучающийся знает: понятие алгоритма и классификацию алгоритмических языков и систем программирования; язык программирования высокого уровня - Си;
ОПК-8 Способен разрабатывать алгоритмы и программы, пригодные для практического применения деятельности	Обучающийся умеет: применять на практике современные технологии разработки алгоритмов и программ, языки программирования, методы тестирования, отладки и решения задач на ЭВМ; Разрабатывать программы в соответствии с выбранной моделью жизненного цикла;
ОПК-8 Способен разрабатывать алгоритмы и программы, пригодные для практического применения деятельности	Обучающийся владеет: навыками алгоритмизации и программной реализации на языке высокого уровня решений практических задач; навыками разработки программ в водопадной модели жизненного цикла

2.2. Примеры тестовых заданий

1. *Инструментальные средства быстрого прототипирования ИС - ...*
 - ☐ языки программирования высокого уровня
 - ☐ приложения в развитых СУБД – класс DEVELOPER
 - ☐ интегрированные инструменты быстрой разработки
 - ☐ приложений – класс BUILDER
2. *Сущность прототипной (RAD) технологии - ...*
 - ☐ создание на ранней стадии реализации действующей интерактивной модели системы, так называемой системы-прототипа, позволяющей наглядно продемонстрировать пользователю будущую систему, уточнить его требования, оперативно модифицировать интерфейсные элементы
 - ☐ формализация процесса проектирования ИС
 - ☐ активное вовлечение конечных пользователей в процесс разработки предназначенных для них интерактивных систем
3. *К технологиям традиционного прототипирования ИС относятся.....*
 - ☐ демонстрация работы прототипа
 - ☐ разработка системы прототипа
 - ☐ доработка системы прототипа
 - ☐ разработка новых спецификаций требований
 - ☐ разработка постановки задачи
 - ☐ документирование готового приложения
4. *Инструментальными средствами быстрого прототипирования ИС являются ...*
 - ☐ приложения в развитых СУБД - класс DEVELOPER
 - ☐ интегрированные инструменты быстрой разработки приложений - класс BUILDER
 - ☐ языки программирования высокого уровня
5. *Технология итерационного прототипирования ИС - ...*
 - ☐ разработка новой постановки задачи
 - ☐ демонстрация работы прототипа
 - ☐ разработка приложения

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.

- ☐ доработка системы прототипа
 - ☐ документирование готового приложения
 - ☐ разработка системы прототипа
6. К основным приемам быстрой разработки прототипа ИС относятся ...
- ☐ разработка приложения итерациями
 - ☐ использование большого количества технических средств
 - ☐ использование автоматических генераторов (мастеров)
 - ☐ высокая параллельность работ
 - ☐ повторное использование частей проекта
 - ☐ низкая производительность проектировщиков
7. Диаграммами, выступающими в качестве инструментальных средств функционально-ориентированного анализа и проектирования являются...
- ☐ диаграммы деятельности (Activity diagram)
 - ☐ SSD (System Structure Diagram) - диаграмма структуры программного приложения
 - ☐ диаграммы взаимодействия объектов (Interaction diagram)
 - ☐ диаграммы состояний (Statechart diagram)
 - ☐ DFD (Data Flow Diagram) - диаграмма потоков данных
8. Признаки классификации CASE-средств по режиму коллективной разработки проекта - ...
- ☐ не поддерживающие коллективную разработку
 - ☐ по виду загруженности разработчиков
 - ☐ работающие под управлением WINDOWS
 - ☐ ориентированные на режим объединения подпроектов
 - ☐ ориентированные на режим реального времени разработки проекта
9. Признаки классификации CASE-средств по типу операционной системы (ОС) работающие под управлением -...
- ☐ WINDOWS 3.11 и выше
 - ☐ Microsoft Office
 - ☐ OS/2 и др.
 - ☐ UNIX
10. Диаграммы, выступающие в качестве инструментальных средств объектно-ориентированного анализа и проектирования - ...
- ☐ диаграммы компонентов (Component diagram)
 - ☐ диаграммы деятельности (Activity diagram)
 - ☐ диаграммы пакетов (Package diagram)
 - ☐ диаграммы размещения (Deployment diagram)
 - ☐ DFD (Data Flow Diagram) - диаграмма потоков данных
 - ☐ диаграммы взаимодействия объектов (Interaction diagram)
11. Преимуществом CASE-технологии является ...
- ☐ возможность коллективной разработки ЭИС в режиме реального времени
 - ☐ возможность повторного использования компонентов разработки
 - ☐ улучшение климата в коллективе разработчиков
 - ☐ поддержание адаптивности и сопровождения ИС
 - ☐ снижение усталости разработчика
12. Признаки классификации CASE-средств по поддерживаемым графическим нотациям построения диаграмм - ...
- ☐ функционально (структурно)-ориентированные
 - ☐ с отдельными нотациями и наиболее распространенными нотациями
 - ☐ с фиксированной нотацией
 - ☐ объектно-ориентированные
13. Признаки классификации CASE-средств по степени интегрированности -...
- ☐ workbench (полностью интегрированные средства, связанные общей базой проектных данных - репозиторием)
 - ☐ с отдельными нотациями
 - ☐ с наиболее распространенными нотациями
 - ☐ tools (отдельные локальные средства)
 - ☐ с фиксированной нотацией
 - ☐ toolkit (набор неинтегрированных средств, охватывающих большинство этапов разработки ИС)
14. К архитектуре CASE-средства относится ...
- ☐ администратор проекта
 - ☐ репозиторий (словарь данных)
 - ☐ верификатор диаграмм
 - ☐ графический редактор диаграмм

- ☐ дизайнер
- ☐ файл-сервер

15. Критерием выбора алгоритмического языка при проектировании программного обеспечения задачи является

...

- ☐ объем памяти, занимаемой разработанной программой
- ☐ синтаксическая и семантическая ясность языка
- ☐ совместимость с другими языками
- ☐ возможность управления файлами
- ☐ решение заказчика проекта
- ☐ время написания программы

16. Стадии жизненного цикла являющиеся основными ...

- ☐ обсуждение проекта ИС
- ☐ проектирование
- ☐ внедрение (тестирование, опытная эксплуатация)
- ☐ реализация (программирование)
- ☐ эксплуатация (сопровождение, модернизация)

17. Требования, предъявляемые к средствам проектирования ...

- ☐ простыми в освоении и применении
- ☐ помогать планировщику
- ☐ технически, программно и информационно совместимыми
- ☐ охватывать в совокупности все этапы жизненного цикла ПО
- ☐ экономически целесообразными

18. Техническое задание включает в себя ...

- ☐ инструкцию по использованию выходных документов
- ☐ обеспечивающую часть системы
- ☐ функциональную часть системы
- ☐ организация работ и исполнители
- ☐ инструкцию по организации хранения информации в архиве
- ☐ общие положения
- ☐ основание для разработки системы

19. Техническим заданием является...

- ☐ документ, утвержденный в установленном порядке, определяющий цели, требования и основные исходные данные, необходимые для разработки автоматизированной системы управления
- ☐ основание для разработки системы
- ☐ техническая документация, утвержденная в установленном порядке, содержащая общесистемные проектные решения, алгоритм решения задач

20. В системе программирования после обнаружения факта ошибки начинается процесс ее поиска и исправления, который называют ...

- ☐ отладкой
- ☐ компиляцией
- ☐ тестированием
- ☐ трассировка

21. Этапы разработки программной системы с использованием объектно-ориентированного подхода:

- ☐ постановка задачи
- ☐ анализ
- ☐ катализ
- ☐ синтез
- ☐ проектирование
- ☐ реализация
- ☐ модификация

22. Этапы разработки программной системы с использованием структурного подхода:

- ☐ постановка задачи
- ☐ анализ
- ☐ катализ
- ☐ синтез
- ☐ проектирование
- ☐ реализация
- ☐ модификация

2.3. Перечень вопросов для подготовки обучающихся к промежуточной аттестации

1. Жизненный цикл программного обеспечения. Содержание этапов классического ЖЦ. Качество ПО. Критерии качества.
2. Виды архитектур программного средства.
3. Основные области в программировании: методологии и технологии программирования; языки и системы программирования; операционные и архитектурные платформы. Краткая характеристика каждой группы.
4. Методология программирования. Атрибуты методологии. Ядра методологий. Основные методологии: императивного программирования, объектно-ориентированного программирования, функционального программирования, логического программирования, активного программирования (экстремального). Краткая характеристика.
5. Технология: система обозначений, процесс и инструментальные средства.
6. Технологии программирования. Основные понятия и определения.
7. Технологический подход. Классификация подходов.
8. Сложность и сложные системы. Источники сложности. Признаки работоспособной сложной системы. Методы борьбы со сложностью. Моделирование, абстракция, декомпозиция.
9. Функциональная и объектная декомпозиция.
10. Три стратегии конструирования программного обеспечения.
11. Водопадная модель разработки приложений. Недостатки и кризис последовательной модели разработки ПО.
12. Инкрементная модель разработки приложений.
13. Спиральная модель разработки приложений.
14. Тяжеловесные и облегченные процессы разработки приложений.
15. Характеристики XP-процесса.
16. Технологический процесс. Классификация технологических процессов.
17. История и эволюция технологий программирования. Проблемы и перспективы развития.
18. Основные подходы к организации процесса создания и использования программных средств.
19. Методы проектирования программ.
20. Структурное программирование.
21. Модульное программирование. Преимущества модульного программирования. Определение модуля. Свойства модуля. Модульная структура приложения.
22. Сущность объектного подхода к разработке программ.
23. Объектно-ориентированное программирование. Основное понятие ООП. Объекты и классы. Определения, назначение. Принцип инкапсуляции. Принцип наследования. Принцип полиморфизма.
24. Основы визуального программирования. Элементы процесса визуализации. Принципы визуального программирования. Элементы управления. Проектирование формы. Визуальные и объектные аспекты форм. Архитектура приложения. Многооконные интерфейсы.

25. Виды пользовательских интерфейсов и требования к ним.
26. Проектирование пользовательского интерфейса. Рекомендации по проектированию пользовательского интерфейса.
27. Принципы построения пользовательского интерфейса.
28. Три размерности согласованности пользовательского интерфейса.
29. Два вида стилей взаимодействия между пользователем и компьютером и способы для связи.
30. Принципы использования цвета при проектировании эргономичного интерфейса.
31. Тексты и диалоги. Принципы создания текстовых диалогов и отображений.
32. Средства управления графического интерфейса пользователя.
33. Принципы проектирования меню приложения.
34. Режим окон приложения. Три типа окон приложения.
35. Основной элемент интерфейса – экранные формы. Принципы проектирования форм.
36. Проектирование сообщений для пользователя по выполнению необходимых действий на пути решения задачи.
37. Тестирование программ. Методы тестирования.
38. Приемы отладки программных средств. Автономная отладка. Комплексная отладка.
39. Отладка программ. Причины ошибок Типы ошибок в программе: времени выполнения, синтаксические, логические. Защита от ошибок.
40. Диалоговые окна. Стандартные диалоги. Процедуры и функции, реализующие диалоги.
41. Стандарты, необходимые для создания программного обеспечения.
42. Основные программные и эксплуатационные документы (по ГОСТ).
43. Спецификация требований к программной системе. Требования к данным, функциональности, интерфейсу, надежности, безопасности, эффективности. Описание предметной области. Требования к окружению. Архитектура системы.
44. Назначение и содержание «Технического задания».
45. Определение жизненного цикла ПО.
46. Группы процессов, входящие в состав ЖЦ ПО.
47. Наиболее часто используемые процессы в реальных проектах.
48. Определение модели ЖЦ ПО. Используемые модели ЖЦ в настоящее время.
49. Характеристика модели «Водопад».
50. Характеристика спиральной модели.
51. Инструментальные среды разработки и сопровождения ПС. Термин «Case-средства», основные возможности Case-средств.
52. Классификация Case-средств.

53. Структура Case-средства.
54. Сущность прототипной (RAD) технологии. Основные возможности и преимущества быстрой разработки.
55. Классификация инструментальных средств быстрого прототипирования.
56. Основы COM-технологии: понятие интерфейса.
57. Сущность автоматизации OLE.
58. История и эволюция языков моделирования. Блок-схемы, диаграммы.
59. Язык моделирования UML. Возникновение и развитие стандарта. Основные концепции моделирования (абстрагирование, множественность представлений). Обзор стандартных элементов и пакетов. Виды диаграмм.
60. Язык моделирования UML. Диаграммы классов. Классы, объекты, отношения ассоциации.
61. Язык моделирования UML. Отношения зависимости, обобщения, реализации. Стереотипы, интерфейсы.
62. Язык моделирования UML. Диаграмма вариантов использования. Применение совместно со спецификацией функций.
63. Язык моделирования UML. Диаграмма развертывания. Статические и динамические диаграммы, как средства изображения структур и процессов.
64. Язык моделирования UML. Диаграмма состояний. Применение при изображении пользовательского интерфейса и протоколов.
65. Язык моделирования UML. Диаграмма деятельности. Аналогии и преимущества перед блок-схемами.
66. Язык моделирования UML. Диаграмма последовательности. Диаграмма кооперации.
67. Компоненты. Каркасы. Паттерны. Паттерны порождения, структуры и управления.
68. Метрики (количественные оценки) ПО. Оценки сложности ПО, качества ПО, эффективности процесса разработки.
69. Правовые аспекты распространения ПО. Copyright и лицензирование. Свободно распространяемое ПО. Аттестация ПС.

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

Критерии формирования оценок по результатам выполнения заданий

«Зачтено»:

- ставится за работу, выполненную полностью без ошибок и недочетов.
- ставится за работу, выполненную полностью, но при наличии в ней не более одной негрубой ошибки и одного недочета, не более трех недочетов.
- 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.

«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.

Экспертный лист

оценочных материалов для проведения промежуточной аттестации по дисциплине «Основы и методологии программирования»

по направлению подготовки/специальности

09.03.01 «Информатика и вычислительная техника»

шифр и наименование направления подготовки/специальности

профиль / специализация

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.