

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)**

Технологии программирования

(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: зачет в 4 семестре, курсовая работа, экзамен в 5 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 6.2. Использует технологию программирования для написания программ пригодных для практического применения

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

Код и наименование индикатора достижения компетенции	Результаты обучения по дисциплине	Оценочные материалы
ОПК 6.2. Использует технологию программирования для написания программ пригодных для практического применения	Обучающийся знает: технологию программирования для написания программ пригодных для практического применения	Вопросы (1 – 10)
	Обучающийся умеет: использовать технологию программирования для написания программ пригодных для практического применения	Задания (1-5)
	Обучающийся владеет: навыками технологии программирования для написания программ пригодных для практического применения	Задания (6-10)

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

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

Промежуточная аттестация (курсовая работа).

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 6.2. Использует	Обучающийся знает: технологию программирования для написания программ

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 несут заведующий кафедрой и преподаватель – разработчик оценочных средств.

технологии программирования для написания программ пригодных для практического применения	пригодных для практического применения
<ul style="list-style-type: none"> • <i>Примеры вопросов/заданий</i> <ol style="list-style-type: none"> 1.Технология программирования и основные этапы ее развития. 2.Проблемы разработки сложных программных систем. 3.Блочнo-иерархический подход к созданию сложных систем. 4.Дружественность, жизненный цикл программы. 5.Этапы разработки программного обеспечения. 6.Постановка задачи и спецификация программы. 7.Эволюция разработки программного обеспечения. 8.Технология RAD. 9.Критерии качества программы. 10.Основные этапы решения задач на ЭВМ. 	

2.2 Типовые задания для оценки навыкового образовательного результата

Проверяемый образовательный результат:

Код и наименование индикатора достижения компетенции	Образовательный результат
ОПК 6.2. Использует технологию программирования для написания программ пригодных для практического применения	Обучающийся умеет: использовать технологию программирования для написания программ пригодных для практического применения
<ul style="list-style-type: none"> • <i>Примеры заданий</i> <ol style="list-style-type: none"> 1. Как написать следующее выражение «Второму элементу массива Myarray присвоено значение пяти»? <ol style="list-style-type: none"> a) <code>int [1] Myarray=«пять»</code> b) <code>int Myarray [1] = 5</code> c) <code>int Myarray [2] = «пять»</code> d) <code>int Myarray [2] = 5</code> 2. Как написать следующее выражение «Если переменная index больше size то мы инкрементируем переменную count»? <ol style="list-style-type: none"> a) <code>if (index>size) { count++; }</code> b) <code>if (index<size) { count--; }</code> c) <code>if (index>=size) { ++count; }</code> d) <code>if (index<size) { --count; }</code> 3. Как обозначается в языке C (C++) следующий режим работы с потоком - создание нового файла для записи и чтения? <ol style="list-style-type: none"> a) a+ b) wb c) w+ d) w+b 4. Какая функция, описанная в заголовочном файле читает строку символов из файла? <ol style="list-style-type: none"> a) <code>gets()</code> b) <code>fputs()</code> c) <code>fgets()</code> d) <code>fscanf()</code> 5. Какой размер массива M будет после выполнения кода: <code>char M[] = "\nGoodlive" ?</code> <ol style="list-style-type: none"> a) 10 b) 8 c) 9 d) Не определен 	

Код и наименование индикатора достижения компетенции	Образовательный результат
ОПК 6.2. Использует технологию программирования для написания программ пригодных для практического применения	Обучающийся владеет: навыками технологии программирования для написания программ пригодных для практического применения
<p>• <i>Примеры заданий</i></p> <p>6. Технологией программирования называют:</p> <ol style="list-style-type: none"> способы описания проектируемой системы, точнее модели, используемой на конкретном этапе разработки; последовательность выполнения технологических операций; совокупность методов и средств, используемых в процессе разработки программного обеспечения. <p>7. Структурный подход к программированию:</p> <ol style="list-style-type: none"> определяется как технология создания сложного программного обеспечения, основанная на представлении программы в виде совокупности объектов; предполагает построение программного обеспечения из отдельных компонентов – физически отдельно существующих частей программного обеспечения, которое взаимодействует между собой через стандартизированные двоичные интерфейсы; представляет собой совокупность рекомендуемых технологических приемов, охватывающих выполнение всех этапов разработки программного обеспечения. <p>8. Жизненным циклом программного обеспечения называют:</p> <ol style="list-style-type: none"> период от момента появления идеи создания некоторого программного обеспечения до момента завершения его поддержки фирмой-разработчиком или фирмой, выполнявшей сопровождение; точное формализованное описание функцией и ограничений разрабатываемого программного обеспечения; действующий программный продукт, реализующий отдельные функции и внешние интерфейсы разрабатываемого программного обеспечения. <p>9. Спецификациями называют:</p> <ol style="list-style-type: none"> период от момента появления идеи создания некоторого программного обеспечения до момента завершения его поддержки фирмой-разработчиком или фирмой, выполнявшей сопровождение; точное формализованное описание функцией и ограничений разрабатываемого программного обеспечения; действующий программный продукт, реализующий отдельные функции и внешние интерфейсы разрабатываемого программного обеспечения. <p>10. Цикл:</p> <ol style="list-style-type: none"> это линейная последовательность действий; управление передается одному из двух блоков в зависимости от истинности или ложности условия; повторение некоторой группы действий по условию. 	

2.3 Задания или иные материалы для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих уровень сформированности компетенций по курсовой работе:

Тема: Функции. Перегрузка и шаблоны функций.

Задания к части 1. Функции

Вариант 1(3)

В одномерном массиве, состоящем из n вещественных элементов, вычислить:

- сумму отрицательных элементов массива;
- произведение элементов массива, расположенных между максимальным и минимальным элементами. Упорядочить элементы массива по возрастанию.

Вариант 1(4)

Дана целочисленная прямоугольная матрица. Определить:

- количество строк, не содержащих ни одного нулевого элемента;

2) максимальное из чисел, встречающихся в заданной матрице более одного раза.

Функции, строки и файлы

Вариант 1(5)

Написать программу, которая считывает из текстового файла три предложения и выводит их в обратном порядке.

Задания к части 2. Шаблоны функций

Выполнить задания («Одномерные массивы») и («Двумерные массивы»), оформив каждый пункт задания в виде шаблона функции. Все необходимые данные для функций должны передаваться им в качестве параметров. Использование глобальных переменных в функциях не допускается. Привести примеры программ, использующих эти шаблоны для типов int, float и double.

Код и наименование индикатора достижения компетенции	Образовательный результат
ОПК 6.2. Использует технологию программирования для написания программ пригодных для практического применения	Обучающийся знает: технологию программирования для написания программ пригодных для практического применения
<p><i>Примеры вопросов/заданий</i></p> <p>1. Все необходимые данные для функций должны передаваться им в качестве параметров. Использование глобальных переменных в функциях не допускается.</p> <p>2. При написании программ можно использовать как динамические, так и нединамические массивы. Размерность последних задается именованной константой.</p> <p>3. Рекомендуется выполнять каждое задание в двух вариантах: используя локальные и динамические массивы. Размерности локальных массивов задавать именованными константами, значения элементов массива – в списке инициализации. Ввод данных в динамический массив выполнять из файла.</p> <p>4. Все необходимые данные для функций должны передаваться им в качестве параметров. Использование глобальных переменных в функциях не допускается.</p>	
ОПК 6.2. Использует технологию программирования для написания программ пригодных для практического применения	Обучающийся умеет и владеет: использовать технологию программирования для написания программ пригодных для практического применения способами конструирования программ с использованием структурной парадигмы; навыками технологии программирования для написания программ пригодных для практического применения;
<p><i>Примеры вопросов/заданий</i></p> <p>1. Выполнить задания («Одномерные массивы») и («Двумерные массивы»), оформив каждый пункт задания в виде функции.</p> <p>2. Выполнить задания («Строки и файлы»), оформив в виде функций законченные последовательности действий.</p> <p>3. Написать программу, которая считывает из текстового файла три предложения и выводит их в обратном порядке.</p> <p>4. Выполнить задания («Одномерные массивы») и («Двумерные массивы»), оформив каждый пункт задания в виде шаблона функции. Все необходимые данные для функций должны передаваться им в качестве параметров. Использование глобальных переменных в функциях не допускается. Привести примеры программ, использующих эти шаблоны для типов int, float и double.</p>	

2.4 Перечень вопросов для подготовки к защите по курсовой работе:

1. Что такое функция?
2. Назовите функцию, с которой начинается выполнение программы.
3. В какой момент начинает выполняться функция?
4. Что необходимо указать для вызова функции?
5. Что такое объявление функции?
6. Что такое определение функции?
7. С помощью какого модификатора функцию можно определить, как встроенную, для чего он применяется?
8. Что совместно определяют тип возвращаемого параметра и типы параметров?
9. Что называется формальными и фактическими параметрами (аргументами)?

10. Что происходит при передаче параметров в функцию по значению?
11. Что происходит при передаче параметров в функцию по адресу?
12. Что передается в функцию при использовании в качестве параметра массива?
13. Может ли функция иметь несколько параметров со значениями по умолчанию?
14. Что такое рекурсивная функция?
15. Что такое многофайловый проект и как его создать?
16. Что называется перегрузкой функции?
17. Что такое разрешение перегрузки?
18. Когда применяется перегрузка?
19. Что необходимо при написании перегруженных функций?
20. В каком случае функции не могут быть перегружены?
21. Для чего применяются шаблоны?
22. Что такое инстанцирование шаблона функции?
23. Как можно перегружать шаблоны?

2.5. Перечень вопросов для подготовки обучающихся к промежуточной аттестации

1. Технология программирования и основные этапы ее развития.
2. Проблемы разработки сложных программных систем.
3. Блочный-иерархический подход к созданию сложных систем.
4. Дружественность, жизненный цикл программы.
5. Этапы разработки программного обеспечения.
6. Постановка задачи и спецификация программы.
7. Эволюция разработки программного обеспечения.
8. Технология RAD.
9. Критерии качества программы.
10. Основные этапы решения задач на ЭВМ.
11. Постановка задачи и спецификация программы.
12. Способы записи алгоритма.
13. Способы конструирования программ.
14. Основные понятия структурного программирования.
15. Линейные программы.
16. Представление основных структур программирования: итерация, ветвление, повторение.
17. Программирование рекурсивных алгоритмов.
18. Стандартные типы данных.
19. Типы данных, определяемые пользователем.
20. Переменные и выражения: переменные, операции, выражения.
21. Указатели и массивы: указатели, ссылки, массивы.
22. Процедуры.
23. Объявление и определение функций.
24. Параметры функции.
25. Шаблоны функций.
26. Функция main().
27. Функции стандартной библиотеки.
28. Директивы препроцессора: #include, #define, #undef.
29. Директивы условной компиляции.
30. Области действия идентификаторов.
31. Внешние объявления.
32. Поименованные области.
33. Модульные программы.
34. Кодирование и документирование программы.
35. Проектирование и тестирование программы.
36. Динамические структуры данных.
37. Списки: основные виды и способы реализации.
38. Стеки.

39.Очереди.

40.Бинарные деревья.

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.*

Процедура и критерии формирования оценок по написанию и защите курсовой работы

Оценивание защиты курсовой работы проводится руководителем курсовой работы. По результатам проверки курсовой работы обучающийся допускается к ее защите при условии соблюдения перечисленных условий:

- выполнены все задания;
- сделаны выводы;
- отсутствуют ошибки;
- оформлено в соответствии с требованиями.

В том случае, если работа не отвечает предъявляемым требованиям, то она возвращается автору на доработку. Обучающийся должен переделать работу с учетом замечаний и предоставить для проверки вариант с результатами работы над ошибками. Если сомнения вызывают отдельные аспекты курсовой работы, то в этом случае они рассматриваются во время устной защиты работы.

Защита курсовой работы представляет собой устный публичный отчет обучающегося о результатах выполнения, ответы на вопросы преподавателя. Ответ обучающегося оценивается преподавателем в соответствии с критериями.

Критерии формирования оценок по написанию и защите курсовой работы

«Отлично» (5 баллов) – получают обучающиеся, 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 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 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/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.