

Документ подписан простой электронной подписью
 Информация о владельце:
 ФИО: Гаранин Максим Александрович
 Должность: Ректор
 Дата подписания: 08.10.2025 15:11:28
 Уникальный программный ключ:
 7708e3a47e66a8ee02711b298d7c78bd1e40bf88

ФЕДЕРАЛЬНОЕ АГЕНТСТВО ЖЕЛЕЗНОДОРОЖНОГО ТРАНСПОРТА
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«ПРИВОЛЖСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ПУТЕЙ СООБЩЕНИЯ»

МОДУЛЬ "ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ И ПРОГРАММИРОВАНИЕ"

Технологии программирования

рабочая программа дисциплины (модуля)

Направление подготовки 09.03.02 Информационные системы и технологии
 Направленность (профиль) Информационные системы и технологии на транспорте
 Квалификация **бакалавр**
 Форма обучения **очная**
 Общая трудоемкость **7 ЗЕТ**

Виды контроля в семестрах:

экзамены 5
 зачеты 4
 курсовые работы 5

Распределение часов дисциплины по семестрам

Семестр (<Курс>.<Семестр на курсе>)	4 (2.2)		5 (3.1)		Итого	
Неделя	16 1/6		16			
Вид занятий	УП	РП	УП	РП	УП	РП
Лекции	16	16	16	16	32	32
Лабораторные	16	16			16	16
Практические	16	16	16	16	32	32
Конт. ч. на аттест.			1	1	1	1
Конт. ч. на аттест. в период ЭС	0,25	0,25	2,3	2,3	2,55	2,55
В том числе в форме практ.подготовки	32	32	50	50	82	82
Итого ауд.	48	48	32	32	80	80
Контактная работа	48,25	48,25	35,3	35,3	83,55	83,55
Сам. работа	51	51	84	84	135	135
Часы на контроль	8,75	8,75	24,7	24,7	33,45	33,45
Итого	108	108	144	144	252	252

Программу составил(и):

к.т.н., доцент, Фатеев В.А.

Рабочая программа дисциплины

Технологии программирования

разработана в соответствии с ФГОС ВО:

Федеральный государственный образовательный стандарт высшего образования - бакалавриат по направлению подготовки 09.03.02 Информационные системы и технологии (приказ Минобрнауки России от 19.09.2017 г. № 926)

составлена на основании учебного плана: 09.03.02-25-4-ИСТб.plm.plx

Направление подготовки 09.03.02 Информационные системы и технологии Направленность (профиль) Информационные системы и технологии на транспорте

Рабочая программа одобрена на заседании кафедры

Цифровые технологии

Зав. кафедрой к.т.н., доцент Авсиевич А.В.

1. ЦЕЛИ ОСВОЕНИЯ ДИСЦИПЛИНЫ (МОДУЛЯ)

1.1	Целью изучения дисциплины «Технологии программирования» является формирование у обучаемых знаний, умений и навыков (уровня сформированности соответствующих компетенций) в результате последовательного изучения содержательно связанных между собой разделов (тем) учебных занятий. Дисциплина развивает навыки разработки алгоритмов, структурного программирования, ставит целью обучения студента принципам, используемым при разработке и реализации иерархии классов объектов, современным визуальным средствам разработки и создания программ с использованием библиотек компонентов.
-----	--

2. МЕСТО ДИСЦИПЛИНЫ (МОДУЛЯ) В СТРУКТУРЕ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

Цикл (раздел) ОП:	Б1.О.12.01
-------------------	------------

3. КОМПЕТЕНЦИИ ОБУЧАЮЩЕГОСЯ, ФОРМИРУЕМЫЕ В РЕЗУЛЬТАТЕ ОСВОЕНИЯ ДИСЦИПЛИНЫ (МОДУЛЯ)

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

В результате освоения дисциплины (модуля) обучающийся должен

3.1	Знать:
3.1.1	технологию программирования для написания программ пригодных для практического применения.
3.2	Уметь:
3.2.1	использовать технологию программирования для написания программ пригодных для практического применения.
3.3	Владеть:
3.3.1	навыками технологии программирования для написания программ пригодных для практического применения.

4. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

Код занятия	Наименование разделов и тем /вид занятия/	Семестр / Курс	Часов	Примечание
	Раздел 1. Технологии программирования. Основные понятия и подходы			
1.1	Технология программирования и основные этапы ее развития. Проблемы разработки сложных программных систем. Блочный-иерархический подход к созданию сложных систем /Лек/	4	1	
1.2	Дружественность, жизненный цикл программы. Этапы разработки программного обеспечения /Лек/	4	2	
1.3	Постановка задачи и спецификация программы. Эволюция разработки программного обеспечения. Технология RAD. Критерии качества программы /Лек/	4	1	
1.4	Интегрированная среда Microsoft Visual Studio 2010 и CodeBlocs /Пр/	4	2	Практическая подготовка
	Раздел 2. Методы построения алгоритмов			
2.1	Основные этапы решения задач на ЭВМ. Постановка задачи и спецификация программы. Способы записи алгоритма /Лек/	4	2	презентация, мультимедийн
2.2	Линейные программы. Организация ввода-вывода /Пр/	4	2	Практическая подготовка
2.3	Способы конструирования программ. Основные понятия структурного программирования. Линейные программы. Представление основных структур программирования: итерация, ветвление, повторение. Операторы передачи управления. Идентификаторы, ключевые слова, знаки операций, константы, комментарии. /Лек/	4	2	презентация, мультимедийн ый проектор
2.4	Линейные программы. Организация ввода-вывода /Лаб/	4	2	Практическая подготовка.
2.5	Разветвляющиеся программы. Циклы /Пр/	4	4	Практическая подготовка
2.6	Разветвляющиеся программы. Циклы /Лаб/	4	2	Практическая подготовка.
2.7	Рекурсивные функции. Программирование рекурсивных алгоритмов. Диалоговые программы. Программа на языке высокого уровня /Лек/	4	2	
2.8	Изучение особенностей циклических и рекурсивных алгоритмов /Ср/	4	4	
	Раздел 3. Базовые средства языка C++			
3.1	Состав языка. Стандартные типы данных. Типы данных, определяемые пользователем /Лек/	4	2	

3.2	Переменные и выражения: переменные, операции, выражения /Лек/	4	2	
3.3	Указатели и массивы: указатели, ссылки, массивы. Процедуры. Записи. Файлы /Лек/	4	2	
3.4	Одномерные массивы и указатели /Пр/	4	4	Практическая подготовка
3.5	Одномерные массивы и указатели /Лаб/	4	2	Практическая подготовка.
3.6	Двумерные массивы /Пр/	4	4	Практическая подготовка
3.7	Двумерные массивы /Лаб/	4	4	Практическая подготовка.
3.8	Строки и файлы /Лаб/	4	4	Практическая подготовка.
3.9	Структуры /Лаб/	4	2	Практическая подготовка.
3.10	Изучение методов работы с файлами, статическими и динамическими массивами /Ср/	4	7	
Раздел 4. Самостоятельная работа				
4.1	Подготовка к лекциям /Ср/	4	8	
4.2	Подготовка к практическим занятиям /Ср/	4	16	
4.3	Подготовка к лабораторным занятиям /Ср/	4	16	
Раздел 5. Контактные часы на аттестацию				
5.1	Зачет /КЭ/	4	0,25	
Раздел 6. Модульное программирование				
6.1	Объявление и определение функций. Параметры функции. Шаблоны функций. Функция main(). Функции стандартной библиотеки /Лек/	5	2	
6.2	Глобальные переменные. Возвращаемое значение. Перегрузка функций (Чтение текста первоисточника, конспектирование текста) /Ср/	5	15	
6.3	Директивы препроцессора: #include, #define, #undef. Директивы условной компиляции /Лек/	5	2	
6.4	Строки и файлы /Пр/	5	2	Практическая подготовка
6.5	Области действия идентификаторов. Внешние объявления. Поименованные области. Модульные программы /Лек/	5	4	
6.6	Функции /Пр/	5	4	Практическая подготовка
6.7	Перегрузка и шаблоны функций /Пр/	5	4	Практическая подготовка
Раздел 7. Технология создания программ				
7.1	Кодирование и документирование программы. Проектирование и тестирование программы /Лек/	5	2	
7.2	Анализ требований и определение спецификаций программного обеспечения при структурном подходе. (Чтение текста первоисточника, конспектирование текста) /Ср/	5	10	
7.3	Динамические структуры данных. Списки: основные виды и способы реализации. Стеки. Очереди. Бинарные деревья /Лек/	5	2	
7.4	Реализация динамических структур с помощью массивов. Основы доказательства правильности. /Лек/	5	2	
7.5	Динамические структуры данных /Пр/	5	2	Практическая подготовка
7.6	Структуры /Пр/	5	4	Практическая подготовка
7.7	Стандартная библиотека шаблонов (STL) в C++. Статические и динамические библиотеки. /Лек/	5	2	
Раздел 8. Самостоятельная работа				
8.1	Подготовка к лекциям /Ср/	5	8	

8.2	Подготовка к практическим занятиям /Ср/	5	16	
8.3	Курсовая работа /Ср/	5	35	Практическая подготовка
Раздел 9. Контактные часы на аттестацию				
9.1	Курсовая работа /КА/	5	1	
9.2	Экзамен /КЭ/	5	2,3	

5. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ

Оценочные материалы для проведения промежуточной аттестации обучающихся приведены в приложении к рабочей программе дисциплины.

Формы и виды текущего контроля по дисциплине (модулю), виды заданий, критерии их оценивания, распределение баллов по видам текущего контроля разрабатываются преподавателем дисциплины с учетом ее специфики и доводятся до сведения обучающихся на первом учебном занятии.

Текущий контроль успеваемости осуществляется преподавателем дисциплины (модуля) в рамках контактной работы и самостоятельной работы обучающихся. Для фиксирования результатов текущего контроля может использоваться ЭИОС.

6. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)

6.1. Рекомендуемая литература

6.1.1. Основная литература

	Авторы, составители	Заглавие	Издательство, год	Эл. адрес
Л1.1	Иванова Г.С.	Технология программирования	Москва: КноРус, 2018	http://www.book.ru/book/926

6.1.2. Дополнительная литература

	Авторы, составители	Заглавие	Издательство, год	Эл. адрес
Л2.1	Доррер А. Г.	Управление ИТ-проектами: учебное пособие	Красноярск: СибГУ им. академика М.Ф. Решетнёва, 2019	http://e.lanbook.com/book/14

6.2 Информационные технологии, используемые при осуществлении образовательного процесса по дисциплине (модулю)

6.2.1 Перечень лицензионного и свободно распространяемого программного обеспечения

6.2.1.1	Операционная система Microsoft® Windows Professional 8 Russian Upgrade OLP NL Academic Edition Договор на поставку № 0342100004813000011 от года.
6.2.1.2	Microsoft Office 2013 Professional Договор № 0342100004814000045
6.2.1.3	Microsoft Visio 2016 Договор №034210000481600009
6.2.1.4	Распространяется свободно, под лицензией GNU: Code Bloks,
6.2.1.5	Dia http://dia-installer.de/ (GNU GENERAL PUBLIC LICENSE)

6.2.2 Перечень профессиональных баз данных и информационных справочных систем

6.2.2.1	Крупнейший веб-сервис для хостинга IT-проектов и их совместной разработки- https://github.com/
6.2.2.2	База книг и публикаций Электронной библиотеки "Наука и Техника" - http://www.n-t.ru
6.2.2.3	Портал для разработчиков электронной техники: http://www.espec.ws/
6.2.2.4	База данных «Библиотека программиста» https://proglib.io/
6.2.2.5	База данных «Отраслевой портал специалистов» http://www.connect-wit.ru/
6.2.2.6	Гарант.ру https://www.garant.ru/
6.2.2.7	КонсультантПлюс http://www.consultant.ru/

7. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ (МОДУЛЯ)	
7.1	Учебные аудитории для проведения занятий лекционного типа, укомплектованные специализированной мебелью и техническими средствами обучения: мультимедийное оборудование для предоставления учебной информации большой аудитории и/или звукоусиливающее оборудование (стационарное или переносное).
7.2	Учебные аудитории для проведения занятий семинарского типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации, укомплектованные специализированной мебелью и техническими средствами обучения: мультимедийное оборудование и/или звукоусиливающее оборудование (стационарное или переносное)
7.3	Помещения для самостоятельной работы, оснащенные компьютерной техникой с возможностью подключения к сети "Интернет" и обеспечением доступа в электронную информационно-образовательную среду университета.
7.4	Помещения для хранения и профилактического обслуживания учебного оборудования

**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 Университета.

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 Университета.

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 технологию программирования	Обучающийся знает: технологию программирования для написания программ пригодных для практического применения

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.

для написания программ пригодных для практического применения	
- Примеры вопросов/заданий 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
- Примеры заданий 1. Как написать следующее выражение «Второму элементу массива Myarray присвоено значение пяти»? a) int [1] Myarray=«пять» b) int Myarray [1] = 5 c) int Myarray [2] = «пять» d) int Myarray [2] = 5 2. Как написать следующее выражение «Если переменная index больше size то мы инкрементируем переменную count»? a) if (index>size) { count++; } b) if (index<size) { count--; } c) if (index>=size) { ++count; } d) if (index<size) { --count; } 3. Как обозначается в языке C (C++) следующий режим работы с потоком - создание нового файла для записи и чтения? a) a+ b) wb c) w+ d) w+b 4. Какая функция, описанная в заголовочном файле читает строку символов из файла? a) gets() b) fputs() c) fgets() d) fscanf() 5. Какой размер массива M будет после выполнения кода: char M[] = "\nGoodlive" ? 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
- Примеры заданий 6. Технологией программирования называют: - 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. 7. Структурный подход к программированию: - 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. 8. Жизненным циклом программного обеспечения называют: - 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. 9. Спецификациями называют: - 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. 10. Цикл: - 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
<i>Примеры вопросов/заданий</i> 1. Все необходимые данные для функций должны передаваться им в качестве параметров. Использование глобальных переменных в функциях не допускается. 2. При написании программ можно использовать как динамические, так и нединамические массивы. Размерность последних задается именованной константой. 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. 4. Все необходимые данные для функций должны передаваться им в качестве параметров. Использование глобальных переменных в функциях не допускается.	
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;
<i>Примеры вопросов/заданий</i> 1. Выполнить задания («Одномерные массивы») и («Двумерные массивы»), оформив каждый пункт задания в виде функции. 2. Выполнить задания («Строки и файлы»), оформив в виде функций законченные последовательности действий. 3. Написать программу, которая считывает из текстового файла три предложения и выводит их в обратном порядке. 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 <code>int</code> , <code>float</code> и <code>double</code> .	

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 зачету

«Зачтено» – 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. Данная оценка выставляется при условии выполнения студентом всех лабораторных работ и не менее 80% обучающих элементов, входящих в учебно-методический комплекс изучаемой дисциплины, а именно: практических работ, прохождения промежуточного тестирования и форум-опросов с правильным количеством ответов – 100 – 75 % от общего объёма заданных тестовых вопросов.

«Не зачтено» – выставляется в том случае, когда обучающийся демонстрирует фрагментарные знания основных разделов программы изучаемого курса: его базовых понятий и фундаментальных проблем. У обучающегося 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. Данная оценка выставляется при условии не выполнения студентом 80% всех обучающих элементов, входящих в учебно-методический комплекс изучаемой дисциплины, а именно: лабораторных и практических работ, форум-опросов, прохождения промежуточного тестирования с правильным количеством ответов 59 % и менее от общего объёма заданных тестовых вопросов.

Кроме того, выбор значения балла-оценки может быть сделан преподавателем по данным балльно-рейтинговой системы, которая формируется автоматически при ведении электронного журнала.

Критерии формирования оценок по экзамену

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

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

«Удовлетворительно» – студент допустил существенные ошибки.

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