

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Попов Анатолий Николаевич
Должность: директор
Дата подписания: 31.05.2023 17:48:18
Уникальный программный ключ:
1e0c38dca0aee73cee1e5e09c1d5873fc7497ba8

Приложение 2
к рабочей программе дисциплины

ОЦЕНОЧНЫЕ МАТЕРИАЛЫ ДЛЯ ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ
ПО ДИСЦИПЛИНЕ (МОДУЛЮ)
Объектно-ориентированное программирование

(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
ОПК-7 Способен разрабатывать алгоритмы и программы, пригодные для практического применения;
ОПК-7.1 Разрабатывает алгоритмы и программы на языке программирования высокого уровня
ПК-1 Способен разрабатывать требования и проектировать программное обеспечение
ПК-1.2 Разрабатывает программный код и осуществляет отладку на языках программирования высокого уровня

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

Код и наименование индикатора достижения компетенции	Результаты обучения по дисциплине	Оценочные материалы
ОПК-7.1 Разрабатывает алгоритмы и программы на языке программирования высокого уровня	ПК-1.3.1 Обучающийся знает: преимущества и особенности программирования на языке высокого уровня, основные понятия, конструкции и структуры языка программирования для решения задач.	Вопросы тестирования №(1-5)
	ПК-1.3.2 Обучающийся умеет: работать с современными средствами программирования на языках высокого уровня.	Задания №(1-4)
	ПК-1.3.3 Обучающийся владеет: инструментальными средствами, методами и навыками разработки программного обеспечения с использованием языка программирование высокого уровня.	Задания №(5-7)
ПК-1.2 Разрабатывает программный код и осуществляет отладку на языках программирования высокого уровня	ПК-1.4.1. Обучающийся знает: методы и средства верификации работоспособности выпусков программной продукции	Вопросы тестирования №(6-10)
	ПК-1.4.2. Обучающийся умеет: производить настройки параметров программного продукта и осуществлять запуск процедуры	Задания №(8-9)

	сборки.	
	ПК-1.4.3 Обучающийся владеет: методами разработки и осуществления процедур верификации выпусков программной продукции.	Задания №(10-13)

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

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

Промежуточная аттестация (курсовая работа) 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-7.1.1	Обучающийся знает: преимущества и особенности программирования на языке высокого уровня, основные понятия, конструкции и структуры языка программирования для решения задач.
	<p>1. Java. класс в Java для работы с сокетами дейтаграммного типа + <i>DatagramSocket</i> - URL - InetAddress - BufferedReader - BufferedWriter</p> <p>2. Java. класс в Java для работы с URL + <i>URL</i> - DatagramSocket - InetAddress - BufferedReader - BufferedWriter</p> <p>3. Java. метод getInputStream() + <i>возвращает входной поток типа InputStream</i> - возвращает выходной поток типа OutputStream - возвращает длину полученной информации в байтах - возвращает полученную информацию в виде объекта типа Object - ничего не возвращает</p> <p>4. Java. метод getOutputStream() + <i>возвращает входной поток типа OutputStream</i> - возвращает выходной поток типа InputStream - возвращает длину полученной информации в байтах - возвращает полученную информацию в виде объекта типа Object - ничего не возвращает</p> <p>5. Какая технология в Java позволяет заменить CGI программы, находящиеся на сервере? + <i>Сервлеты</i> - Апплеты - Многопоточность - Синхронизация - В Java нет такой технологии</p>

Код и наименование индикатора достижения компетенции	Образовательный результат
ПК-1.2.1.	Обучающийся знает: методы и средства верификации работоспособности выпусков программной продукции

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.

6. На каком языке можно написать CGI программу?

+ **Все ответы верны**

- Python
- Pascal
- Perl, PHP
- C, C++

7. Java. Пакет в Java, в котором множество классов связанных для работы с сетью

+ **java.net**

- javax
- java.util
- javax.swing
- java.awt

8. Java. Объект какого класса нужно создать, чтобы узнать к какому типу относится переданный файл?

+ **URLConnection**

- Iterator
- StringTokenizer
- Vector
- ArrayList

9. Java. Класс представляющий сокет в Java

+ **Socket**

- SocketClass
- SocketURL
- SocketConnection

10. Java. К какому пакету относится класс Socket?

+ **java.io**

- javax.swing
- java.awt
- javax
- java.util

2.2 Типовые задания для оценки навыкового образовательного результата

Проверяемый образовательный результат

Код и наименование индикатора достижения компетенции	Образовательный результат
ОПК-7.1.2	Обучающийся умеет: работать с современными средствами программирования на языках высокого уровня.
Примерные задания	
1. Разработка классов с учетом обработки исключительных ситуаций	
2. Транзакции	
3. Использование классов и алгоритмов библиотеки STL.	
4. Параметризованные классы (шаблоны классов). Перегруженные функции и функции-шаблоны.	
ОПК-7.1.3	Обучающийся владеет: инструментальными средствами, методами и навыками разработки программного обеспечения с использованием языка программирования высокого уровня.
Примеры заданий	
Темы заданий:	
5. Система управления местами гостиницы	
Система заказа номеров в гостинице.	
Ведение БД: Номера, Услуги, Клиенты	
Функции: Поселение и выселение клиентов, бронирование мест, учёт оказанных услуг	

<p>Выходные документы: Счёт за проживание и услуги, Список проживавших на момент времени, Список номеров, Прейскурант услуг.</p> <p>6. Система отслеживания успеваемости студентов Система ведения результатов успеваемости студентов. Ведение БД: кафедры, студенты, предметы Функции: занесение данных по студентам, формирование справочных документов, Выходные документы: Ведомость успеваемости по группе студентов.</p> <p>7. Система учета пациентов больницы. Ведение справочников: Пациенты, Болезни, Палаты, Врачи, История болезни Функции: Ведение справочников, приём пациента, ведение истории болезни, выписка. Выходные документы: Список пациентов, Список врачей, Карточка больного.</p>	
ПК-1.2.2.	Обучающийся умеет: производить настройки параметров программного продукта и осуществлять запуск процедуры сборки.
<p>Примеры заданий Темы заданий:</p> <p>8. Система учета заказов клиентов ресторана Ведение справочников: Продукты, Блюда, Заказы Функции: Ведение справочников, хранение рецептов, расчёт себестоимости блюда, приём заказов Выходные документы: Меню, Счёт заказа, Отчёт по продуктам на складе, Заказы за период.</p> <p>9. Система управления складом аптеки Ведение справочников: Группы лекарств, Лекарства, Производители, Поставщики Функции: ведение справочников, учёт прихода и продаж лекарств Выходные документы: Отчёт по наличию лекарств на складе по группам, Отчёт по продажам по группам, Счёт-фактура</p>	
ПК-1.2.3	Обучающийся владеет: методами разработки и осуществления процедур верификации выпусков программной продукции.
<p>Примеры заданий</p> <p>10. Написать программное обеспечение с методом перегрузки операций</p> <p>11. Написать программное обеспечение с методом наследования.</p> <p>12. Написать программное обеспечение с методом композиции классов.</p> <p>13. Написать программное обеспечение с методом структур классов.</p>	

2.3. Перечень вопросов для подготовки обучающихся к промежуточной аттестации

Вопросы к экзамену

1. Общая характеристика и основные принципы Объектно-ориентированного программирования.
2. Определение класса в C++. Понятие объектов.
3. Функциональные компоненты класса. Типы функциональных компонент.
4. Конструкторы.
5. Деструкторы.
6. Перегрузка функциональных элементов и операций.
7. Константные объекты и функции. Ключевое слово this.
8. Наследование классов. Конструкторы и деструкторы при наследовании.
9. Виртуальные функции. Понятие полиморфизма и позднего связывания.
10. Неоднозначность при множественном наследовании. Виртуальный базовый класс.
11. Абстрактный класс и чистые виртуальные функции.
12. Дружественные функции и классы.
13. Классы потоков языка C++. Операции извлечения и помещения данных.
14. Способы форматирования при работе с классами потоков. Форматирующие функции.
15. Флаги форматирования.
16. Манипуляторы.

17. Ошибки потоков.
18. Файловый ввод-вывод с применением потоков. Открытие файла.
19. Ввод-вывод в файлы с использованием потоков.
20. Строко - ориентированный ввод-вывод.
21. Неформатированный ввод-вывод.
22. Пространство имен.
23. Обработка исключительных ситуаций. Понятие исключений. Общий механизм обработки исключений.
24. Синтаксис исключений. Перехват исключений.
25. Списки исключений функций. Исключения в конструкторах и деструкторах.
26. Стандартные исключения. Иерархии исключений.
27. Шаблоны функций. Использование шаблонов функций.
28. Специализированная функция шаблона. Перегрузка шаблонов.
29. Определение и использование шаблонов классов.
30. Использование в шаблонных классах аргументов по умолчанию. Специализация шаблонов классов. Достоинства и недостатки шаблонов.
31. Класс string. Конструкторы. Преобразование строк.
32. Класс string. Поиск подстрок. Сравнение частей строк.
33. Класс string. Получение количества элементов. Изменение размера стро
34. Библиотека шаблонов STL. Назначение STL и состав библиотеки.
35. Понятие контейнера и их типы. итератора. Поля контейнеров.
36. Итераторы. Их типы. Методы для просмотра контейнеров с помощью итераторов.
37. Последовательные контейнеры. Общие операции для работы с контейнерами.
38. Контейнерный класс vector. Его конструкторы. Операции присваивания и копирования.

Доступ к элементам вектора.

39. Контейнерный класс vector. Его методы capacity, reserve, resize. Методы для изменения объектов класса vector.
40. Векторы логических значений.
41. Двухсторонние очереди (deque). Конструкторы. Операции, которые реализованы в deque. Методы добавления и выборки.
42. Списки list. Понятие и реализация. Доступ к элементам. Занесение в начало и конец.
43. Методы изменения объектов списка. Сцепление списков. Удаление элементов.

Сортировка.

44. Адаптеры стеки и очереди.
45. Особенности языка C++.
46. Структура программы на языке C++.
47. Целочисленные и вещественные типы в C++.
48. Типы char и Boolean в C++.
49. Переменные, константы и операции в C++.
50. Управляющие конструкции в C++.
51. Массивы в C++.
52. Описание классов. Объекты в C++.
53. Статические элементы. Доступ к членам класса в C++. Слово this.
54. Внутренние классы. Особенности передачи параметров в C++.
55. Наследование в C++. Переопределение методов. Конструкторы и наследование.
56. Переопределение методов при наследовании. Абстрактные методы и классы в C++.
57. Динамическое управление методами в C++. Полиморфизм. Пакеты в C++.
58. Интерфейсы в C++.

Вопросы к зачету

1. Эволюция методологий программирования. Парадигмы программирования.
2. Основные принципы объектного подхода. Абстрагирование.
3. Основные принципы объектного подхода. Инкапсуляция.
4. Основные принципы объектного подхода. Модульность.
5. Основные принципы объектного подхода. Иерархия.
6. Основные принципы объектного подхода. Типизация.

7. Объект с точки зрения ООП. Состояние. Поведение.
8. Объект с точки зрения ООП. Идентичность и жизненный цикл объектов.
9. Объект с точки зрения ООП. Взаимоотношения между объектами.
10. Классы. Природа классов. Мета модель. Инстанцирование.
11. Классы. Структура класса. Абстрактные классы и интерфейсы.
12. Классы. Принцип подстановки Лисковой. Принцип разделения интерфейсов.
13. Классы. Средства UML для построения диаграмм классов.
14. Классы. Отношения между классами. Ассоциация и агрегация.
15. Классы. Иерархии классов. Зависимость.
16. Модель памяти и структура программы. Классы памяти. Ссылки.
17. Средства абстракции C++. Структура класса. Статические члены и их инициализация
18. Средства инкапсуляции C++. Инкапсуляция и наследование. Друзья.
19. Модульность, раздельная компиляция, пространства имен, using директива.
20. Представление иерархических отношений. Наследование.
21. Представление иерархических отношений. Агрегация. Зависимость по времени жизни.
22. Правила преобразования типов в C++. Параметрический и виртуальный полиморфизм.
23. C++: средства реализации состояния объектов; реализация поведения.
24. Перегрузка операторов.
25. Жизненный цикл объекта. Инициализация массивов. Конструкторы и деструкторы.

Порядок вызова конструкторов и деструкторов при наследовании.

26. Варианты реализации отношения клиент-сервер. Объекты при передаче параметров и возврате из методов.
27. Исключения в C++. Обработка исключений. Умные указатели.
28. Шаблоны классов и шаблоны функций. Специализация.
29. Основы STL. Структура и назначение. Контейнеры.
30. Основы STL. Аллокаторы и итераторы.

2.4. Перечень тем курсовых работ

1. Однотабличная СУБД с обработкой основных *SQL* запросов
2. Редактор трехмерных сцен с использованием *OpenGL*
3. Редактор векторной графики
5. Библиотека графического пользовательского интерфейса с использованием *DirectX*
6. Клиент базы данных с использованием библиотеки *MFC (ODBC)*
7. Клиент базы данных с использованием библиотеки *MFC (ADO/DAO)*
8. Редактор трехмерных сцен с использованием *Direct3D*
9. Электронное обучающее пособие по операторам языка C++
11. Универсальный построитель графиков функций
12. Электронное обучающее пособие по инструкциям языка C++
13. Электронное обучающее пособие по контейнерным классам *STL*
14. Электронное обучающее пособие по библиотеке ввода-вывода C++
16. Объектно-ориентированная библиотека основных алгоритмов линейной алгебры
17. Поисковая система на основе хэш-таблиц
18. Редактор блок-схем
19. Информационная система склада предприятия
20. Система моделирования электрических цепей
21. Система разработки схемы компьютерной сети
22. Редактор трехмерных сцен с использованием *Direct3D*
23. Элементы графического пользовательского интерфейса с использованием библиотеки *OpenGL*

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 написанию и защите курсовой работы

«Отлично» (5 баллов) – получают обучающиеся студенты, оформившие курсовую работу 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 студент допустил не более двух ошибок.

«Удовлетворительно» (3 балла) – получают обучающиеся, оформившие курсовую работу в соответствии с предъявляемыми требованиями. При этом при ответах на вопросы преподавателя обучающийся студент 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/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 «Объектно-ориентированное программирование»

по направлению подготовки/специальности

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)