МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

«ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ УПРАВЛЕНИЯ
И РАДИОЭЛЕКТРОНИКИ» (ТУСУР)

УТВЕРЖДАЮ

Проректор по учебной работе
_____ Троян П.Е.

«___» _____ 2016 г.

Рабочая программа учебной дисциплины

ИНФОРМАТИКА И ПРОГРАММИРОВАНИЕ

Уровень основной образовательной программы: **бакалавриат**

Направление подготовки: **09.03.04 «Программная инженерия»**

Форма обучения: **очная**

Факультет систем управления (ФСУ)

Кафедра автоматизации обработки информации (АОИ)

Курс 1, 2 Семестр 1, 2, 3

Учебный план набора 2015 года

Распределение рабочего времени:

Виды учебной работы	Семестр 1	Семестр 2	Семестр 3	Всего	Единицы
1. Лекции	18	26	36	80	часов
2. Практические занятия	-	18	-	18	часов
3. Лабораторные работы	36	16	36	**88**	часов
4. Курсовой проект/работа (КРС) (аудиторная)	*не предусмотрено*				
5. Всего аудиторных занятий (сумма 1, 2, 3)	**54**	**60**	**72**	**186**	часов
6. Из них в интерактивной форме	*не предусмотрено*				
7. Самостоятельная работа студентов (СРС)	90	84	36	210	часов
8. Всего (без экзамена) (сумма 5, 7)	144	144	108	396	часов
9. Самостоятельная работа на подготовку, сдачу экзамена	36	36	36	108	часов
10. **Общая трудоемкость (сумма 8, 9)**	**180**	**180**	**144**	**504**	часа
(в зачетных единицах)	5	5	4	15	ЗЕТ

Экзамен — 1, 2, 3 (первый, второй, третий) семестр

Томск 2016

Согласована на портале № 5570

Рассмотрена и одобрена на заседании кафедры
протокол № 300 от « 20 » 10 2016 г.

Лист согласований

Рабочая программа для дисциплины **«Информатика и программирование» (Б1.Б.14)** составлена с учетом требований Федерального государственного образовательного стандарта высшего образования (ФГОС ВО) по направлению подготовки бакалавра 09.03.04 «Программная инженерия», утвержденного приказом Министерства образования и науки РФ 12.03.2015 г. № 229, рассмотрена и утверждена на заседании кафедры «___» _____ 20__ г., протокол № _____.

Разработчики:
ст. преподаватель каф. АОИ _____ Пермякова А.В.

ст. преподаватель каф. АОИ _____ Морозова Ю.В.

Рабочая программа согласована с факультетом, профилирующей и выпускающей кафедрами направления подготовки (специальности).

Декан ФСУ _____ Сенченко П.В.

Зав. профилирующей
выпускающей кафедрой _____ Ехлаков Ю.П.

Методист кафедры АОИ _____ Коновалова Н.В.

Согласована на портале № 5570

1. ЦЕЛИ И ЗАДАЧИ ДИСЦИПЛИНЫ

Основная цель курса – развитие теоретических представлений и практических навыков работы с информацией, хранящейся или обрабатываемой в вычислительных системах, способам представления данных и их обработки с помощью современных информационных технологий. А так же, формирование у студентов объектно-ориентированного мышления, обучение объектно-ориентированному (ОО) подходу к анализу предметной области и использованию объектно-ориентированной методологии программирования при разработке программных продуктов.

Для достижения указанной цели в процессе изучения дисциплины решаются следующие **задачи:**

- формирование у студента знаний основных понятий, концепции, принципов и теорий, связанные с информатикой, понятия количества информации, типов систем счисления, структуры операционных систем, устройства файловых систем, основ архитектуры компьютера, основных принципов структурного программирования;
- получение студентами навыков осуществления операций преобразования и математических операций над данными, представленными в разных системах счисления, представления алгоритмов, программирования на языке высокого уровня;
- обучение студентов владению языками структурного программирования, математическим аппаратом систем счисления, навыками использования прикладных программ, навыками разработки и отладки программ на алгоритмических языках программирования;
- изучение техники объектно-ориентированного анализа;
- изучение приемов объектно-ориентированного программирования.

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП

Дисциплина **«Информатика и программирование» (Б1.Б.13)** относится к блоку дисциплин Б1.Б базовой части ОПОП.

Для успешного изучения дисциплины «Информатика и программирование» необходимы знания по дисциплинам «Алгебра и геометрия» (Б1.Б.11), «Математический анализ» (Б1.Б.10), «Дискретная математика» (Б1.В.ОД.2) и «Основы алгоритмизации» (Б1.Б.12), изучаемые студентами параллельно.

Последующими дисциплинами, в которых используются знания, полученные при изучении дисциплины «Информатика и программирование», являются «Математическая логика и теория алгоритмов» (Б1.В.ОД.4), «Теория вероятностей и математическая статистика» (Б1.Б.13), «Теория автоматов и формальных языков» (Б1.В.ОД.13), «Вычислительная математика» (Б1.В.ОД.3), «Операционные системы и сети» (Б1.Б.15), «Методы и технологии программирования» (Б1.Б.17), «Организация баз данных» (Б1.Б.20), « Функциональное и логическое программирование» (Б1.В.ОД.1), «Тестирование программного обеспечения» (Б1.В.ОД.12), «Проектирование и архитектура программных систем» (Б1.В.ОД.9), «Компьютерная графика» (Б1.В.ОД.5), «Имитационное моделирование» (Б1.В.ОД.15), «Объектно-ориентиро-ванное программирование » (Б1.В.ДВ.4), «Графическое трехмерное программирование» (Б1.В.ДВ.5).

3. ТРЕБОВАНИЯ К РЕЗУЛЬТАТАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Процесс изучения дисциплины направлен на **формирование следующих компетенций**:

общепрофессиональных:

1) владение основными концепциями, принципами, теориями и фактами, связанными с информатикой (**ОПК-1**);

профессиональных в производственно-технологической деятельности:

2) готовность применять основные методы и инструменты разработки программного обеспечения (**ПК-1**).

В результате изучения дисциплины **студент должен:**

знать:

- основные факты, концепции, принципы и теории, связанные с информатикой;
- понятие количества информации;
- типы систем счисления;
- структуру операционных систем;
- устройство файловых систем;
- основы архитектуры компьютера;
- основные принципы структурного программирования;
- синтаксис языка программирования Си;
- методы обработки и способы реализации основных структур данных в объектно-ориентированных программных средах;

уметь:

- работать с современными системами программирования;

Согласована на портале № 5570

- осуществлять операции преобразования и математические операции над данными, представленными в разных системах счисления;
- представлять алгоритмы на языке программирования Си;
- разрабатывать объектно-ориентированные программы в современных инструментальных средах;

владеть:

- языками процедурного программирования
- навыками разработки и отладки программ на алгоритмических языках программирования.
- навыками использования прикладных программ;
- техникой объектно-ориентированного анализа и проектирования информационных систем;
- практическими приемами объектно-ориентрованного программирования;
- навыками работы в средах объектно-ориентированного программирования (составление, отладка и тестирование программ).

4. ОБЪЕМ ДИСЦИПЛИНЫ И ВИДЫ УЧЕБНОЙ РАБОТЫ

Вид учебной работы	Всего, ч	Семестр I	Семестр II	Семестр III
Аудиторные занятия (всего), в том числе:	**186**	**54**	**60**	**72**
Лекции	80	18	26	36
Практические занятия (ПЗ)	18	-	18	-
Лабораторные работы (ЛР)	88	36	16	36
Самостоятельная работа (всего), в том числе:	**210**	**90**	**84**	**36**
Изучение тем курса, вынесенных на самост. изучение	12	-	-	12
Подготовка к контрольным работам	43	20	20	3
Подготовка к практическим занятиям	16	-	16	-
Подготовка к лабораторным работам	48	22	8	18
Выполнение домашних заданий	40	16	24	-
Подготовка к коллоквиуму	16	16	-	-
Подготовка к тестовому опросу	35	16	16	3
Вид промежуточной аттестации (экзамен)	**108**	**36**	**36**	**36**
Общая трудоемкость, ч	504	180	180	144
Зачетные Единицы Трудоемкости	15	5	5	4

5. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

5.1. Разделы дисциплин и виды занятий

Наименование раздела дисциплины	Лекции	Лабораторные работы	Практические занятия	Самостоятельная работа студента	Всего часов (без экзамена)	ОПК, ПК
1 семестр						
1. Введение в информатику	2	-	-	13	15	ОПК-1
2. Структурное программирование	2	-	-	15	17	ОПК-1
3. Программирование на языке Си	14	36	-	62	112	ПК-1
Итого по 1 семестру	**18**	**36**	**-**	**90**	**144**	
2 семестр						
4. Сортировки	10	16	8	38	66	ПК-1, ОПК-1
5. Поиск	8	-	4	22	34	
6. BST-деревья	8	-	6	24	38	
Итого по 2 семестру	**26**	**16**	**18**	**84**	**144**	
3 семестр						
7. Введение в объектно-ориентированный анализ и проектирование	6	12	-	11	29	ПК-1, ОПК-1
8. Основные понятия объектно-ориентированного программирования	12	12	-	8	32	ПК-1
9. Применение библиотек и иерархий объектов при программировании	18	12	-	17	47	ПК-1
Итого по 3 семестру	**36**	**36**	**-**	**36**	**108**	
ВСЕГО	**80**	**106**	**16**	**158**	**360**	

Согласована на портале № 5570

5.2. Содержание разделов дисциплины (по лекциям)

Наименование разделов	Содержание разделов	Трудоемкость, ч	ОПК, ПК
	1 семестр		
1. История развития информатики	Объект и предмет курса. Цели и задачи. Содержание. Связь с другими дисциплинами. Информация в материальном мире. Данные. Основные виды обработки информации. Информация и управление, информационные процессы. Понятие информационных технологий. Системы счисления. Двоичная, восьмеричная, шестнадцатеричная системы счисления. Двоичная арифметика. Биты. Байты. Слова. Структура памяти.	2	ОПК-1
2. Структурное программирование	История развития программирования. Теорема о структурном программировании. Конструкции структурного программирования. Алгоритм, переменная, константа. Способы представления алгоритмов. Основные приемы алгоритмизации (алгоритмы суммы, произведения; нахождение минимального и максимального значения; поиск элемента с заданным значением).	2	ОПК-1
3. Программирование на языке Си	Алфавит языка. Лексемы языка. Правила построения выражений. Правила записи имен и констант. Операторы языка Си. Структура программы на языке Си. Простые типы данных. Правила преобразования типов. Явное Преобразование типов. Неявное преобразование типов. Следование. Проверка условия, конструкции *if, case*. Циклы *for, while, do while*. Правила создания сложных логических выражений. Примеры использования. Массивы в Си. Одномерные, двумерные массивы. Алгоритмы поиска на массивах. Динамические и статические массивы. Алгоритмы сортировки. Строки. Синтаксис описания функции в Си. Формальные и фактические параметры. Локальные и глобальные переменные. Возвращаемое значение. Прототип функции. Параметры по ссылке. Типы файлов в Си. Текстовые и бинарные файлы. Функции для работы с файлами. Прямой и последовательный доступ к файлам. Понятие связного списка. Однонаправленный список, двунаправленный список. Способы добавления и удаления элементов из списка. Функции управления консолью. ***WINDOWS***- приложения. Функция *WinMain*. Оконная функция. Взаимодействие программы с ОС. Диалоговые окна. Элементы управления.	14	ПК-1
	2 семестр		
4. Сортировки	Простые сортировки**.** Сортировка выбором, сортировка вставками, сортировка обменом. Анализ эффективности сортировок. Понятие устойчивости и естественности сортировок. Способы улучшения алгоритмов сортировок. Улучшенные сортировки на месте. Сортировка Шелла. Сортировка комбинированная. Сортировка пирамидальная. Сортировка Хоара. Анализ эффективности сортировок. Сортировки слиянием. – Нисходящая и восходящая сортировки. Способы слияния. Прямое слияние. Абстрактное обменное слияние. Поразрядные сортировки. LSD и MSD сортировки. Двоичная быстрая сортировка. Средства языка для программирования поразрядной быстрой сортировки.	10	ОПК-1, ПК-1
5. Поиск	Общая задача поиска. Прямой поиск. Бинарный поиск. Интерполяционный поиск. Поиск подстроки в строке. Прямой поиск. Алгоритм Кнута. Алгоритм Боуера–Мура.	8	ОПК-1, ПК-1

Согласована на портале № 5570

	Очереди по приоритетам. Основные операции очереди. Неосновные операции над очередью. Способы представления очереди и их анализ. Биномиальная очередь. Сортирующие деревья степени 2. Алгоритмы построения биномиальной очереди. Добавление нового элемента. Удаление максимального элемента. Объединение двух очередей.		
6. BST-деревья	BST – деревья. Основные операции над деревьями – вставка элемента, поиск элемента, удаление элемента. Операции ротации. Понятия сбалансированности. Обходы дерева. AVL-деревья.	8	ОПК-1, ПК-1
3 семестр			
7. Введение в объектно-ориентированный анализ и проектирование	Программирование как фундаментальный метод реализации моделей на компьютере. Сложность программы. Возникновение объектно–ориентированного программирования. Основные понятия объектно-ориентированного анализа.	6	ПК-1, ОПК-1
8. Основные понятия объектно-ориентированного программирования.	Объектно-ориентированная методология программирования. Объектно-ориентированные языки. Понятия объекта, класса, метода, сообщения. Отношения простого и множественного наследования, виртуальные базовые классы. Абстракция данных, наследование и полиморфизм. Абстрактные классы. Полиморфизм параметрический и динамический. Совместимость типов в объектно-ориентированном программировании. Перегрузка и переопределение методов. Методы реализации различных конструкций объектно-ориентированного программирования.	12	ПК-1
9. Применение библиотек и иерархий объектов при программировании	Потоки. Установка потока. Чтение и запись потока. Вывод в поток. Ввод из потока. Удаление потока. Использование объектов с потоком. Механизм потоков. Процедуры обмена информации в потоках. Шаблоны. Библиотека стандартных шаблонов. Основные концепции STL. Последовательные и ассоциативные контейнеры. Итераторы STL. Общие свойства контейнеров. Использование последовательных контейнеров. Адаптеры контейнеров. Алгоритмы. Инструментальные средства объектно-ориентированного программирования.	18	ПК-1
Итого		**80**	

5.3. Разделы дисциплины и междисциплинарные связи с обеспечивающими (предыдущими) дисциплинами и обеспечиваемыми (последующими) дисциплинами

Наименование обеспечивающих (предыдущих) дисциплин и обеспечиваемых (последующих) дисциплин	Разделы данной дисциплины, которые необходимы для изучения обеспечиваемых (последующих) дисциплин								
	1	2	3	4	5	6	7	8	9
Дисциплины, изучаемые параллельно									
«Алгебра и геометрия» (Б1.Б.11)			+						
«Математический анализ» (Б1.Б.10)			+	+	+	+			
«Дискретная математика» (Б1.В.ОД.2)			+	+		+			
«Основы алгоритмизации» (Б1.Б.12)	+	+	+	+	+	+	+	+	+
Последующие дисциплины									
Математическая логика и теория алгоритмов (Б1.В.ОД.4)		+	+		+				
Теория вероятностей и математическая статистика (Б1.Б.13)	+	+	+	+					
«Теория автоматов и формальных языков» (Б1.В.ОД.13)		+	+				+	+	+
«Вычислительная математика» (Б1.В.ОД.3)		+	+	+	+				
«Операционные системы и сети» (Б1.Б.15)	+	+	+						
«Методы и технологии программирования» (Б1.Б.17)	+	+					+		
«Организация баз данных» (Б1.Б.20)	+	+			+		+		

Согласована на портале № 5570

Функциональное и логическое программирование (Б1.В.ОД.1)	+	+				+	+		
«Тестирование программного обеспечения» (Б1.В.ОД.12)	+	+	+	+	+	+	+		
«Проектирование и архитектура программных систем» (Б1.В.ОД.9)	+	+	+				+	+	
«Компьютерная графика» (Б1.В.ОД.5)	+	+	+				+		
«Имитационное моделирование» (Б1.В.ОД.15)	+	+	+		+				
«Объектно-ориентир. программирование» (Б1.В.ДВ.4)	+						+	+	+
«Графическое трехмерное программирование» (Б1.В.ДВ.5)	+	+	+	+	+	+	+	+	+

5.4. Соответствие компетенций, формируемых при изучении дисциплины, и видов занятий

Перечень компетенций	Л	ЛР	ПР	СРС	Формы контроля
ОПК-1	+	+	–	+	Тест, контрольная работа, отчет по ЛР, коллоквиум защита домашних заданий, защита командного задания, экзамен
ПК-1	+	+	+	+	Тест, контрольная работа, отчет по ЛР, коллоквиум, защита домашних заданий, экзамен

Л – лекция; ЛР – лабораторные работы; СРС – самостоятельная работа; ПЗ – практические занятия

6. МЕТОДЫ И ФОРМЫ ОРГАНИЗАЦИИ ОБУЧЕНИЯ

Методы организации обучения	Формы организации обучения, ч			
	Лекции	ЛР	ПР	СРС
1. Работа в команде	–	+	+	+
2. Мозговой штурм	–	+	+	+
3. Тестирование с использованием оценочной группы из числа студентов с последующим обсуждением	+	+		+
4. Поисковый метод	–	–	+	–

7. ЛАБОРАТОРНЫЕ РАБОТЫ

Раздел дисциплины	Тематика лабораторных работ	Трудо-емкость, ч	ОПК, ПК
	1 семестр		
3	Ввод-вывод данных. Проверка условий. Геометрия на плоскости	4	ПК-1
	Вычисление суммы бесконечного ряда с заданной точностью	4	
	Статические массивы. Алгоритмы поиска.	4	
	Динамические матрицы. Функции.	4	
	Строки	4	
	Текстовые файлы	4	
	Двоичные файлы	4	
	Односвязные динамические списки.	4	
	Управление консолью	4	
Итого по 1-му семестру		**36**	
	2 семестр		
4	Простые сортировки на месте	4	ОПК-1, ПК-1
	Улучшенные сортировки на месте	6	
	Сортировка слиянием	6	
Итого по 2-му семестру		**16**	
	3 семестр		
7	Знакомство с языком программирования Java. Простые программы.	4	ОПК-1, ПК-1
	Управляющие конструкции.	4	
	Массивы и строки.	4	
8	Создание класса. Конструкторы.	4	ПК-1
	Внутренние классы.	4	

Согласована на портале № 5570

	Абстрактные классы и интерфейсы. Наследование и полиморфизм. Создание класса-наследника	4	
9	Создание класса для работы с файлами.	4	ПК-1
	Обработка исключительных ситуаций.	4	
	Разработка и создание программной мини-системы	4	
Итого по 3-му семестру		**36**	
Всего		**106**	

8. ПРАКТИЧЕСКИЕ ЗАНЯТИЯ

Раздел дисциплины	Тематика практических занятий	Трудо-емкость, ч	ОПК, ПК
4	Поразрядные сортировки	8	ОПК-1, ПК-1
5	Бинарный и интерполяционный поиск	2	
	Поиск подстроки в строке	2	
6	Построение *BST*-деревьев	3	
	Операции на *BST*-деревьях	3	
Всего		**18**	

9. САМОСТОЯТЕЛЬНАЯ РАБОТА

Виды самостоятельной работы	Трудоемкость, ч										ОПК, ПК	Контроль выполнения работы
	По разделам дисциплины									Всего по виду СРС		
	1 семестр			2 семестр			3 семестр					
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**			
1. Изучение тем, вынесенных на самостоятельное изучение										**12**	ОПК-1, ПК-1	Тестовый опрос, контрольная работа
Этап анализа проблемы							1		1	2		
Управление масштабом проекта.									2	2		
Требования к ПО							1		2	3		
Организация требований							1		2	3		
Классы-утилиты									2	2		
2. Подготовка к КР:										**43**	ОПК-1, ПК-1	Контрольная работа
№ 3. Синтаксис языка Си			4							4		
№ 4. Циклы, проверка условий в языке Си			4							4		
№ 5. Работа с массивами в языке Си			4							4		
№ 6. Работа с матрицами в языке Си			4							4		
№ 7. Работа со строками в языке Си			4							4		
№ 8. Эффективность алгоритмов				3						3		
№ 9. Сортировки №1				2						2		
№10. Сортировки №2				3						3		
№ 11. Слияние				3						3		
№ 12. Поразрядные сортировки				3						3		
№ 13. Поиск подстроки					2					2		
№ 14. Бинарные деревья № 1						2				2		
№15. Бинарные деревья № 2						2				2		
№ 16. Объектно-ориентированная методология программирования							1			1		
№ 17. Разработка простого класса								1		1		
№ 18. Иерархия классов									1	1		
Итого в 1-м семестре			**20**							**20**	ОПК-1, ПК-1	
Итого в 2-м семестре				**14**	**2**	**4**				**20**		
Итого в 3-м семестре							**1**	**1**	**1**	**3**		
3. Подготовка к ПЗ										**16**		
Поразрядные сортировки				2						2		
Бинарный и интерполяционный поиск					3					3		
Поиск подстроки в строке					4					4		
Построение *BST*-деревьев						3				3		
Операции на *BST*-деревьях						4				4		

Согласована на портале № 5570

4. Подготовка к ЛР										**48**	ОПК-1, ПК-1	Защита отчета по ЛР
Ввод-вывод данных. Проверка условий. Геометрия на плоскости			1							1		
Вычисление суммы бесконечного ряда с заданной точностью.			2							2		
Статические массивы. Алгоритмы поиска			2							2		
Динамические матрицы. Функции.			2							2		
Строки			3							3		
Текстовые файлы			4							4		
Двоичные файлы			2							2		
Односвязные динамические списки.			3							3		
Управление консолью			3							3		
Простые сортировки на месте				2						2		
Улучшенные сортировки на месте				3						3		
Сортировка слиянием				3						3		
Знакомство с языком программирования Java. Простые программы							2			2		
Управляющие конструкции							2			2		
Массивы и строки							2			2		
Создание класса. Конструкторы								2		2		
Внутренние классы								2		2		
Абстрактные классы и интерфейсы. Наследование и полиморфизм. Создание класса-наследника								2		2		
Создание класса для работы с файлами									2	2		
Обработка исключительных ситуаций									2	2		
Разработка и создание программной мини-системы									2	2		
Итого в 1-м семестре			**22**							**22**		
Итого в 2-м семестре				**8**						**8**		
Итого в 3-м семестре							**6**	**6**	**6**	**18**		
5. Выполнение домашних заданий										**40**	ОПК-1, ПК-1	Отчеты по домашним заданиям
Системы счисления	4									4		
Алгоритмизация		4								4		
Циклические процессы			2							2		
Файлы			2							2		
Двусвязные списки			4							4		
Проверка эффективности алгоритмов				5						5		
Сортировка подсчетом				5						5		
Очереди по приоритетам					7					7		
AVL-деревья						7				7		
Итого в 1-м семестре	**4**	**4**	**8**							**16**		
Итого в 2-м семестре				**10**	**7**	**7**				**24**		
6. Подготовка к коллоквиуму										**16**	ОПК-1, ПК-1	Коллоквиум
Проработка теор. материала разд. 1-3	3	5	5							13		
Обсуждение результатов, работа над ошибками	1	1	1							3		
7. Подготовка к тестовому опросу (проработка лекц. материала)	5	5	6	4	6	6	1	1	1	**35**		Тестирование
Итого в 1-м семестре	***5***	***5***	***6***							***16***		
Итого в 2-м семестре				***4***	***6***	***6***				***16***		
Итого в 3-м семестре							***1***	***1***	***1***	***3***		
Всего по 1-му семестру (1-3 разделы)	**13**	**15**	**62**	**-**	**-**	**-**	**-**	**-**	**-**	**90**		
Всего по 2-му семестру (4-6 разделы)	**-**	**-**	**-**	**38**	**22**	**24**	**-**	**-**	**-**	**84**		
Всего по 3-му семестру (7-9 разделы)	**-**	**-**	**-**	**-**	**-**	**-**	**11**	**8**	**17**	**36**		
ВСЕГО по разделу дисциплины	**13**	**15**	**62**	**38**	**22**	**24**	**11**	**8**	**17**	**210**		
Подготовка к экзамену											108	Экзамен

Согласована на портале № 5570

11. РЕЙТИНГОВАЯ СИСТЕМА ДЛЯ ОЦЕНКИ УСПЕВАЕМОСТИ СТУДЕНТОВ

11.1. Балльные оценки для элементов контроля Экзамен – 1 семестр

Элементы учебной деятельности	Максимальный балл на 1-ю КТ с начала семестра	Максимальный балл за период между 1КТ и 2КТ	Максимальный балл за период между 2КТ и на конец семестра	Всего за семестр
Защита лабораторных работ	7	7	7	**21**
Выполнение домашних заданий	5	5	5	**15**
Тесты, контрольные работы	8	8	8	**24**
Коллоквиум	–	–	10	**10**
Итого максимум за период:	**20**	**20**	**30**	**70**
Сдача экзамена (максимум)				30
Нарастающим итогом	**20**	**40**	**70**	**100**

Экзамен состоит из трех вопросов по 10 баллов каждый.

Экзамен – 2 семестр

Элементы учебной деятельности	Максимальный балл на 1-ю КТ с начала семестра	Максимальный балл за период между 1КТ и 2КТ	Максимальный балл за период между 2КТ и на конец семестра	Всего за семестр
Защита лабораторных работ	11	11	9	**31**
Домашние задания	7	7	–	**14**
Практические занятия	–	5	5	**10**
Тесты, контрольные работы	5	5	5	**15**
Итого максимум за период:	**23**	**28**	**19**	**70**
Сдача экзамена (максимум)				30
Нарастающим итогом	**23**	**51**	**70**	**100**

Экзамен состоит из трех вопросов по 10 баллов каждый.

Экзамен – 3 семестр

Элементы учебной деятельности	Максимальный балл на 1-ю КТ с начала семестра	Максимальный балл за период между 1КТ и 2КТ	Максимальный балл за период между 2КТ и на конец семестра	Всего за семестр
Защита лабораторных работ	15	15	15	**45**
Тесты, контрольные работы	5	10	10	**25**
Итого максимум за период:	**20**	**25**	**25**	**70**
Сдача экзамена (максимум)				**30**
Нарастающим итогом	**20**	**45**	**70**	**100**

Экзамен состоит из трех вопросов по 10 баллов каждый.

11.2. Пересчет баллов в оценки за контрольные точки

Баллы на дату контрольной точки	Оценка
≥ 90 % от максимальной суммы баллов на дату КТ	5
От 70% до 89% от максимальной суммы баллов на дату КТ	4
От 60% до 69% от максимальной суммы баллов на дату КТ	3
< 60 % от максимальной суммы баллов на дату КТ	2

11.3. Пересчет суммы баллов в традиционную и международную оценку

Оценка (ГОС)	Итоговая сумма баллов (учитывает успешно сданный экзамен)	Оценка (ECTS)
5 (отлично) (зачтено)	**90 – 100**	A (отлично)
4 (хорошо) (зачтено)	**85 – 89**	B (очень хорошо)
	75 – 84	C (хорошо)
	70 – 74	D (удовлетворительно)
3 (удовлетворительно) (зачтено)	**65 – 69**	
	60 – 64	E (посредственно)
2 (неудовлетворительно) (не зачтено)	**Ниже 60 баллов**	F (неудовлетворительно)

Согласована на портале № 5570

12. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

12.1. Основная литература

1. Макарова Н.В. Информатика : учебник для вузов / Н.В. Макарова, В.Б. Волков. - СПб. : ПИТЕР, 2012. - 576 с. В библиотеке ТУСУРа: 51 экз.

3. Головин И. Г. Языки и методы программирования : учебник для вузов / И. Г. Головин, И. А. Волкова. - М. : Академия, 2012. - 304 с. В библиотеке ТУСУРа: 30 экз.

12.2. Дополнительная литература

1. Вирт Н. Алгоритмы и структуры данных : Пер. с англ. / Никлаус Вирт. - 2-е изд., испр. - СПб. : Невский Диалект, 2001. - 352 с. В библиотеке ТУСУРа: 1 экз.

2. Каширин И. Ю. От С к С++ : Учебное пособие для вузов / И. Ю. Каширин, В. С. Новичков. - М. : Горячая линия-Телеком, 2005. - 334[2] с. В библиотеке ТУСУРа: 49 экз.

3. Степанов А. Н. Информатика: Учебник для вузов / А. Н. Степанов. – 5-е изд. – СПб.: Питер, 2007. – 764 с. В библиотеке ТУСУРа: 30 экз.

4. Павловская Т. А. С/С++. Программирование на языке высокого уровня : учебник для вузов / Т. А. Павловская. - СПб. : Питер, 2007. - 464 с. В библиотеке ТУСУРа: 47 экз.

5. Павловская Т.А. С++. Объектно-ориентированное программирование. Практикум: учеб. пособие / Т.А. Павловская, Ю.А. Щупак. - СПб. : Питер, 2005. - 464 с. В библиотеке ТУСУРа**:** 15 экз.

12.3. Учебно-методические пособия и требуемое программное обеспечение

Для обеспечения дисциплины используются следующие УМП:

1. Пермякова Н.В. Информатика и программирование: метод. указания к лабораторным работам и практическим занятиям для студентов направления «Программная инженерия». – Томск: ТУСУР, 2016. – 113 с. [Электронный ресурс]: сайт кафедры АОИ. – URL: http://aoi.tusur.ru/upload/methodical_materials/MU_LR_PR_Informatika_i_programmirovanie_09_03_04_file__678_697.pdf.

2. Пермякова Н.В. Информатика и программирование: метод. указания по самостоятельной работе для студентов направления «Программная инженерия» - Томск – 2016. – 8 с. [Электронный ресурс]: сайт кафедры АОИ. – URL: http://aoi.tusur.ru/upload/methodical_materials/MU_SAM_Informatika_i_programmirovanie_09_03_04_file__679_5521.pdf.

Для организации самостоятельной работы студентов требуется свободный доступ в компьютерные классы с наличием программных систем

- Microsoft PowerPoint – для подготовки презентаций;
- Microsoft Word – для подготовки отчетов по работам;
- Интегрированная среда DEV-CPP – для реализации алгоритмов.

12.4. Необходимые базы данных, информационно-справочные и поисковые системы

Образовательный портал университета (http://portal.tusur.ru, http://lib.tusur.ru); электронные информационно-справочные ресурсы вычислительных залов кафедры АОИ

13. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Аудитории с мультимедийным оборудованием для проведения лекционных занятий. Компьютерные классы для практических и лабораторных занятий. Доступ в Интернет из компьютерных классов.

Согласована на портале № 5570

Приложение

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

«ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ УПРАВЛЕНИЯ
И РАДИОЭЛЕКТРОНИКИ» (ТУСУР)

Кафедра автоматизации обработки информации (АОИ)

УТВЕРЖДАЮ

Заведующий кафедрой АОИ

_____Ю.П. Ехлаков

«____»_____2016 г.

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ДЛЯ ПРОВЕДЕНИЯ ТЕКУЩЕЙ И ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ПО УЧЕБНОЙ ДИСЦИПЛИНЕ «ИНФОРМАТИКА И ПРОГРАММИРОВАНИЕ»

для направления подготовки бакалавра 09.03.04 «Программная инженерия»

(учебный план набора 2015 г.)

Составитель:

ст. преподаватель каф. АОИ

_____ Н.В. Пермякова

Томск 2016

Согласована на портале № 5570

1. ОСНОВНЫЕ ПОЛОЖЕНИЯ

Фонд оценочных средств (ФОС) является приложением к рабочей программе дисциплины и представляет собой совокупность контрольно-измерительных материалов и методов их использования, предназначенных для измерения уровня достижения студентом установленных результатов обучения. ФОС используется при проведении текущего контроля успеваемости и промежуточной аттестации студентов.

При описании фонда оценочных средств по учебной дисциплине используется нижеприведенная терминология.

Компетенция – комплекс взаимосвязанных аспектов профессиональной деятельности, складывающихся из знаний, умений, навыков и/или опыта, объединенных с потенциальной способностью и готовностью студента (выпускника) справляться с решением задач, обусловленных видами и объектами профессиональной деятельности.

Этапы освоения компетенции – логически увязанные части жизненного цикла освоения компетенции

Оценочные средства – совокупность контрольных/контрольно-измерительных и методических материалов, необходимых для определения степени сформированности компетенций по конкретной дисциплине.

Контрольные материалы оценочного средства – конкретные задания, позволяющие определить результативность учебно-познавательной и проектной деятельности студента.

Показатели оценивания компетенций – сформулированные на содержательном уровне требования к освоению компетенции, распределенные по этапам ее формирования и обусловленные видами и объектами профессиональной деятельности, обобщенными трудовыми функциями профессиональных стандартов,

Критерии оценивания компетенций – правило дифференциации показателя уровня освоения компетенции

Таблица 1 – Обобщенная модель формирования содержания показателей оценивания компетенции

Этапы	Обобщенные показатели		
	Теоретические основы	**Методологические основы**	**Инструментальные основы**
Знать	Обладает знаниями теоретического материала, в том числе по содержанию терминов, понятий, взаимосвязей между ними	Обладает знаниями по технологиям решения профессиональных задач	Обладает знаниями в области инструментальных средств (программной и/или программно-аппаратной реализации профессиональных задач)
Уметь	Обладает умениями по использованию теоретического материала для решения профессиональных задач	Обладает умениями адаптации технологий решения профессиональных задач на контрольных (модельных) заданиях	Обладает умениями применения инструментальных средств для решения профессиональных задач на контрольных (модельных) заданиях
Владеть	Обладает навыками и/или опытом преобразования (трансформации) теоретического материала в рамках получения нового знания	Обладает навыками и/или опытом адаптации технологий решения профессиональных задач для реальных данных / ситуаций / условий	Обладает навыками и/или опытом применения инструментальных средств для решения профессиональных задач для реальных данных / ситуаций / условий

Таблица 2 – Шкала оценивания уровня освоения компетенции

Уровни освоения компетенции	**Экзаменационная оценка / дифференцированный зачет**	**Зачет**
Неудовлетворительный	неудовлетворительно	не зачтено
Пороговый	удовлетворительно	зачтено
Базовый	хорошо	зачтено
Высокий	отлично	зачтено

Согласована на портале № 5570

2. КОМПЕТЕНЦИИ, ЭТАПЫ И ОЦЕНОЧНЫЕ СРЕДСТВА

Перечень компетенций с указанием этапов их формирования в процессе освоения дисциплины приведен в таблице 3.

Таблица 3 – Перечень закрепленных за дисциплиной компетенций

Код	Формулировка компетенции	Этапы формирования компетенции
ОПК-1	Владение основными концепциями, принципами, теориями и фактами, связанными с информатикой	Знать, уметь, владеть
ПК-1	Готовность применять основные методы и инструменты разработки программного обеспечения	

Для оценки качества степени освоения компетенций по дисциплине используются следующие оценочные средства.

Промежуточная аттестация

Экзамен — проверка знаний студентов, выявление навыков и умений применения знаний при решении профессиональных задач. Экзамен проводится в письменной и устной форме. Целью экзамена является выявление индивидуальных достижений студента в освоении:

основных понятий информатики;

навыков алгоритмизации;

принципов структурного и объектно-ориентированного подходов к программированию;

синтаксиса изучаемых языков программирования.

Текущая аттестация (текущий контроль освоения компетенций)

Тестирование – учебная технология, позволяющая измерять знания, умения и навыки студентов, состоящая из тестовых заданий и формализованных процедур проведения, обработки и анализа результатов.

Контрольная работа – средство промежуточного контроля остаточных знаний и умений, обычно состоящее из нескольких вопросов или заданий, которые студент должен решить, выполнить.

Лабораторная работа – оценка способности студента применить полученные ранее знания для проведения анализа, опыта, эксперимента и выполнения последующих расчетов, а также составления выводов.

Коллоквиум – способ промежуточной проверки знаний, умений, навыков студента в середине семестра по пройденным темам изучаемого предмета.

Командное задание – проблемное задание, в котором группе студентов предлагают решить конкретную задачу общими усилиями. Студенты самостоятельно распределяют роли в команде, ищут варианты решения задачи, программно реализуют поставленную задачу и защищают результаты своей работы.

Домашняя работа – вид индивидуального задания, выполняемого студентом в часы, запланированные для самостоятельной работы.

3. ПОКАЗАТЕЛИ И КРИТЕРИИ ОЦЕНИВАНИЯ КОМПЕТЕНЦИЙ

3.1. Компетенция ОПК-1

ОПК-1: владение основными концепциями, принципами, теориями и фактами, связанными с информатикой.

Этапы формирования компетенции, показатели и используемые оценочные средства представлены в таблице 4.

Критерии и уровни оценивания компетенции на каждом этапе приведены в таблице 5.

Согласована на портале № 5570

Таблица 4 – Этапы, показатели и используемые оценочные средства формирования компетенции

Состав	Показатели оценивания компетенций по этапам		
	Знать	**Уметь**	**Владеть**
Описание показателей	Иметь представление об основных фактах, концепциях, принципах, связанных с информатикой - системах счисления, структуре операционных систем, устройстве файловых систем, основах архитектуры компьютера, понятии количества информации.	осуществлять операции преобразования и математические операции над данными, представленными в различных системах счисления.	навыками использования прикладных программ.
Виды занятий	Лекции, самостоятельная работа	Самостоятельная работа	Лабораторные работы, самостоятельная работа
Используемые оценочные средства	Контрольные работы, экзамен, коллоквиум	Защита домашнего задания	Защита домашнего задания

Таблица 5 – Критерии и уровни оценивания компетенции

Уровни оценивания	Критерии оценивания компетенций по этапам		
	Знать	**Уметь**	**Владеть**
Отлично (высокий уровень)	Способен перечислить основные термины и понятия и самостоятельно раскрыть содержание термина или понятия во взаимосвязи с иными элементами терминологии	Способен корректно обрабатывать и анализировать материалы требуемые для выполнения заданий домашней работы, лабораторных работ и работы в команде из информационных и учебно-методических научно – образовательных ресурсов	Способен свободно использовать информационные, компьютерные и сетевые технологий для поиска информации из различных источников и баз данных
Хорошо (базовый уровень)	Способен перечислить основные термины и понятия и самостоятельно раскрыть содержание термина или понятия	Способен обрабатывать материалы, требуемые для выполнения заданий домашней работы, лабораторных работ и работы в команде из учебно-методических ресурсов	Способен использовать информационные, компьютерные и сетевые технологий для поиска информации из различных источников и баз данных, пользуясь инструктивными и справочными материалами
Удовлетворительно (пороговый уровень)	Способен перечислить основные термины и понятия и корректно определить значение термина или понятия через выбор из предложенного списка вариантов	Способен корректно обрабатывать материалы требуемые для выполнения заданий домашней работы, лабораторных работ и работы в команде из учебно-методических ресурсов, содержащих примеры выполнения подобных заданий	Способен использовать информационные, компьютерные и сетевые технологий для поиска информации из различных источников и баз данных, периодически обращаясь за помощью к преподавателю

3.2. Компетенция ПК-1

ПК-1: Готовность применять основные методы и инструменты разработки программного обеспечения**.**

Этапы формирования компетенции, показатели и используемые оценочные средства представлены в таблице 6.

Согласована на портале № 5570

Таблица 6 – Этапы, показатели и используемые оценочные средства формирования компетенции

Состав	Показатели оценивания компетенций по этапам		
	Знать	**Уметь**	**Владеть**
Описание показателей	основные принципы структурного программирования; синтаксис языка программирования Си; методы обработки и способы реализации основных структур данных в объектно-ориентированных программных средах.	представлять алгоритмы на языке программирования Си; разрабатывать объектно-ориентированные программы в современных инструментальных средах.	навыками разработки и отладки программ на алгоритмических языках программирования техникой объектно-ориентированного анализа и проектирования информационных систем; практическими приемами объектно-ориентрованного программирования; навыками работы в средах объектно-ориентированного программирования (составление, отладка и тестирование программ).
Виды занятий	Лекции, лабораторные работы, самостоятельная работа.	Лекции, лабораторные работы, самостоятельная работа.	Лекции, лабораторные работы, самостоятельная работа.
Используемые оценочные средства	Контрольная работа, защита отчета по ЛР, защита командного задания, защита домашнего задания, тестирование, экзамен, коллоквиум.	Контрольная работа, защита отчета по ЛР, защита командного задания, защита домашнего задания, тестирование, экзамен, коллоквиум.	защита отчета по ЛР, защита командного задания, защита домашнего задания, экзамен.

Критерии и уровни оценивания компетенции на каждом этапе приведены в таблице 7.

Таблица 7 – Показатели и критерии оценивания компетенции

Уровни оценивания	Критерии оценивания компетенций по этапам		
	Знать	**Уметь**	**Владеть**
Отлично (высокий уровень)	Знает синтаксис изучаемого языка программирования и может применять его для разработки программного кода без использования справочной литературы.	Способен самостоятельно разработать алгоритм решения поставленной задачи и реализовать его на изучаемом языке программирования;	Способен самостоятельно отладить (исправить синтаксические ошибки) и протестировать (исключить логические ошибки) разработанную программу.
Хорошо (базовый уровень)	Знает синтаксис изучаемого языка программирования и может применять его для разработки программного кода с использованием справочной литературы.	Способен самостоятельно разработать алгоритм решения поставленной задачи и реализовать его на изучаемом языке программирования, предварительно обсудив идею алгоритма с преподавателем;	Способен самостоятельно отладить программу (исправить синтаксические ошибки). Способен исправить логические ошибки программы, если преподаватель укажет на их наличие.
Удовлетворительно (пороговый уровень)	Знает синтаксис изучаемого языка программирования и может применять его для разработки программного кода с использованием справочной литературы.	Способен разработать алгоритм решения поставленной задачи по имеющемуся шаблону и реализовать его на изучаемом языке программирования	Способен самостоятельно отладить (исправить синтаксические ошибки).

Согласована на портале № 5570

4. КОНТРОЛЬНЫЕ МАТЕРИАЛЫ ОЦЕНОЧНЫХ СРЕДСТВ

4.1. Промежуточная аттестация

Промежуточная аттестация реализуется посредством проведения экзамена в 1, 2 и 3 семестре изучения дисциплины. Экзамен может быть проставлен по рейтингу, полученному студентом по результатам освоения компетенции в течение семестра, либо проведен в формате письменного и устного опроса. Экзамен выставляется при успешном выполнении всех текущих элементов контроля: выполнении лабораторных работ, индивидуальных и домашних заданий. Для проведения экзамена составляются билеты.

Список тем для проведения экзамена (1 семестр)

1. Конструкции структурного программирования (следование, проверка условия, цикл). Логика выполнения действий.
2. Системы кодирования (блок-диаграмма, диаграмма Насси-Шнайдермана, псевдокод)
3. Основные алгоритмы(алгоритмы суммы, произведения, поиска экстремальных значений)
4. Синтаксис и алфавит языка Си (правила написания программ на Си, формирование имен переменных, разделители языка Си).
5. Основные типы данных. Преобразование типов.
6. Оператор проверки условия if [else]. Синтаксис и логика работы.
7. Цикл for. Синтаксис и логика работы.
8. Циклы while и do while. Синтаксис и логика работы.
9. Множественный выбор switch. Синтаксис и логика работы
10. Производные типы данных – указатели и ссылки. Основные правила работы с указателями и ссылками.
11. Функция printf
12. Функция scanf
13. Массивы (способы описания массивов, способы инициализации элементов массива, вывод элементов массива на экран, обращение к элементу массива)
14. Сортировка элементов массива (алгоритмы обмена, выбора и вставки)
15. Матрицы(способы описания матриц, способы инициализации элементов матрицы, вывод элементов матрицы на экран, обращение к элементу матрицы).

Пример экзаменационного билета

Билет № 1

1. Представление алгоритмов в системе «псевдокод». Запишите алгоритм поиска минимального элемента массива.
2. Цикл for в языке Си. Синтаксис. Принцип работы. Запишите с помощью цикла for фрагмент программы, выводящий на экран значения 2,4,8,10,12.
3. Напишите программу, которая запрашивает с клавиатуры размерность массива, заполняет массив случайными значениями из интервала [-5,10] и находит сумму элементов, расположенных между минимальным и максимальным элементами. Если минимальных и максимальных элементов несколько, вычисляется сумма элементов, расположенных между первыми встреченными элементами.

Список тем для проведения экзамена (2 семестр)

1. Матрицы(способы описания матриц, способы инициализации элементов матрицы, вывод элементов матрицы на экран, обращение к элементу матрицы).
2. Обработка матриц
3. Функции в языке Си
4. Текстовые файлы.
5. Двоичные файлы.

Согласована на портале № 5570

6. Связные списки.
7. Программирование в *Win-API*

Пример экзаменационного билета

Билет 1

1. В текстовом файле записан массив целых чисел. Считать массив из файла. Найти сумму чисел. В конец исходного файла дописать строку: «Сумма чисел = [найденная сумма]».
2. Написать:

– функцию создания двоичного файла, содержащего матрицу целых чисел размерности *n*x*m*. Значения *n, m* задаются с клавиатуры. Элементы матрицы – целые случайные числа;
– функцию печати содержимого двоичного файла;
– функцию, выводящую на экран четные столбцы матрицы, не считывая при этом матрицу в память.

3. В текстовом файле хранится произвольное количество чисел. Считать данные из файла в однонаправленный динамический список, организованный по правилу очереди (первое число из файла должно оказаться в «голове», последнее в «хвосте» списка). Поменять местами первый и последний элементы списка.

4.2. Текущая аттестация (текущий контроль освоения компетенций)

4.2.1. Тестирование

Тестирование проводится в целях оперативного мониторинга качества усвоения теоретического и практического материала (таблица 8).

Таблица 8 – Шкала оценивания компетенций при тестировании

Шкала оценивания	Уровень освоения компетенции		
	Высокий уровень	Базовый уровень	Пороговый уровень
Удельный вес правильных ответов по темам дисциплины, связанным с соответствующей компетенцией, %	Более 90	70–90	50–70

Список проводимых тестов

1. *Системы счисления*
2. *Структурное программирование*
3. *Синтаксис и алфавит языка Си*
4. *Простые типы данных языка Си*
5. *Конструкции структурного программирования*
6. *Сложные типы данных*
7. *Функции*
8. *Файлы*
9. *Динамические списки*
10. *Программирование в WinAPI*

Пример тестового билета приведен в таблице 9.

Таблица 9 – Билет тестового опроса «Системы счисления»

Вариант 1	**ФИО** _____ **гр.**
Выберите числа, записанные в двоичной системе счисления: 1. 12010 2. 10 3. 1100 4. 765	**Переведите число из двоичной системы счисления в десятичную:** 111011_2
Представьте число 123_{10} в 16-ричной системе счисления.	

Согласована на портале № 5570

4.2.2. Контрольная работа

Контрольная работа это продукт самостоятельной работы (активности) студента по кругу вопросов, составляющих предмет изучения, ограниченных ранее определенной темой. Ответы на поставленные вопросы даются письменно. Контрольные работы проводятся в целях оперативного мониторинга качества усвоения теоретического и практического материала (таблица 10).

Таблица 10 – Шкала оценивания компетенций при выполнении контрольных работ

Шкала оценивания	**Уровень освоения компетенции**		
	Высокий уровень	**Базовый уровень**	**Пороговый уровень**
Удельный вес правильных ответов по темам дисциплины, связанным с соответствующей компетенцией, %	Более 90	75–90	30–75

Ниже перечислены темы контрольных работ, проводимых во время изучения дисциплины.

1. Представление данных в ЭВМ
2. Способы представления алгоритмов
3. Синтаксис языка Си
4. Циклы, проверка условий в языке Си
5. Работа с массивами в языке Си
6. Работа с матрицами в языке Си
7. Работа со строками в языке Си
8. Динамические списки

Пример билета контрольной работы приведен в таблице 11.

Таблица 11 – Пример билета контрольной работы «Циклы. Проверка условий в языке Си»

Вариант 1
Фамилия _____ гр____
 	
1. Используя цикл while, запишите фрагмент программы, который выводит на экран числа 2 5 8 11 14 17 20. Описание использованных переменных обязательно.
	2. Что будет выведено на экран при выполнении следующего фрагмента программы:
	int x = 7;
	int y = 9;
	int z = 0;
	if (x>y) {z = y*2;
	y = x*4; }
	else { z = x*2;
	x = y+x;}
	printf (“ %d %d %d”, x,y,z);
		3. Запишите фрагмент программы, решающей следующую задачу (используйте цикл for):
		
		Вывести на экран числа от 0 до 12 с шагом 0.25. Фрагмент обязательно должен содержать описания использованных переменных.

4.2.3. Лабораторная работа

Лабораторные работы проводятся в соответствии с методическими указаниями по лабораторным работам, содержащими цель, порядок выполнения, контрольные задания (вопросы), форму отчетности. При проведении текущей аттестации используются показатели и критерии оценивания, а также качественная шкала, представленные в табл. 12.

Таблица 12 – Шкала оценивания компетенций при выполнении лабораторных работ

Согласована на портале № 5570

Шкала оценивания	Уровень освоения компетенции		
	Высокий уровень	**Базовый уровень**	**Пороговый уровень**
Критерии оценивания	Студент выполнил лабораторную работу самостоятельно в положенный срок, отчет по лабораторной работе выполнен грамотно и соответствует требованиям, изложенным в методических указаниях. При защите лабораторной работы студент свободно ориентируется в теоретическом материале, умеет анализировать полученные результаты, отвечает на контрольные вопросы.	Студент выполнил лабораторную работу самостоятельно, возможно для выполнения работы понадобилось дополнительное время. Отчет по лабораторной работе соответствует требованиям, изложенным в методических указаниях. При защите лабораторной работы студент ориентируется в теоретическом материале с помощью справочной литературы, может выполнить частичный анализ полученных результатов.	При выполнении лабораторной работы студент использовал шаблон задания, разработанный не самостоятельно, для выполнения лабораторной работы понадобилось дополнительное время. Отчет по лабораторной работе соответствует требованиям, изложенным в методических указаниях. При защите лабораторной работы студент использует справочные материалы. Анализ полученных результатов может быть выполнен по заранее разработанному шаблону.

4.2.4. Коллоквиум

Коллоквиум проводится в конце первого семестра изучения дисциплины. Основная цель коллоквиума – оценка подготовленности студентов к последующей сдаче экзамена. Коллоквиум проводится в письменной форме, пример билета коллоквиума приведен ниже. После проведения письменного опроса и проверки выполненных работ проводится обсуждение полученных результатов и выполнение работы над ошибками. Шкала оценивания компетенций при проведении коллоквиума совпадает со шкалой оценивания компетенций при тестировании (табл. 8).

Пример билета для проведения коллоквиума

Билет № 1
1. На псевдокоде записан алгоритм:
ввод *n*
ввод *x[n]*
i = 1, k = 0
пока *i<=n* делать
    если *x[i]>0* то *k = k+1*
    i = i+1
конец цикла
вывод *k*
Запишите имена переменных-счетчиков _____
2. Выберите задачи, для решения которых требуется построить алгоритм, использующий ветвление:
а) нахождение гипотенузы по двум катетам
б) нахождение суммы элементов массива
в) нахождение произведения двух матриц
г) решение квадратного уравнения
3. Запишите на псевдокоде алгоритм проверки существования положительных элементов в массиве *X[n].*
4. Чему будет равна переменная *S* после выполнения алгоритма:
Шаг1. *S:=0*
Шаг2. ДЛЯ *i* ОТ 1 ДО 3
    Шаг2.1. *S:=S+i;*
КОНЕЦ ЦИКЛА
Шаг3. Печать *S*.
5. Может ли переменная на Си иметь имя *31_p20?*

Согласована на портале № 5570

6. Какой тип нужно использовать для описания переменной *n*, если ее значение используется в формуле:

$$S = \sum_{n=1}^{\infty} \frac{(-1)^n}{n!} \cdot (n+1)$$

7. Что будет выведено на экран при выполнении следующего фрагмента программы:

```
int x = 1;
int y = 2;
int z = 3;
if  (x>y) {z = y*2;
           y = x*4; }
else { z = x*2;
       x = y+x;}
printf (" %d    %d    %d", x,y,z);
```

8. Запишите фрагмент программы, решающей следующую задачу (используйте цикл for):

Вывести на экран следующие числа:

2 5 8 11 14 17 20 23

Фрагмент обязательно должен содержать описания использованных переменных.

9. Что будет выведено на экран при выполнении следующего фрагмента программы:

```
int k = 0;
while (k<10)
   k++;
printf("%d \n",k);
```

10. Чему будет равно значение переменной *f* после выполнения следующего фрагмента программы

```
...
int k = 11;
int f = 0;
switch(k){
case 1: f = 1; break;
case 2: f = 2; break;
case 3: f = 3; break;
case 4: f = 4; break;
case 5: f = 5; break;
default: f = 10;
}
```

...

11. Запишите фрагмент программы, который выполняет следующие действия:

1. Опишите z как указатель на float.
2. Выделите память под z
3. Запишите по адресу z значение 3.11

12.В программе переменные *j* и *i* описаны следующим образом: *char j; int i*

Запишите функцию *printf*, выводящую на экран значения этих переменных в строке:

“Значение i = <…> Значение j = <…>”

13.В программе переменная *j* описана следующим образом: *int j;*

Запишите функцию *scanf*, считывающую значение переменной *j* с клавиатуры

14. Что будет выведено на экран при выполнении следующей программы:

```
int main(int argc, char *argv[])
{    system("chcp 1251");
   int x[10] = {2,7,6,1,9,5,8,3,4,0};
   int k = x[0];
   for (int i=0;i<10;i++)
      if (x[i]>k) k = x[i];
  printf("%3d",k);
   printf("\n");
   system("PAUSE");
   return EXIT_SUCCESS;}
```

15. Назовите алгоритм сортировки, который фиксирует элементы массива, начиная с первого, и ищет на зафиксированное место минимальный элемент, среди элементов с номерами от 1 до n. Далее фиксируется второй элемент и ищется минимальный элемент среди оставшихся, найденный элемент выставляется на второе место и т.д..

16. Элементы матрицы задаются следующим фрагментом программы:

```
int x[3][3];
int n = 3;
for (int i=0;i<n;i++)
```

Согласована на портале № 5570

```
for(int j=0;j<n;j++)
  if(i==j) x[i][j] = i;
    else x[i][j] = -1;
```

Запишите получившуюся матрицу.

17. Опишите структуру, хранящую данные следующих типов:

I - вещественное число

M – символ

S – указатель на целое число

18. Чему равен размер переменной *p*, если в программе сделаны следующие описания:

```
union Example {
int k;
char z[5];
} p;
```

19. Результат какого типа возвращает функция *int f10 (float x, int y) {…}*

20. Что будет выведено на экран при выполнении следующей программы:

```
#include <cstdlib>
#include <iostream>

using namespace std;
int Function (int x, int y, int *k){
x+=13;
y+=10;
(*k)+=12;
return y;
}

int main(int argc, char *argv[])
{
int x = 1;
int y = 1;
int k = 1;
x = Function(x,y,&k);
printf("x = %d y = %d k = %d\n",x,y,k);
system("PAUSE");
return EXIT_SUCCESS;
}
```

4.2.5. Домашние задания

Домашние задания выполняются студентами самостоятельно, во время самостоятельной работы, предусмотренной учебным планом дисциплины. Темы домашних заданий и методические указания по их выполнению содержаться в методических указаниях по самостоятельной работе. Результатом выполнения домашней работы является корректно работающая компьютерная программа.

В таблице 13 приведена шкала оценивания компетенции при выполнении индивидуальных заданий.

Таблица 13 – Шкала оценивания компетенций при выполнении домашних заданий

Шкала оценивания	**Уровень освоения компетенции**		
	Высокий уровень	**Базовый уровень**	**Пороговый уровень**
Критерии оценивания	Студент выполнил домашнее задание в срок и в полном объеме. Защищаемая программа корректно работает на любых наборах данных. Студент свободно ориентируется в коде программы и может внести в код программы изменения по просьбе преподавателя.	Студент выполнил домашнее задание в полном объеме. Для выполнения домашней работы потребовалось дополнительное время. Защищаемая программа может некорректно работать на выборочных данных. При обнаружении таких ошибок студент может самостоятельно их исправить.	Студент выполнил задание частично или для выполнения задания потребовалось дополнительное время. Защищаемая программа может некорректно работать на выборочных данных. При обнаружении таких ошибок студент может их исправить, прибегнув к помощи справочной литературы.

Согласована на портале № 5570