

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Гнатюк Максим Александрович
Должность: Первый проректор
Дата подписания: 11.07.2022 09:51:21
Уникальный программный ключ:
8873f497f100e798ae8c92c0d38e105c818d5410

Приложение
к рабочей программе дисциплины

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

Алгоритмы и структуры данных

(наименование дисциплины(модуля))

Направление подготовки / специальность

09.03.02 Информационные системы и технологии

(код и наименование)

Направленность (профиль)/специализация

Информационные системы и технологии 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 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.

Формы промежуточной аттестации: экзамены во 2 семестре.

Перечень компетенций, формируемых в процессе освоения дисциплины

Код и наименование компетенции	Код индикатора достижения компетенции
ОПК-6 Способен разрабатывать алгоритмы и программы, пригодные для практического применения в области информационных систем и технологий	ОПК-6.1 Разрабатывать алгоритмы и программы, пригодные для практического применения в области информационных систем и технологий

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

Код и наименование индикатора достижения компетенции	Результаты обучения по дисциплине	Оценочные материалы
ОПК-6.1 Разрабатывать алгоритмы и программы, пригодные для практического применения в области информационных систем и технологий	Обучающийся знает: области применения различных структур данных в задачах проектирования программного обеспечения информационных систем; области применения типовых алгоритмов в базовых и прикладных информационных технологиях; особенности использования алгоритмов в задачах обработки больших массивов данных; общие принципы программной реализации алгоритмов различных классов; способы реализации алгоритмов по математическому описанию проблемы; современные информационные, математические и программные средства реализации алгоритмов.	Вопросы (1-12)
	Обучающийся умеет: определить структуры данных для задачи проектирования в информационных системах; осуществлять оптимальный выбор алгоритма для решения поставленной задачи; выявлять при проектировании критичные к скорости обработки участки алгоритма и оптимизировать их; правильно подбирать необходимые средства разработки; использовать современные технологии для реализации информационных технологий; разрабатывать программное обеспечение для многократного использования в различных информационных технологиях.	Вопросы (10-25)
	Обучающийся владеет: навыками использования структур данных; навыками проектирования базовых и прикладных информационных технологий; навыками анализа алгоритмов; методами выбора современных технических и программных средств для оптимальной реализации алгоритмов; навыками необходимыми для разработки современных алгоритмических решений; навыками использования современных технических и программных средств для разработки алгоритмов.	Вопросы (23-39)

Промежуточная аттестация (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. 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-6.1 Разрабатывать алгоритмы и программы, пригодные для практического применения в области информационных систем и технологий	Обучающийся знает: области применения различных структур данных в задачах проектирования программного обеспечения информационных систем; области применения типовых алгоритмов в базовых и прикладных информационных технологиях; особенности использования алгоритмов в задачах обработки больших массивов данных; общие принципы программной реализации алгоритмов различных классов; способы реализации алгоритмов по математическому описанию проблемы; современные информационные, математические и программные средства реализации алгоритмов.

Примеры вопросов/заданий

1. Особенности этапов разработки алгоритмов.
2. Оценка производительности алгоритма, оптимизация алгоритмов.
3. Область применения динамических типов данных. Достоинства и недостатки.
4. Варианты использования алгоритмов на графах.
5. Конечные автоматы и алгоритмы распознавания.
6. Анализ, декомпозиция, синтез.
7. Алгоритмы поиска.
8. Особенности использования рекурсивных алгоритмов.
9. Методы оптимизации алгоритмов.
10. Алгоритмы сортировки.
11. Свойства алгоритма.
12. Использование псевдокода для описания алгоритма.
13. Способы графического представления алгоритма.
14. Асимптотическая нотация анализа алгоритмов.
15. Абстрактные типы данных.
16. Особенности алгоритмов сортировки.
17. Древовидные структуры и алгоритмы на их основе.
18. Бинарные деревья поиска.
19. Реализация алгоритмов в стандартной библиотеке STL.
20. Использование шаблонов в алгоритмах на языке C++.

¹ 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 несет заведующий кафедрой и преподаватель – разработчик оценочных средств.

2.2 Типовые задания для оценки навыкового образовательного результата

Проверяемый образовательный результат:

Код и наименование индикатора достижения компетенции	Образовательный результат
ОПК-6.1 Разрабатывать алгоритмы и программы, пригодные для практического применения в области информационных систем и технологий	Обучающийся умеет: определить структуры данных для задачи проектирования в информационных системах; осуществлять оптимальный выбор алгоритма для решения поставленной задачи; выявлять при проектировании критичные к скорости обработки участки алгоритма и оптимизировать их; правильно подбирать необходимые средства разработки; использовать современные технологии для реализации информационных технологий; разрабатывать программное обеспечение для многократного использования в различных информационных технологиях.
<p><i>Примеры заданий</i></p> <ol style="list-style-type: none"> 1. Составьте алгоритм поиска по фамилии с предварительной сортировкой списка. 2. Составьте алгоритм поиска с использованием бинарного дерева. 3. Выполните сравнительный анализ таких алгоритмов сортировки как сортировка выбором и сортировка слиянием. 4. Составьте алгоритм поиска заданного ключа. 5. Составьте алгоритм поиска минимального значения массива. 6. Составьте алгоритм поиска максимального значения массива. 7. Составьте алгоритм вставки элемента в массив. 8. Составьте алгоритм удаления элемента из массива. <p><i>Примеры заданий</i></p> <ol style="list-style-type: none"> 1. Структура это: <ol style="list-style-type: none"> a. Индексированный набор элементов одного типа b. Массив, хранящий строку символов c. Набор различных элементов (полей записи), хранимых как единое целое и предусматривающий доступ к отдельным полям структуры d. Место для хранения адреса в памяти компьютера 2. Динамическими называются данные, которые: <ol style="list-style-type: none"> a. Можно размещать в памяти машины в процессе исполнения программы и освобождать память, когда эти данные становятся ненужными b. Можно получить на стадии компиляции программы c. Определяются программистом с помощью специального ключевого слова d. Добавляются в процессе сборки программы 3. Стек является: <ol style="list-style-type: none"> a. Односвязной линейной структурой b. Многосвязной структурой c. Кольцевой структурой d. Разветвленной структурой 4. Очередь является: <ol style="list-style-type: none"> a. Односвязной линейной структурой b. Многосвязной структурой c. Кольцевой структурой d. Файлом 5. Какие операции допускает список: <ol style="list-style-type: none"> a. Перебор элементов списка b. Поиск заданного элемента c. Проверка содержимого списка d. Вставка в список нового элемента 	
ОПК-6.1 Разрабатывать алгоритмы и программы, пригодные	Обучающийся владеет: навыками использования структур

<p>для практического применения в области информационных систем и технологий</p>	<p>данных; навыками проектирования базовых и прикладных информационных технологий; навыками анализа алгоритмов; методами выбора современных технических и программных средств для оптимальной реализации алгоритмов; навыками необходимыми для разработки современных алгоритмических решений; навыками использования современных технических и программных средств для разработки алгоритмов.</p>
<p style="text-align: center;"><i>Примеры заданий</i></p> <ol style="list-style-type: none"> 1. Какие существуют способы обхода бинарных деревьев? 2. В каких случаях выгодно использовать рекурсивные алгоритмы? 3. Способы обхода графа. 4. От чего зависит глубина рекурсии? 5. Реализовать алгоритм поиска подстроки в строке. 6. Реализовать алгоритм поиска минимального значения в двумерном массиве целых чисел. 7. Реализовать алгоритм сортировки значений по возрастанию в столбца двумерного массива. 8. Построить алгоритм сортировки значений по убыванию в строках двумерного массива. 9. Построить алгоритм вычисления суммы главной диагонали двумерного массива. 10. Используя библиотеку STL, составить алгоритм реализующий поиск элемента в кольцевом списке.. 11. Используя библиотеку STL реализовать алгоритм сортировки по фамилиям в алфавитном порядке. 12. Используя библиотеку STL, реализовать слияние двух массивов целых чисел, с последующей сортировкой по убыванию. <p style="text-align: center;">Задание</p> <ol style="list-style-type: none"> 1. В соответствии с вариантом исходных данных из табл. нужно составить алгоритм поиска. 2. Выполнить асимптотический анализ полученного алгоритма. 3. Построить блок схему алгоритма. 4. Выполнить описание алгоритма псевдокодом. 	

2.3. Перечень вопросов для подготовки обучающихся к промежуточной аттестации

1. Простые типы данных
2. Целочисленные типы данных
3. Вещественные типы данных
4. Логические типы данных
5. Базовые алгоритмические структуры ветвления
6. Базовые алгоритмические структуры циклов
7. Указатели и ссылки
8. Функции
9. Массивы
10. Записи
11. Записи с вариантами
12. Множество
13. Представление массивов
14. Представление записей
15. Представление множеств
16. Последовательный файл
17. Операции над файлами
18. Файлы со сложной структурой
19. Текстовые файлы
20. Бинарные файлы
21. Основы сортировки
22. Сортировка массивов
23. Сортировка простыми включениями
24. Сортировка простым выбором
25. Сортировка простым обменом
26. Сортировка Шелла
27. Сортировка с разделением
28. Алгоритм поиска медианы
29. Рекурсивные алгоритмы
30. Динамические структуры данных
31. Ссылки и указатели
32. Линейные списки
33. Упорядоченные списки
34. Однонаправленный список
35. Двухнаправленный список
36. Кольцевой список
37. Очередь
38. Стек
39. Древовидные структуры

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 объема 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 экзамену

«**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.