МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

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

УТВЕРЖДАЮ
Проректор по УР
_____Сенченко П.В.
«22» _____02_____ 2023 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

ВВЕДЕНИЕ В ПРОГРАММИРОВАНИЕ

Уровень образования: **высшее образование - бакалавриат**
Направление подготовки / специальность: **09.03.02 Информационные системы и технологии**
Направленность (профиль) / специализация: **Компьютерные технологии в радиотехнике**
Форма обучения: **очная**
Факультет: **Радиотехнический факультет (РТФ)**
Кафедра: **Кафедра телевидения и управления (ТУ)**
Курс: **1**
Семестр: **1**
Учебный план набора 2023 года

Объем дисциплины и виды учебной деятельности

Виды учебной деятельности	1 семестр	Всего	Единицы
Лекционные занятия	36	36	часов
Практические занятия	36	36	часов
Самостоятельная работа	108	108	часов
Общая трудоемкость (включая промежуточную аттестацию)	180	180	часов
	5	5	з.е.

Формы промежуточной аттестация	Семестр
Зачет	1

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Сенченко П.В.
Должность: Проректор по УР
Дата подписания: 22.02.2023
Уникальный программный ключ:
a1119608-cdff-4455-b54e-5235117c185c

Томск

Согласована на портале № 72083

1. Общие положения

1.1. Цели дисциплины

1. Целью курса является обучение студентов основам проектирования и программирования в современных средах разработки ПО.

1.2. Задачи дисциплины

1. Получить знания и практические навыки в области проектирования и разработки программного обеспечение.
2. Получить представление о предпосылках возникновения парадигм программирования (процедурное, функциональное, объектно-ориентированное), а также о принципах проектирования программного обеспечения (ПО) используя различные инструменты и практики.
3. Получить представление о применении принципов ООП при разработке на языках C++ и Python.
4. Научиться разрабатывать программные модули на языках C\C++ и Python с применением как стандартных так и сторонних библиотек.

2. Место дисциплины в структуре ОПОП

Блок дисциплин: Б1. Дисциплины (модули).
Часть блока дисциплин: Часть, формируемая участниками образовательных отношений.
Модуль дисциплин: Модуль направленности (профиля) (major).
Индекс дисциплины: Б1.В.02.01.01.
Реализуется с применением электронного обучения, дистанционных образовательных технологий.

3. Перечень планируемых результатов обучения по дисциплине, соотнесенных с индикаторами достижения компетенций

Процесс изучения дисциплины направлен на формирование следующих компетенций в соответствии с ФГОС ВО и основной образовательной программой (таблица 3.1):

Таблица 3.1 – Компетенции и индикаторы их достижения

Компетенция	Индикаторы достижения компетенции	Планируемые результаты обучения по дисциплине
Универсальные компетенции		
-	-	-
Общепрофессиональные компетенции		
-	-	-
Профессиональные компетенции		

Согласована на портале № 72083

ПК-4. Способен выполнять математическое и компьютерное моделирование объектов и процессов по типовым методикам для решения профессиональных задач	ПК-4.1. Знает типовые методы математического моделирования, используемые в специализируемых прикладных программах для проектирования и разработки радиотехнических систем	Знает стандарты оформления и документирования программного кода, унифицированный язык UML, а также стандарты вуза ОС ТУСУР
	ПК-4.2. Умеет выполнять моделирование физических объектов и процессов с использованием специализированных прикладных программ	Умеет выполнять моделирование различных систем с использованием специализированных программ и компонентов
	ПК-4.3. Владеет типовыми методиками разработки радиоэлектронных средств и их составных частей, в том числе с использованием прикладных программ	Владеет навыками и методиками разработки радиоэлектронных средств и их составных частей, в том числе с использованием прикладных программ, включая проектную документацию, программную документацию

4. Объем дисциплины в зачетных единицах с указанием количества академических часов, выделенных на контактную работу обучающихся с преподавателем и на самостоятельную работу обучающихся

Общая трудоемкость дисциплины составляет 5 зачетных единиц, 180 академических часов.

Распределение трудоемкости дисциплины по видам учебной деятельности представлено в таблице 4.1.

Таблица 4.1 – Трудоемкость дисциплины по видам учебной деятельности

Виды учебной деятельности	Всего часов	Семестры
		1 семестр
Контактная аудиторная работа обучающихся с преподавателем, всего	72	72
Лекционные занятия	36	36
Практические занятия	36	36
Самостоятельная работа обучающихся, в т.ч. контактная внеаудиторная работа обучающихся с преподавателем, всего	108	108
Подготовка к зачету	70	70
Подготовка к тестированию	38	38
Общая трудоемкость (в часах)	180	180
Общая трудоемкость (в з.е.)	5	5

5. Структура и содержание дисциплины

5.1. Разделы (темы) дисциплины и виды учебной деятельности

Структура дисциплины по разделам (темам) и видам учебной деятельности приведена в таблице 5.1.

Таблица 5.1 – Разделы (темы) дисциплины и виды учебной деятельности

Названия разделов (тем) дисциплины	Лек. зан., ч	Прак. зан., ч	Сам. раб., ч	Всего часов (без экзамена)	Формируемые компетенции
1 семестр					

Согласована на портале № 72083

1 Введение в теорию программирования	1	-	6	7	ПК-4
2 Интерпретаторы и компиляторы: преобразование программного кода в машинный код.	1	-	6	7	ПК-4
3 Основные понятия программирования: переменные, типы данных, операторы, условные выражения, циклы.	2	4	6	12	ПК-4
4 Алгоритмы и структуры данных: массивы, списки, стеки, очереди.	2	4	6	12	ПК-4
5 Функции и процедуры: создание, вызов, передача параметров.	2	2	6	10	ПК-4
6 Работа с файлами: чтение и запись данных, обработка текстовых и бинарных файлов.	2	2	6	10	ПК-4
7 Ошибки и исключения: обработка исключительных ситуаций, отладка программы.	2	2	6	10	ПК-4
8 Работа с библиотеками и сторонними модулями: подключение, использование и документация.	2	2	6	10	ПК-4
9 Классы и структуры. Члены классов. Дружественность	2	-	6	8	ПК-4
10 Основы объектно-ориентированного программирования: классы, объекты, наследование, полиморфизм.	2	2	6	10	ПК-4
12 Введение в алгоритмическое мышление и решение задач: разработка алгоритмов, построение блок-схем.	2	4	6	12	ПК-4
13 Работа с отладчиком: пошаговое выполнение кода, отслеживание значений переменных.	2	-	6	8	ПК-4
15 Основные конструкции языка программирования Python	2	-	6	8	ПК-4
16 Основные конструкции языка программирования C++	2	2	6	10	ПК-4
18 Жизненный цикл программного обеспечения	2	2	6	10	ПК-4
21 Работа с базами данных: основы SQL, создание и управление таблицами, выполнение запросов.	2	2	6	10	ПК-4
22 Работа с графическим интерфейсом: основы разработки GUI-приложений, создание элементов управления, обработка событий.	2	2	4	8	ПК-4
25 Тестирование программного кода: основные подходы и методы тестирования, модульное и функциональное тестирование.	2	4	4	10	ПК-4
26 Проектирование и документирование программного кода: понятия архитектуры ПО, комментирование кода, создание диаграмм UML.	2	2	4	8	ПК-4
Итого за семестр	36	36	108	180	
Итого	36	36	108	180	

5.2. Содержание разделов (тем) дисциплины

Согласована на портале № 72083

Содержание разделов (тем) дисциплины (в т.ч. по лекциям) приведено в таблице 5.2.

Таблица 5.2 – Содержание разделов (тем) дисциплины (в т.ч. по лекциям)

Названия разделов (тем) дисциплины	Содержание разделов (тем) дисциплины (в т.ч. по лекциям)	Трудоемкость (лекционные занятия), ч	Формируемые компетенции
1 семестр			
1 Введение в теорию программирования	Теория программирования. Алгоритм. Структуры данных. Формальные языки. Синтаксическая и семантическая структура языка программирования. Основные принципы выбора и применения теории программирования в различных областях, включая разработку программного обеспечения, компьютерные сети, базы данных, искусственный интеллект. принципы работы программ. Понятия в теории программирования: вычислимость, сложность вычислений, формальная верификация программ.	1	ПК-4
	Итого	1	
2 Интерпретаторы и компиляторы: преобразование программного кода в машинный код.	Интерпретаторы и компиляторы. Программные инструменты, преобразующие исходный код программы в машинный код. различные этапы включающие компиляцию программного кода: лексический анализ, синтаксический анализ, семантический анализ и генерацию машинного кода.	1	ПК-4
	Итого	1	
3 Основные понятия программирования: переменные, типы данных, операторы, условные выражения, циклы.	Процесс создания компьютерных программ, которые выполняют определенные задачи. Переменные. Типы данных, характеристики данных, такие как числа, строки, логические значения и т. д. Символы или ключевые слова, используемые для выполнения операций над данными. Арифметические операторы. Логические операторы. Условные выражения. Условные операторы. Циклы. Управляющие операторы.	2	ПК-4
	Итого	2	
4 Алгоритмы и структуры данных: массивы, списки, стеки, очереди.	Алгоритмы. Способы организации и хранения данных в памяти компьютера, структуры данных. Коллекции элементов одного типа, массивы. Списки. Структура данных, работающая по принципу "последним пришел, первым вышел" (LIFO). Структура данных, работающая по принципу "первым пришел, первым вышел" (FIFO). Основные операции с массивами, списками, стеками и очередями, вставка, удаление, доступ и поиск элементов. Алгоритмы сортировки. Алгоритмы поиска. Алгоритмы сортировки. Методы упорядочивания элементов в коллекции. Понимание алгоритмов и структур данных. Производительность алгоритмов и структур данных.	2	ПК-4
	Итого	2	

Согласована на портале № 72083

5 Функции и процедуры: создание, вызов, передача параметров.	Функции и процедуры. Фрагменты кода, предназначенные для выполнения определенной задачи. Создание функций и процедур. Вызов функций и процедур с передачей необходимых аргументов. Передача данных в функции и процедуры для их обработки. Возвращаемые значения функций после выполнения. Локальные переменные. Переменные, объявленные внутри функций и процедур и видимые только в их области видимости. Рекурсия - вызов функции или процедуры из самой себя. Модульность - разделение программы на независимые функции и процедуры для повышения читаемости и удобства разработки. Передача параметров по значению и по ссылке - различные способы передачи параметров, влияющие на работу функций и процедур. Инкапсуляция - сокрытие деталей реализации функций и процедур от внешнего кода для обеспечения абстракции. Функции и процедуры позволяют структурировать код, повторно использовать фрагменты и упрощать разработку и поддержку программного обеспечения.	2	ПК-4
	Итого	2	

Согласована на портале № 72083

6 Работа с файлами: чтение и запись данных, обработка текстовых и бинарных файлов.	Работа с файлами. Процесс взаимодействия программы с внешними файлами для чтения и записи данных. Чтение данных из файла. Извлечение информации из файла для последующей обработки в программе. Запись данных в файл. Сохранение результатов работы программы в файл для дальнейшего использования или обмена информацией. Текстовые файлы. Файлы, содержащие данные в виде текста, обычно в формате ASCII или Unicode. Бинарные файлы. Файлы, содержащие данные в формате, который может быть специфическим для определенного приложения или формата файла. Открытие файла. Создание связи между программой и файлом для последующего чтения или записи данных. Завершение связи между программой и файлом после чтения или записи данных. Обработка текстовых файлов - чтение или запись текстовых данных, выполнение операций поиска, замены, сортировки или других манипуляций с текстом. Обработка бинарных файлов - чтение или запись бинарных данных, выполнение операций с битами, структурами или другими элементами бинарного формата. Указатели на файлы. Специальные переменные, используемые для отслеживания текущей позиции чтения или записи в файле. Обработка ошибок при работе с файлами. Обработка возможных ошибок, таких как отсутствие файла, недостаточные права доступа или неправильный формат данных. Работа с файлами позволяет программам сохранять и загружать данные, обмениваться информацией между разными системами и осуществлять долгосрочное хранение данных.	2	ПК-4
	Итого	2	

Согласована на портале № 72083

7 Ошибки и исключения: обработка исключительных ситуаций, отладка программы.	Ошибки и исключения - неожиданные ситуации, возникающие во время выполнения программы, которые могут привести к некорректной работе или сбою. Обработка исключительных ситуаций - механизм, позволяющий программе обнаружить и обработать ошибки или исключения, чтобы предотвратить аварийное завершение программы. Исключения - специальные объекты, которые генерируются при возникновении ошибки или исключительной ситуации и передаются по стеку вызовов, пока не будут обработаны или пойманы. Блок try-catch - конструкция, которая позволяет программе перехватывать и обрабатывать исключения, предоставляя блок кода для выполнения в случае возникновения исключения. Обработка исключений - процесс анализа исключения, определение причины возникновения и принятие соответствующих мер для восстановления работы программы. Выброс исключения - явное создание исключительной ситуации в коде программы с использованием ключевого слова throw, чтобы указать на наличие ошибки или необычной ситуации. Отладка программы - процесс поиска и исправления ошибок в программе с целью обеспечения корректного ее выполнения. Инструменты отладки - программные средства, которые предоставляют возможности для поиска и исправления ошибок, такие как трассировка выполнения, точки останова и вывод отладочной информации. Логирование - процесс записи информации о ходе выполнения программы, включая ошибки и исключения, в специальные файлы журналов для последующего анализа и отладки. Обработка ошибок и исключений позволяет программам предотвратить аварийное завершение, предоставить пользователю информацию о возникших проблемах и выполнить соответствующие действия для нормального продолжения работы программы.	2	ПК-4
	Итого	2	

Согласована на портале № 72083

8 Работа с библиотеками и сторонними модулями: подключение, использование и документация.	Библиотеки и сторонние модули - готовые наборы функций, классов или компонентов, разработанные другими разработчиками и предназначенные для повторного использования в программных проектах. Подключение библиотек и модулей - процесс добавления их в проект с помощью специальных инструкций или команд, чтобы сделать доступными их функциональность и возможности в коде программы. Использование библиотек и модулей - включение функций, классов или компонентов из библиотеки в код программы для выполнения определенных задач или достижения определенного функционального поведения. Документация библиотек и модулей - набор информации и инструкций, предоставленных разработчиками библиотеки или модуля, о его функциях, классах, методах и параметрах, а также о способах использования и интеграции в программный проект. API (Application Programming Interface) - набор соглашений и интерфейсов, предоставляемых библиотекой или модулем, для взаимодействия с ними и использования их функциональности в приложении. Версионирование библиотек - процесс управления различными версиями библиотеки или модуля, чтобы обеспечить совместимость и поддержку в различных проектах. Зависимости - связи между различными библиотеками и модулями, которые требуются для их работы, и которые должны быть установлены или подключены, чтобы обеспечить корректную работу программы. Интеграция библиотек и модулей - процесс объединения функциональности и возможностей различных библиотек и модулей для создания комплексных программных решений. Управление зависимостями - процесс установки, обновления и управления версиями необходимых библиотек и модулей для обеспечения правильной работы программного проекта. Проверка документации - процесс ознакомления с документацией библиотеки или модуля для получения информации о его возможностях, функциях, примерах использования и ограничениях, а также для проверки соответствия документации реальной функциональности библиотеки или модуля.	2	ПК-4
	Итого	2	

Согласована на портале № 72083

9 Классы и структуры. Члены классов. Дружественность	Классы и структуры - основные средства организации и абстракции данных и поведения в объектно-ориентированном программировании. Члены классов - переменные и функции, определенные внутри класса, которые предоставляют доступ к данным и определяют поведение объектов этого класса. Поля классов - переменные, объявленные внутри класса, которые хранят данные, принадлежащие объектам этого класса. Методы классов - функции, объявленные внутри класса, которые определяют операции и поведение объектов этого класса. Конструкторы - специальные методы класса, которые вызываются при создании объекта класса и инициализируют его поля начальными значениями. Деструкторы - специальные методы класса, которые вызываются при уничтожении объекта класса и выполняют необходимые операции по освобождению ресурсов. Наследование - механизм, позволяющий создавать новые классы на основе уже существующих классов, наследуя их поля и методы. Полиморфизм - свойство объектов классов иметь различное поведение в зависимости от контекста использования. Дружественные классы и функции - классы и функции, которые имеют доступ к закрытым членам других классов, что обеспечивает совместное использование данных и функциональности между классами. Инкапсуляция - принцип объектно-ориентированного программирования, который обеспечивает сокрытие данных и реализации внутри класса и предоставляет доступ к ним только через определенные интерфейсы класса.	2	ПК-4
	Итого	2	

10 Основы объектно-ориентированного программирования: классы, объекты, наследование, полиморфизм.	Классы определяют структуру и поведение объектов в объектно-ориентированном программировании. Объекты - экземпляры классов, которые имеют собственное состояние и поведение. Наследование - механизм, позволяющий создавать новые классы на основе уже существующих классов, наследуя их свойства и методы. Родительский класс - класс, от которого наследуется другой класс. Он предоставляет общую функциональность, которую наследники могут использовать. Дочерний класс - класс, который наследует свойства и методы от родительского класса. Он может добавлять новую функциональность или изменять поведение унаследованных методов. Полиморфизм - возможность объектов разных классов обладать одним и тем же интерфейсом, но вести себя по-разному. Это позволяет использовать один и тот же код для работы с различными типами объектов. Полиморфные методы - методы, которые могут быть реализованы по-разному в разных классах, но имеют одно и то же имя и сигнатуру. Абстракция - процесс выделения общих характеристик и свойств объектов и их представление в виде классов и их отношений. Инкапсуляция - принцип, который позволяет объединить данные и методы, работающие с этими данными, в одну единицу, скрывая детали реализации и обеспечивая доступ к данным только через определенные методы. Композиция - механизм, позволяющий создавать более сложные объекты путем объединения нескольких объектов внутри других объектов.	2	ПК-4
	Итого	2	

Согласована на портале № 72083

12 Введение в алгоритмическое мышление и решение задач: разработка алгоритмов, построение блок-схем.	Алгоритм - последовательность шагов, предназначенных для решения определенной задачи. Алгоритмическое мышление - способность анализировать задачи, разбивать их на подзадачи, разрабатывать эффективные алгоритмы для их решения. Разработка алгоритмов - процесс создания алгоритмов для решения конкретных задач. Построение блок-схем - метод визуализации алгоритмов, основанный на использовании графических элементов (блоков), представляющих операции, условия и поток управления. Входные данные - информация, которая подается на вход алгоритма для обработки. Выходные данные - результат работы алгоритма, получаемый после его выполнения. 7. Переменные - символические обозначения, используемые для хранения данных во время выполнения алгоритма. Операторы - инструкции, определяющие действия, выполняемые алгоритмом, такие как присваивание значений переменным, выполнение арифметических операций и т.д. Условные операторы - операторы, позволяющие выполнять различные действия в зависимости от условий. Циклы - операторы, позволяющие повторять выполнение определенных действий до тех пор, пока выполняется определенное условие.	2	ПК-4
	Итого	2	

Согласована на портале № 72083

13 Работа с отладчиком: пошаговое выполнение кода, отслеживание значений переменных.	Отладчик - инструмент разработки программного обеспечения, позволяющий анализировать и исправлять ошибки в программе. Пошаговое выполнение кода - возможность отладчика выполнять программу пошагово, по одной инструкции за раз, для более детального анализа ее работы. Установка точек останова - механизм отладчика, позволяющий указать конкретные места в программе, на которых выполнение программы будет приостановлено для дальнейшего анализа. Продолжение выполнения программы - функция отладчика, которая позволяет программе продолжить выполнение после остановки на точке останова. Отслеживание значений переменных - возможность отладчика отслеживать значения переменных на каждом шаге выполнения программы, что помогает выявить ошибки и неправильное поведение программы. Просмотр стека вызовов - функция отладчика, позволяющая просматривать стек вызовов программы, чтобы понять последовательность вызовов функций и их взаимодействие. Инспектирование объектов - возможность отладчика просматривать свойства и состояние объектов во время выполнения программы для анализа и проверки корректности их работы. Установка условных точек останова - возможность установки точек останова, которые активируются только при выполнении определенного условия, что помогает сосредоточиться на конкретных ситуациях или проблемах в программе. Интерактивная консоль отладчика - дополнительный инструмент, позволяющий взаимодействовать с программой через консоль отладчика, выполнять команды, изменять значения переменных и просматривать результаты. Переключение контекста выполнения - возможность отладчика изменять контекст выполнения программы, например, переходить от одной функции к другой или изменять значения параметров функций для анализа различных сценариев работы программы.	2	ПК-4
	Итого	2	

Согласована на портале № 72083

15 Основные конструкции языка программирования Python	Переменные и присваивание значений: объявление переменных, присваивание значений, типы данных (числа, строки, списки и т.д.). Операторы и выражения: арифметические операторы, логические операторы, операторы сравнения, операторы присваивания. Условные выражения: if-else конструкции, операторы сравнения, операторы с логическими значениями (and, or, not). Циклы: цикл while, цикл for, использование циклов для итерации по последовательностям. Функции: определение функций, передача аргументов, возвращаемые значения, локальные и глобальные переменные. Структуры данных: списки, кортежи, словари, множества и их методы. Работа с файлами: открытие и закрытие файлов, чтение и запись данных, обработка исключений. Модули и пакеты: импортирование модулей, использование функций и классов из модулей, создание собственных модулей, организация кода в пакеты. Обработка строк: операции со строками, форматирование строк, работа с подстроками, методы строк. Исключения: обработка исключительных ситуаций, использование конструкции try-except-finally. Работа с классами и объектами: определение классов, создание объектов, методы классов, наследование. Регулярные выражения: использование регулярных выражений для поиска и обработки текстовых данных. Стандартная библиотека Python: обзор основных модулей и функций из стандартной библиотеки Python. Обработка и управление ошибками: использование исключений для обработки ошибок, логирование ошибок. Работа с внешними библиотеками: установка и использование сторонних библиотек с помощью инструментов управления зависимостями, импортирование и использование функций из внешних библиотек.	2	ПК-4
	Итого	2	

Согласована на портале № 72083

16 Основные конструкции языка программирования C++	Переменные и типы данных: объявление переменных, основные типы данных (целочисленные, вещественные, символьные, логические), модификаторы типов данных. Операторы и выражения: арифметические операторы, логические операторы, операторы сравнения, операторы присваивания, операторы инкремента и декремента. Условные выражения: if-else конструкции, оператор switch, операторы сравнения, операторы с логическими значениями (&&, ||, !). Циклы: цикл while, цикл do-while, цикл for, использование циклов для итерации по массивам и другим структурам данных. Функции: объявление функций, передача аргументов по значению и по ссылке, возвращаемые значения, перегрузка функций. Массивы: объявление массивов, инициализация, доступ к элементам массива, многомерные массивы. Указатели: объявление и инициализация указателей, операции с указателями, указатели на функции. Структуры и классы: определение структур и классов, объявление объектов, доступ к членам структуры или класса, наследование, полиморфизм. Исключения: обработка исключительных ситуаций с помощью try-catch блоков, генерация исключений, стандартные исключения. Пространства имён: создание и использование пространств имён для организации кода и предотвращения конфликтов имен. Ввод/вывод: работа с потоками ввода/вывода, использование стандартных потоков ввода/вывода (cin, cout, cerr), работа с файлами. Стандартная библиотека C++: обзор основных модулей и классов из стандартной библиотеки C++ (STL, алгоритмы, контейнеры и т.д.).	2	ПК-4
	Итого	2	

Согласована на портале № 72083

18 Жизненный цикл программного обеспечения	Анализ и определение требований: сбор и анализ требований к программному обеспечению, определение функциональных и нефункциональных требований. Проектирование: создание архитектуры программного обеспечения, проектирование интерфейсов, выбор используемых технологий и инструментов. Разработка: написание и тестирование кода, создание модулей и компонентов, интеграция их вместе, отладка и исправление ошибок. Тестирование: планирование и выполнение тестовых сценариев, проверка соответствия программного обеспечения требованиям, обнаружение и исправление ошибок. Развёртывание и внедрение: подготовка программного обеспечения к установке и использованию, установка на целевую платформу, настройка и настройка программы. Эксплуатация и поддержка: поддержка и обслуживание программного обеспечения в процессе его использования, исправление ошибок, улучшение и обновление функциональности. Снятие с эксплуатации: принятие решения о прекращении использования программного обеспечения, архивирование данных и удаление программы из эксплуатации. Управление изменениями: управление изменениями в программном обеспечении, версионирование, контроль и управление конфигурацией.	2	ПК-4
	Итого	2	
21 Работа с базами данных: основы SQL, создание и управление таблицами, выполнение запросов.	Основы SQL: описание SQL (Structured Query Language), его роль в работе с базами данных, основные операторы SQL. Создание таблиц: создание таблиц в базе данных, определение структуры таблицы, выбор подходящих типов данных для столбцов. Управление таблицами: добавление новых записей, изменение и удаление записей, изменение структуры таблицы (добавление, изменение и удаление столбцов). Выполнение запросов: написание SQL-запросов для извлечения данных из таблицы, использование операторов SELECT, WHERE, GROUP BY, ORDER BY и других для фильтрации и сортировки данных. Соединение таблиц: объединение двух или более таблиц для получения связанных данных, использование операторов JOIN, INNER JOIN, LEFT JOIN и других для выполнения соединений. Операции агрегации: использование функций агрегации (например, SUM, AVG, COUNT) для выполнения расчетов над данными в таблице.	2	ПК-4
	Итого	2	

Согласована на портале № 72083

22 Работа с графическим интерфейсом: основы разработки GUI-приложений, создание элементов управления, обработка событий.	Основы разработки GUI-приложений: понятие графического интерфейса пользователя (GUI), его роль в создании пользовательских приложений, преимущества GUI по сравнению с консольными интерфейсами. Создание элементов управления: использование библиотек и фреймворков для создания графических компонентов, таких как кнопки, поля ввода, списки, таблицы и другие. Расположение элементов управления: управление размещением элементов на экране, использование контейнеров и компоновщиков для создания структуры пользовательского интерфейса. Оформление элементов управления: настройка внешнего вида элементов, выбор цветовой схемы, шрифтов и других атрибутов, применение стилей и тем для достижения желаемого визуального эффекта. Обработка событий: реакция на пользовательские действия, такие как клики мыши, нажатия клавиш, перемещение указателя и другие события, использование обработчиков событий для выполнения соответствующих действий. Валидация данных: проверка корректности введенных пользователем данных, обработка ошибок и отображение сообщений об ошибках. Многопоточность и асинхронность: работа с потоками и асинхронными операциями для обеспечения отзывчивости пользовательского интерфейса и предотвращения блокировки интерфейса при выполнении длительных операций. Тестирование и отладка GUI-приложений: разработка и выполнение тестовых сценариев, поиск и исправление ошибок, использование инструментов для отладки GUI-кода. Документирование и сопровождение GUI-приложений: создание документации, описание интерфейса и функциональности, обновление и поддержка приложения в течение его жизненного цикла.	2	ПК-4
	Итого	2	

Согласована на портале № 72083

25 Тестирование программного кода: основные подходы и методы тестирования, модульное и функциональное тестирование.	Основные подходы к тестированию программного кода: ручное и автоматизированное тестирование, статическое и динамическое тестирование. Модульное тестирование: определение модуля программного кода, создание набора тестовых сценариев для проверки функциональности модуля, использование заглушек и фиктивных объектов для изоляции модуля от зависимостей. Функциональное тестирование: проверка соответствия функциональных требований программного кода, создание тестовых сценариев, включающих различные варианты использования и комбинации функций. Тестирование черного ящика: тестирование программного кода без знания его внутренней структуры и реализации, проверка входных и выходных данных, ожидаемых результатов и соответствия спецификации. Тестирование белого ящика: тестирование программного кода с полным знанием его внутренней структуры и реализации, проверка покрытия кода тестами, анализ путей выполнения и структуры кода. Интеграционное тестирование: проверка взаимодействия между модулями и компонентами программного кода, тестирование интерфейсов и передачи данных между ними. Регрессионное тестирование: повторное выполнение тестов после внесения изменений в программный код, чтобы убедиться, что ранее работающая функциональность осталась неповрежденной. Автоматизированное тестирование: использование специальных инструментов и фреймворков для автоматизации выполнения тестовых сценариев, ускорение процесса тестирования и обеспечение повторяемости результатов. Тестирование производительности: оценка производительности программного кода, проверка его отклика на различные нагрузки и объемы данных. Тестирование безопасности: проверка программного кода на наличие уязвимостей и возможных угроз безопасности, обеспечение защиты конфиденциальности и целостности данных.	2	ПК-4
	Итого	2	

Согласована на портале № 72083

26 Проектирование и документирование программного кода: понятия архитектуры ПО, комментирование кода, создание диаграмм UML.	Архитектура программного обеспечения: определение, роль и цели архитектуры, принципы и паттерны проектирования. Комментирование кода: важность комментариев в программном коде, правила и рекомендации по комментированию, типичные ошибки и проблемы. Диаграммы UML: обзор основных типов диаграмм UML (структурные и поведенческие), их назначение и применение в проектировании программного кода. Структурные диаграммы UML: диаграмма классов, диаграмма компонентов, диаграмма развертывания, их элементы и семантика. Поведенческие диаграммы UML: диаграмма последовательностей, диаграмма состояний, диаграмма активностей, их элементы и семантика. Процесс проектирования программного кода: этапы и последовательность проектирования, учет требований, выбор архитектурного стиля, определение компонентов и их взаимодействия. Проектирование классов: принципы SOLID, создание классов, определение атрибутов и методов, реализация ассоциаций и наследования. Документирование программного кода: роль и цели документирования, структура документации, правила и рекомендации по написанию документации. Инструменты для проектирования и документирования: популярные CASE-средства, среды разработки, инструменты для создания UML-диаграмм. Практические аспекты проектирования и документирования: лучшие практики, общие принципы и рекомендации, роли и ответственности в команде разработки.	2	ПК-4
	Итого	2	
	Итого за семестр	36	
	Итого	36	

5.3. Практические занятия (семинары)

Наименование практических занятий (семинаров) приведено в таблице 5.3.

Таблица 5.3. – Наименование практических занятий (семинаров)

Названия разделов (тем) дисциплины	Наименование практических занятий (семинаров)	Трудоемкость, ч	Формируемые компетенции
1 семестр			
3 Основные понятия программирования: переменные, типы данных, операторы, условные выражения, циклы.	Основы языка программирования: переменные и операторы	2	ПК-4
	Управляющие конструкции: условные операторы и циклы.	2	ПК-4
	Итого	4	

Согласована на портале № 72083

4 Алгоритмы и структуры данных: массивы, списки, стеки, очереди.	Массивы и строковые операции.	2	ПК-4
	Структуры данных: массивы, списки, стеки и очереди	2	ПК-4
	Итого	4	
5 Функции и процедуры: создание, вызов, передача параметров.	Работа с функциями: создание и вызов функций.	2	ПК-4
	Итого	2	
6 Работа с файлами: чтение и запись данных, обработка текстовых и бинарных файлов.	Работа с файлами: чтение и запись данных	2	ПК-4
	Итого	2	
7 Ошибки и исключения: обработка исключительных ситуаций, отладка программы.	Обработка исключительных ситуаций: использование исключений.	2	ПК-4
	Итого	2	
8 Работа с библиотеками и сторонними модулями: подключение, использование и документация.	Работа с библиотеками и модулями: подключение и использование готовых решений.	2	ПК-4
	Итого	2	
10 Основы объектно-ориентированного программирования: классы, объекты, наследование, полиморфизм.	Основы объектно-ориентированного программирования: классы и объекты.	2	ПК-4
	Итого	2	
12 Введение в алгоритмическое мышление и решение задач: разработка алгоритмов, построение блок-схем.	Разработка простых программных проектов.	2	ПК-4
	Основы алгоритмического мышления: разработка эффективных алгоритмов и структур данных.	2	ПК-4
	Итого	4	
16 Основные конструкции языка программирования C++	Работа с памятью: динамическое выделение и освобождение	2	ПК-4
	Итого	2	
18 Жизненный цикл программного обеспечения	Проработка этапов жизненного цикла ПО.	2	ПК-4
	Итого	2	
21 Работа с базами данных: основы SQL, создание и управление таблицами, выполнение запросов.	Работа с базами данных: создание, чтение, изменение и удаление данных.	2	ПК-4
	Итого	2	
22 Работа с графическим интерфейсом: основы разработки GUI-приложений, создание элементов управления, обработка событий.	Работа с графическим интерфейсом: создание пользовательских интерфейсов.	2	ПК-4
	Итого	2	

Согласована на портале № 72083

25 Тестирование программного кода: основные подходы и методы тестирования, модульное и функциональное тестирование.	Введение в отладку: отслеживание ошибок и исправление программ.	2	ПК-4
	Тестирование программного кода: проверка функциональности и обнаружение ошибок.	2	ПК-4
	Итого	4	
26 Проектирование и документирование программного кода: понятия архитектуры ПО, комментирование кода, создание диаграмм UML.	Разработка программной документации.	2	ПК-4
	Итого	2	
Итого за семестр		36	
Итого		36	

5.4. Лабораторные занятия

Не предусмотрено учебным планом

5.5. Курсовой проект / курсовая работа

Не предусмотрено учебным планом

5.6. Самостоятельная работа

Виды самостоятельной работы, трудоемкость и формируемые компетенции представлены в таблице 5.6.

Таблица 5.6. – Виды самостоятельной работы, трудоемкость и формируемые компетенции

Названия разделов (тем) дисциплины	Виды самостоятельной работы	Трудоемкость, ч	Формируемые компетенции	Формы контроля
1 семестр				
1 Введение в теорию программирования	Подготовка к зачету	4	ПК-4	Зачёт
	Подготовка к тестированию	2	ПК-4	Тестирование
	Итого	6		
2 Интерпретаторы и компиляторы: преобразование программного кода в машинный код.	Подготовка к зачету	4	ПК-4	Зачёт
	Подготовка к тестированию	2	ПК-4	Тестирование
	Итого	6		
3 Основные понятия программирования: переменные, типы данных, операторы, условные выражения, циклы.	Подготовка к зачету	4	ПК-4	Зачёт
	Подготовка к тестированию	2	ПК-4	Тестирование
	Итого	6		

Согласована на портале № 72083

4 Алгоритмы и структуры данных: массивы, списки, стеки, очереди.	Подготовка к зачету	4	ПК-4	Зачёт
	Подготовка к тестированию	2	ПК-4	Тестирование
	Итого	6		
5 Функции и процедуры: создание, вызов, передача параметров.	Подготовка к зачету	4	ПК-4	Зачёт
	Подготовка к тестированию	2	ПК-4	Тестирование
	Итого	6		
6 Работа с файлами: чтение и запись данных, обработка текстовых и бинарных файлов.	Подготовка к зачету	4	ПК-4	Зачёт
	Подготовка к тестированию	2	ПК-4	Тестирование
	Итого	6		
7 Ошибки и исключения: обработка исключительных ситуаций, отладка программы.	Подготовка к зачету	4	ПК-4	Зачёт
	Подготовка к тестированию	2	ПК-4	Тестирование
	Итого	6		
8 Работа с библиотеками и сторонними модулями: подключение, использование и документация.	Подготовка к зачету	4	ПК-4	Зачёт
	Подготовка к тестированию	2	ПК-4	Тестирование
	Итого	6		
9 Классы и структуры. Члены классов. Дружественность	Подготовка к зачету	4	ПК-4	Зачёт
	Подготовка к тестированию	2	ПК-4	Тестирование
	Итого	6		
10 Основы объектно-ориентированного программирования: классы, объекты, наследование, полиморфизм.	Подготовка к зачету	4	ПК-4	Зачёт
	Подготовка к тестированию	2	ПК-4	Тестирование
	Итого	6		
12 Введение в алгоритмическое мышление и решение задач: разработка алгоритмов, построение блок-схем.	Подготовка к зачету	4	ПК-4	Зачёт
	Подготовка к тестированию	2	ПК-4	Тестирование
	Итого	6		
13 Работа с отладчиком: пошаговое выполнение кода, отслеживание значений переменных.	Подготовка к зачету	4	ПК-4	Зачёт
	Подготовка к тестированию	2	ПК-4	Тестирование
	Итого	6		

Согласована на портале № 72083

15 Основные конструкции языка программирования Python	Подготовка к зачету	4	ПК-4	Зачёт
	Подготовка к тестированию	2	ПК-4	Тестирование
	Итого	6		
16 Основные конструкции языка программирования C++	Подготовка к зачету	4	ПК-4	Зачёт
	Подготовка к тестированию	2	ПК-4	Тестирование
	Итого	6		
18 Жизненный цикл программного обеспечения	Подготовка к зачету	4	ПК-4	Зачёт
	Подготовка к тестированию	2	ПК-4	Тестирование
	Итого	6		
21 Работа с базами данных: основы SQL, создание и управление таблицами, выполнение запросов.	Подготовка к зачету	4	ПК-4	Зачёт
	Подготовка к тестированию	2	ПК-4	Тестирование
	Итого	6		
22 Работа с графическим интерфейсом: основы разработки GUI-приложений, создание элементов управления, обработка событий.	Подготовка к зачету	2	ПК-4	Зачёт
	Подготовка к тестированию	2	ПК-4	Тестирование
	Итого	4		
25 Тестирование программного кода: основные подходы и методы тестирования, модульное и функциональное тестирование.	Подготовка к зачету	2	ПК-4	Зачёт
	Подготовка к тестированию	2	ПК-4	Тестирование
	Итого	4		
26 Проектирование и документирование программного кода: понятия архитектуры ПО, комментирование кода, создание диаграмм UML.	Подготовка к зачету	2	ПК-4	Зачёт
	Подготовка к тестированию	2	ПК-4	Тестирование
	Итого	4		
Итого за семестр		108		
Итого		108		

5.7. Соответствие компетенций, формируемых при изучении дисциплины, и видов учебной деятельности

Соответствие компетенций, формируемых при изучении дисциплины, и видов учебной деятельности представлено в таблице 5.7.

Таблица 5.7 – Соответствие компетенций, формируемых при изучении дисциплины, и видов занятий

Формируемые компетенции	Виды учебной деятельности			Формы контроля
	Лек. зан.	Прак. зан.	Сам. раб.	
ПК-4	+	+	+	Зачёт, Тестирование

Согласована на портале № 72083

6. Рейтинговая система для оценки успеваемости обучающихся

6.1. Балльные оценки для форм контроля

Балльные оценки для форм контроля представлены в таблице 6.1.

Таблица 6.1 – Балльные оценки

Формы контроля	Максимальный балл на 1-ую 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
1 семестр				
Зачёт	10	10	20	40
Тестирование	20	20	20	60
Итого максимум за период	30	30	40	100
Нарастающим итогом	30	60	100	100

6.2. Пересчет баллов в оценки за текущий контроль

Пересчет баллов в оценки за текущий контроль представлен в таблице 6.2.

Таблица 6.2 – Пересчет баллов в оценки за текущий контроль

Баллы на дату текущего контроля	Оценка
≥ 90% от максимальной суммы баллов на дату ТК	5
От 70% до 89% от максимальной суммы баллов на дату ТК	4
От 60% до 69% от максимальной суммы баллов на дату ТК	3
< 60% от максимальной суммы баллов на дату ТК	2

6.3. Пересчет суммы баллов в традиционную и международную оценку

Пересчет суммы баллов в традиционную и международную оценку представлен в таблице 6.3.

Таблица 6.3 – Пересчет суммы баллов в традиционную и международную оценку

Оценка	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)

7. Учебно-методическое и информационное обеспечение дисциплины

7.1. Основная литература

1. Теория языков программирования и методы трансляции: Учебное пособие / В. В. Романенко, В. Т. Калайда - 2019. 264 с. [Электронный ресурс]: — Режим доступа: https://edu.tusur.ru/publications/9043.

2. Кривцов, А. Н. Технологии программирования. Технология программирования на C/C++ : учебное пособие / А. Н. Кривцов. — Санкт-Петербург : СПбГУТ им. М.А. Бонч-Бруевича, 2021. — 274 с. [Электронный ресурс]: — Режим доступа: https://e.lanbook.com/book/279680.

Согласована на портале № 72083

3. Барков, И. А. Объектно-ориентированное программирование / И. А. Барков. — 2-е изд., стер. — Санкт-Петербург : Лань, 2023. — 700 с. [Электронный ресурс]: — Режим доступа: https://e.lanbook.com/book/329549.
4. Программирование. Сборник задач : учебное пособие / О. Г. Архипов, В. С. Батасова, П. В. Гречкина [и др.] ; под редакцией М. М. Марана. — Санкт-Петербург : Лань, 2022. — 140 с. [Электронный ресурс]: — Режим доступа: https://e.lanbook.com/book/206768.
5. Аршинский, В. Л. Программирование на языке высокого уровня. Основы алгоритмизации и программирования : учебное пособие / В. Л. Аршинский, И. А. Серышева. — Иркутск : ИРНИТУ, 2021. — 122 с. [Электронный ресурс]: — Режим доступа: https://e.lanbook.com/book/325445.
6. Забихуллин, Ф. З. Структурное программирование на C++ : учебное пособие / Ф. З. Забихуллин. — Уфа : БГПУ имени М. Акмуллы, 2019. — 45 с. [Электронный ресурс]: — Режим доступа: https://e.lanbook.com/book/131001.
7. Пай, П. Реактивное программирование на C++ / П. Пай, П. Абрахам ; перевод с английского В. Ю. Винника. — Москва : ДМК Пресс, 2019. — 324 с. [Электронный ресурс]: — Режим доступа: https://e.lanbook.com/book/131698.
8. Технология программирования: Учебное пособие / Н. В. Зариковская - 2018. 130 с. [Электронный ресурс]: — Режим доступа: https://edu.tusur.ru/publications/8244.
9. Технологии программирования: Учебное пособие / В. В. Кручинин - 2013. 271 с. [Электронный ресурс]: — Режим доступа: https://edu.tusur.ru/publications/2834.

7.2. Дополнительная литература

1. Введение в программирование и структуры данных / К. Фислер, Ш. Кришнамурти, Б. С. Лернер, Д. Г. Политц ; перевод с английского А. В. Снастина. — Москва : ДМК Пресс, 2022. — 440 с. [Электронный ресурс]: — Режим доступа: https://e.lanbook.com/book/314846.
2. Юрина, Т. А. Объектно-ориентированное программирование : учебно-методическое пособие / Т. А. Юрина. — Омск : СибАДИ, 2023. — 72 с. [Электронный ресурс]: — Режим доступа: https://e.lanbook.com/book/338576.
3. Городняя, Л. В. Парадигма программирования : учебное пособие для вузов / Л. В. Городняя. — 2-е изд., стер. — Санкт-Петербург : Лань, 2021. — 232 с. [Электронный ресурс]: — Режим доступа: https://e.lanbook.com/book/151660.
4. Лебеденко, Л. Ф. Основы программирования на C++ : учебное пособие / Л. Ф. Лебеденко, О. И. Моренкова ; RU. — 2-е изд., переработанное и дополненное. — Новосибирск : СибГУТИ, 2021. — 200 с. [Электронный ресурс]: — Режим доступа: https://e.lanbook.com/book/257261.
5. Программирование : учебное пособие / А. Ф. Базаркин, О. А. Бакаева, Н. В. Вознесенская, О. Н. Шалина. — Саранск : МГПИ им. М.Е. Евсевьева, 2018. — 174 с. [Электронный ресурс]: — Режим доступа: https://e.lanbook.com/book/163496.
6. Рацеев, С. М. Программирование на языке Си : учебное пособие для вузов / С. М. Рацеев. — Санкт-Петербург : Лань, 2022. — 332 с. [Электронный ресурс]: — Режим доступа: https://e.lanbook.com/book/193320.
7. Дженесерет, М. Введение в логическое программирование / М. Дженесерет, В. К. Чаудри ; перевод с английского С. В. Минц. — Москва : ДМК Пресс, 2022. — 192 с. [Электронный ресурс]: — Режим доступа: https://e.lanbook.com/book/241130.
8. Фаулер, М. Asyncio и конкурентное программирование на Python / М. Фаулер ; перевод с английского А. А. Слинкина. — Москва : ДМК Пресс, 2023. — 398 с. [Электронный ресурс]: — Режим доступа: https://e.lanbook.com/book/314963.
9. Згуральская, Е. Н. Технологии программирования : учебное пособие / Е. Н. Згуральская. — Ульяновск : УлГТУ, 2020. — 71 с. [Электронный ресурс]: — Режим доступа: https://e.lanbook.com/book/165011.
10. Хилл, К. Научное программирование на Python / К. Хилл ; перевод с английского А. В. Снастина. — Москва : ДМК Пресс, 2021. — 646 с. [Электронный ресурс]: — Режим доступа: https://e.lanbook.com/book/241031.
11. Потапов, И. И. Язык программирования : учебное пособие / И. И. Потапов. — Хабаровск : ДВГУПС, 2019. — 79 с. [Электронный ресурс]: — Режим доступа: https://e.lanbook.com/book/179358.

Согласована на портале № 72083

12. Введение в программирование и структуры данных / К. Фислер, Ш. Кришнамурти, Б. С. Лернер, Д. Г. Политц ; перевод с английского А. В. Снастина. — Москва : ДМК Пресс, 2022. — 440 с. [Электронный ресурс]: — Режим доступа: https://e.lanbook.com/book/314846.

13. Булгаков, Д. А. Программирование интерактивных приложений на языке C# : учебное пособие / Д. А. Булгаков. — Санкт-Петербург : ГУАП, 2022. — 159 с. пользователей. [Электронный ресурс]: — Режим доступа: https://e.lanbook.com/book/341012.

14. Гримм, Р. Параллельное программирование на современном C++. Что каждый профессионал должен знать о параллельном программировании / Р. Гримм ; перевод с английского В. Ю. Винника. — Москва : ДМК Пресс, 2022. — 616 с. пользователей. [Электронный ресурс]: — Режим доступа: https://e.lanbook.com/book/314870.

15. Федоров, В. Б. Основы алгоритмизации и программирования : учебное пособие / В. Б. Федоров, М. М. Сенявин. — Москва : РТУ МИРЭА, 2022 — Часть 1 — 2022. — 203 с. [Электронный ресурс]: — Режим доступа: https://e.lanbook.com/book/311198.

16. Пош, М. Программирование встроенных систем на C++ 17 : учебное пособие / М. Пош ; перевод с английского А. В. Снастина. — Москва : ДМК Пресс, 2020. — 394 с. [Электронный ресурс]: — Режим доступа: https://e.lanbook.com/book/140589.

17. Как проектировать программы. Введение в программирование и компьютерные вычисления / М. Фелляйзен, Р. Б. Findлер, М. Флэтт, Ш. Кришнаму ; под редакцией П. Б. Иванова [и др.] ; перевод с английского А. Н. Киселева. — Москва : ДМК Пресс, 2022. — 724 с. [Электронный ресурс]: — Режим доступа: https://e.lanbook.com/book/314867.

18. Рамальо, Л. Python - К вершинам мастерства. Лаконичное и эффективное программирование / Л. Рамальо ; перевод с английского А. А. Слинкина. — 2-е изд. — Москва : ДМК Пресс, 2022. — 898 с. [Электронный ресурс]: — Режим доступа: https://e.lanbook.com/book/314918.

19. Информационные технологии. Языки и системы программирования: Учебное пособие / А. О. Семкин, А. С. Перин - 2021. 180 с. [Электронный ресурс]: — Режим доступа: https://edu.tusur.ru/publications/9500.

20. Дейл, Н. Программирование на C++ : самоучитель / Н. Дейл, Ч. Уимз, М. Хедингтон. — Москва : ДМК Пресс, 2007. — 672 с. [Электронный ресурс]: — Режим доступа: https://e.lanbook.com/book/1219.

21. Залогова, Л. А. Основы объектно-ориентированного программирования на базе языка C# : учебное пособие для вузов / Л. А. Залогова. — 3-е изд., стер. — Санкт-Петербург : Лань, 2021. — 192 с. [Электронный ресурс]: — Режим доступа: https://e.lanbook.com/book/176894.

22. Иванченко, А. Н. Основы программирования (язык C++) : учебное пособие / А. Н. Иванченко, А. А. Масленников, П. А. Иванченко. — Новочеркасск : ЮРГПУ, 2016. — 160 с. [Электронный ресурс]: — Режим доступа: https://e.lanbook.com/book/180936.

7.3. Учебно-методические пособия

7.3.1. Обязательные учебно-методические пособия

1. Объектно-ориентированное программирование: Учебное пособие / В. В. Романенко - 2016. 475 с. [Электронный ресурс]: — Режим доступа: https://edu.tusur.ru/publications/6300.

2. Кривцов, А. Н. Технологии программирования. Технология программирования на C/C++ : учебное пособие / А. Н. Кривцов. — Санкт-Петербург : СПбГУТ им. М.А. Бонч-Бруевича, 2021. — 274 с. [Электронный ресурс]: — Режим доступа: https://e.lanbook.com/book/279680.

3. Новицкая, И. А. Математическое программирование. Линейное программирование : учебное пособие / И. А. Новицкая, Т. С. Зайцева, А. Е. Мастилин. — Новосибирск : СГУПС, 2020. — 170 с. [Электронный ресурс]: — Режим доступа: https://e.lanbook.com/book/164616.

4. Кривцов, А. Н. Алгоритмизация и программирование. Основы программирования на C/C++ : учебное пособие / А. Н. Кривцов, С. В. Хорошенко. — Санкт-Петербург : СПбГУТ им. М.А. Бонч-Бруевича, 2020. — 202 с. [Электронный ресурс]: — Режим доступа: https://e.lanbook.com/book/180057.

5. Основы программирования на языке C++ : учебное пособие / А. И. Давыдов, Е. С. Калинина, И. Л. Саля, С. А. Ступаков. — Омск : ОмГУПС, 2022. — 86 с. [Электронный ресурс]: — Режим доступа: https://e.lanbook.com/book/264491.

Согласована на портале № 72083

6. Резова, Н. Л. Технология программирования : учебное пособие / Н. Л. Резова, Г. Ш. Шкаберина. — Красноярск : СибГУ им. академика М. Ф. Решетнёва, 2019. — 94 с. [Электронный ресурс]: — Режим доступа: https://e.lanbook.com/book/147448.
7. Златопольский, Д. М. 1400 задач по программированию : руководство / Д. М. Златопольский. — Москва : ДМК Пресс, 2020. — 192 с. [Электронный ресурс]: — Режим доступа: https://e.lanbook.com/book/140594.

7.3.2. Учебно-методические пособия для лиц с ограниченными возможностями здоровья и инвалидов

Учебно-методические материалы для самостоятельной и аудиторной работы обучающихся из числа лиц с ограниченными возможностями здоровья и инвалидов предоставляются в формах, адаптированных к ограничениям их здоровья и восприятия информации.

Для лиц с нарушениями зрения:
– в форме электронного документа;
– в печатной форме увеличенным шрифтом.

Для лиц с нарушениями слуха:
– в форме электронного документа;
– в печатной форме.

Для лиц с нарушениями опорно-двигательного аппарата:
– в форме электронного документа;
– в печатной форме.

7.4. Современные профессиональные базы данных и информационные справочные системы

1. При изучении дисциплины рекомендуется обращаться к современным базам данных, информационно-справочным и поисковым системам, к которым у ТУСУРа открыт доступ: https://lib.tusur.ru/ru/resursy/bazy-dannyh.

8. Материально-техническое и программное обеспечение дисциплины

8.1. Материально-техническое и программное обеспечение для лекционных занятий

Для проведения занятий лекционного типа, групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации используется учебная аудитория с достаточным количеством посадочных мест для учебной группы, оборудованная доской и стандартной учебной мебелью. Имеются мультимедийное оборудование и учебно-наглядные пособия, обеспечивающие тематические иллюстрации по лекционным разделам дисциплины.

8.2. Материально-техническое и программное обеспечение для практических занятий

Лаборатория цифровой обработки сигналов: учебная аудитория для проведения занятий практического типа, учебная аудитория для проведения занятий лабораторного типа, учебная аудитория для проведения занятий семинарского типа, помещение для курсового проектирования (выполнения курсовых работ), помещение для проведения групповых и индивидуальных консультаций, помещение для проведения текущего контроля и промежуточной аттестации; 634034, Томская область, г. Томск, Вершинина улица, д. 47, 210а ауд.

Описание имеющегося оборудования:
- ТВ BBQ;
- Магнитно-маркерная доска;
- Магнитно-маркерная (переносная);
- Комплект специализированной учебной мебели;
- Рабочее место преподавателя.

Программное обеспечение:
- Google Chrome;
- Microsoft Visual Studio 2010;
- Microsoft Windows XP Professional;

8.3. Материально-техническое и программное обеспечение для самостоятельной работы

Согласована на портале № 72083

Для самостоятельной работы используются учебные аудитории (компьютерные классы), расположенные по адресам:

- 634050, Томская область, г. Томск, Ленина проспект, д. 40, 233 ауд.;
- 634045, Томская область, г. Томск, ул. Красноармейская, д. 146, 209 ауд.;
- 634034, Томская область, г. Томск, Вершинина улица, д. 47, 126 ауд.;
- 634034, Томская область, г. Томск, Вершинина улица, д. 74, 207 ауд.

Описание имеющегося оборудования:

- учебная мебель;
- компьютеры;
- компьютеры подключены к сети «Интернет» и обеспечивают доступ в электронную информационно-образовательную среду ТУСУРа.

Перечень программного обеспечения:

- Microsoft Windows;
- OpenOffice;
- Kaspersky Endpoint Security 10 для Windows;
- 7-Zip;
- Google Chrome.

8.4. Материально-техническое обеспечение дисциплины для лиц с ограниченными возможностями здоровья и инвалидов

Освоение дисциплины лицами с ограниченными возможностями здоровья и инвалидами осуществляется с использованием средств обучения общего и специального назначения.

При занятиях с обучающимися с **нарушениями слуха** предусмотрено использование звукоусиливающей аппаратуры, мультимедийных средств и других технических средств приема/передачи учебной информации в доступных формах, мобильной системы преподавания для обучающихся с инвалидностью, портативной индукционной системы. Учебная аудитория, в которой занимаются обучающиеся с нарушением слуха, оборудована компьютерной техникой, аудиотехникой, видеотехникой, электронной доской, мультимедийной системой.

При занятиях с обучающимися с **нарушениями зрения** предусмотрено использование в лекционных и учебных аудиториях возможности просмотра удаленных объектов (например, текста на доске или слайда на экране) при помощи видеоувеличителей для комфортного просмотра.

При занятиях с обучающимися с **нарушениями опорно-двигательного аппарата** используются альтернативные устройства ввода информации и другие технические средства приема/передачи учебной информации в доступных формах, мобильной системы обучения для людей с инвалидностью.

9. Оценочные материалы и методические рекомендации по организации изучения дисциплины

9.1. Содержание оценочных материалов для текущего контроля и промежуточной аттестации

Для оценки степени сформированности и уровня освоения закрепленных за дисциплиной компетенций используются оценочные материалы, представленные в таблице 9.1.

Таблица 9.1 – Формы контроля и оценочные материалы

Названия разделов (тем) дисциплины	Формируемые компетенции	Формы контроля	Оценочные материалы (ОМ)
1 Введение в теорию программирования	ПК-4	Зачёт	Перечень вопросов для зачета
		Тестирование	Примерный перечень тестовых заданий
2 Интерпретаторы и компиляторы: преобразование программного кода в машинный код.	ПК-4	Зачёт	Перечень вопросов для зачета
		Тестирование	Примерный перечень тестовых заданий

Согласована на портале № 72083

3 Основные понятия программирования: переменные, типы данных, операторы, условные выражения, циклы.	ПК-4	Зачёт	Перечень вопросов для зачета
		Тестирование	Примерный перечень тестовых заданий
4 Алгоритмы и структуры данных: массивы, списки, стеки, очереди.	ПК-4	Зачёт	Перечень вопросов для зачета
		Тестирование	Примерный перечень тестовых заданий
5 Функции и процедуры: создание, вызов, передача параметров.	ПК-4	Зачёт	Перечень вопросов для зачета
		Тестирование	Примерный перечень тестовых заданий
6 Работа с файлами: чтение и запись данных, обработка текстовых и бинарных файлов.	ПК-4	Зачёт	Перечень вопросов для зачета
		Тестирование	Примерный перечень тестовых заданий
7 Ошибки и исключения: обработка исключительных ситуаций, отладка программы.	ПК-4	Зачёт	Перечень вопросов для зачета
		Тестирование	Примерный перечень тестовых заданий
8 Работа с библиотеками и сторонними модулями: подключение, использование и документация.	ПК-4	Зачёт	Перечень вопросов для зачета
		Тестирование	Примерный перечень тестовых заданий
9 Классы и структуры. Члены классов. Дружественность	ПК-4	Зачёт	Перечень вопросов для зачета
		Тестирование	Примерный перечень тестовых заданий
10 Основы объектно-ориентированного программирования: классы, объекты, наследование, полиморфизм.	ПК-4	Зачёт	Перечень вопросов для зачета
		Тестирование	Примерный перечень тестовых заданий
12 Введение в алгоритмическое мышление и решение задач: разработка алгоритмов, построение блок-схем.	ПК-4	Зачёт	Перечень вопросов для зачета
		Тестирование	Примерный перечень тестовых заданий
13 Работа с отладчиком: пошаговое выполнение кода, отслеживание значений переменных.	ПК-4	Зачёт	Перечень вопросов для зачета
		Тестирование	Примерный перечень тестовых заданий
15 Основные конструкции языка программирования Python	ПК-4	Зачёт	Перечень вопросов для зачета
		Тестирование	Примерный перечень тестовых заданий

Согласована на портале № 72083

16 Основные конструкции языка программирования C++	ПК-4	Зачёт	Перечень вопросов для зачета
		Тестирование	Примерный перечень тестовых заданий
18 Жизненный цикл программного обеспечения	ПК-4	Зачёт	Перечень вопросов для зачета
		Тестирование	Примерный перечень тестовых заданий
21 Работа с базами данных: основы SQL, создание и управление таблицами, выполнение запросов.	ПК-4	Зачёт	Перечень вопросов для зачета
		Тестирование	Примерный перечень тестовых заданий
22 Работа с графическим интерфейсом: основы разработки GUI-приложений, создание элементов управления, обработка событий.	ПК-4	Зачёт	Перечень вопросов для зачета
		Тестирование	Примерный перечень тестовых заданий
25 Тестирование программного кода: основные подходы и методы тестирования, модульное и функциональное тестирование.	ПК-4	Зачёт	Перечень вопросов для зачета
		Тестирование	Примерный перечень тестовых заданий
26 Проектирование и документирование программного кода: понятия архитектуры ПО, комментирование кода, создание диаграмм UML.	ПК-4	Зачёт	Перечень вопросов для зачета
		Тестирование	Примерный перечень тестовых заданий

Шкала оценки сформированности отдельных планируемых результатов обучения по дисциплине приведена в таблице 9.2.

Таблица 9.2 – Шкала оценки сформированности планируемых результатов обучения по дисциплине

Оценка	Баллы за ОМ	Формулировка требований к степени сформированности планируемых результатов обучения		
		знать	уметь	владеть
2 (неудовлетворительно)	< 60% от максимальной суммы баллов	отсутствие знаний или фрагментарные знания	отсутствие умений или частично освоенное умение	отсутствие навыков или фрагментарные применение навыков
3 (удовлетворительно)	от 60% до 69% от максимальной суммы баллов	общие, но не структурированные знания	в целом успешно, но не систематически осуществляемое умение	в целом успешное, но не систематическое применение навыков

Согласована на портале № 72083

4 (хорошо)	от 70% до 89% от максимальной суммы баллов	сформированные, но содержащие отдельные проблемы знания	в целом успешное, но содержащие отдельные пробелы умение	в целом успешное, но содержащие отдельные пробелы применение навыков
5 (отлично)	≥ 90% от максимальной суммы баллов	сформированные систематические знания	сформированное умение	успешное и систематическое применение навыков

Шкала комплексной оценки сформированности компетенций приведена в таблице 9.3.

Таблица 9.3 – Шкала комплексной оценки сформированности компетенций

Оценка	Формулировка требований к степени компетенции
2 (неудовлетворительно)	Не имеет необходимых представлений о проверяемом материале или Знать на уровне **ориентирования**, представлений. Обучающийся знает основные признаки или термины изучаемого элемента содержания, их отнесенность к определенной науке, отрасли или объектам, узнает в текстах, изображениях или схемах и знает, к каким источникам нужно обращаться для более детального его усвоения.
3 (удовлетворительно)	Знать и уметь на **репродуктивном** уровне. Обучающихся знает изученный элемент содержания репродуктивно: произвольно воспроизводит свои знания устно, письменно или в демонстрируемых действиях.
4 (хорошо)	Знать, уметь, владеть на **аналитическом** уровне. Зная на репродуктивном уровне, указывать на особенности и взаимосвязи изученных объектов, на их достоинства, ограничения, историю и перспективы развития и особенности для разных объектов усвоения.
5 (отлично)	Знать, уметь, владеть на **системном** уровне. Обучающийся знает изученный элемент содержания системно, произвольно и доказательно воспроизводит свои знания устно, письменно или в демонстрируемых действиях, учитывая и указывая связи и зависимости между этим элементом и другими элементами содержания дисциплины, его значимость в содержании дисциплины.

9.1.1. Примерный перечень тестовых заданий

1. Что такое переменная в программировании?
   a) Комментарий в коде
   b) Имя функции
   c) Место хранения данных
   d) Ошибка в программе
2. Какой символ используется для обозначения комментария на языке Си?
   a) #
   b) *
   c) !
   d) //
3. Какая операция используется для объединения двух строк в языке программирования?
   a) +
   b) -
   c) *
   d) /

Согласована на портале № 72083

4. Что означает условное выражение в программировании?
   a) Цикл, повторяющийся определенное количество раз
   b) Блок кода, выполняющийся только при выполнении определенного условия
   c) Функция, возвращающая результат вычисления
   d) Ошибка, возникающая при выполнении программы
5. Какой оператор используется для проверки равенства значений в языке программирования Си?
   a) ==
   b) =
   c) !=
   d) <>
6. Какой цикл в программировании на Си позволяет выполнять код до тех пор, пока заданное условие истинно?
   a) for
   b) while
   c) repeat
   d) if
7. Что такое функция в программировании?
   a) Условный оператор
   b) Цикл повторения
   c) Блок кода для выполнения определенной задачи
   d) Значение, хранящееся в переменной
8. Какой оператор используется для ввода данных с клавиатуры в языке программирования Си?
   a) input()
   b) print()
   c) read()
   d) scanf()
9. Какой оператор используется для выполнения одного из нескольких возможных вариантов кода в зависимости от значения переменной?
   a) if-else
   b) for
   c) switch-case
   d) while
10. Что такое компиляция в программировании?
    a) Процесс преобразования исходного кода в машинный код
    b) Процесс отладки программы
    c) Процесс создания документации к программе
    d) Ошибка, возникающая при выполнении программы

9.1.2. Перечень вопросов для зачета

1. Что такое алгоритм и какие основные этапы разработки алгоритма?
2. Какие основные типы данных используются в программировании и какие операции можно выполнять с этими типами данных?
3. Что такое условные операторы и как они используются для принятия решений в программе?
4. Какие основные циклы программирования существуют и в каких случаях они применяются?
5. Что такое функция и какие основные принципы работы с функциями в программировании?
6. Как осуществляется работа с динамической памятью и какие проблемы могут возникнуть при работе с памятью?
7. Как обрабатывать пользовательский ввод в программе и как выполнять валидацию пользовательских данных?
8. Как осуществляется работа с базами данных и какие операции можно выполнять с таблицами?
9. Как создавать графический интерфейс в программе и как обрабатывать события от

Согласована на портале № 72083

пользователей?
10. Что такое многопоточное программирование и какие проблемы могут возникнуть при работе с потоками?
11. Как осуществляется работа с сетью и какие сетевые протоколы используются в программировании?
12. Как разрабатывать веб-приложения и какие технологии используются для работы с серверной частью?
13. Как осуществлять тестирование программного кода и какие методы тестирования существуют?
14. Что такое архитектура ПО и какие инструменты используются для документирования программного кода?
15. Какие основные принципы разработки open source проектов и как они способствуют развитию программирования?
16. Как осуществлять комментирование программного кода и какие основные соглашения по оформлению кода существуют?
17. Как использовать диаграммы UML для проектирования программного кода и визуализации его структуры?

9.2. Методические рекомендации

Учебный материал излагается в форме, предполагающей самостоятельное мышление студентов, самообразование. При этом самостоятельная работа студентов играет решающую роль в ходе всего учебного процесса.

Начать изучение дисциплины необходимо со знакомства с рабочей программой, списком учебно-методического и программного обеспечения. Самостоятельная работа студента включает работу с учебными материалами, выполнение контрольных мероприятий, предусмотренных учебным планом.

В процессе изучения дисциплины для лучшего освоения материала необходимо регулярно обращаться к рекомендуемой литературе и источникам, указанным в учебных материалах; пользоваться через кабинет студента на сайте Университета образовательными ресурсами электронно-библиотечной системы, а также общедоступными интернет-порталами, содержащими научно-популярные и специализированные материалы, посвященные различным аспектам учебной дисциплины.

При самостоятельном изучении тем следуйте рекомендациям:

– чтение или просмотр материала осуществляйте со скоростью, достаточной для индивидуального понимания и освоения материала, выделяя основные идеи; на основании изученного составить тезисы. Освоив материал, попытаться соотнести теорию с примерами из практики;

– если в тексте встречаются незнакомые или малознакомые термины, следует выяснить их значение для понимания дальнейшего материала;

– осмысливайте прочитанное и изученное, отвечайте на предложенные вопросы.

Студенты могут получать индивидуальные консультации, в т.ч. с использованием средств телекоммуникации.

По дисциплине могут проводиться дополнительные занятия, в т.ч. в форме вебинаров. Расписание вебинаров и записи вебинаров публикуются в электронном курсе / электронном журнале по дисциплине.

9.3. Требования к оценочным материалам для лиц с ограниченными возможностями здоровья и инвалидов

Для лиц с ограниченными возможностями здоровья и инвалидов предусмотрены дополнительные оценочные материалы, перечень которых указан в таблице 9.4.

Таблица 9.4 – Дополнительные материалы оценивания для лиц с ограниченными возможностями здоровья и инвалидов

Категории обучающихся	Виды дополнительных оценочных материалов	Формы контроля и оценки результатов обучения

Согласована на портале № 72083

С нарушениями слуха	Тесты, письменные самостоятельные работы, вопросы к зачету, контрольные работы	Преимущественно письменная проверка
С нарушениями зрения	Собеседование по вопросам к зачету, опрос по терминам	Преимущественно устная проверка (индивидуально)
С нарушениями опорно-двигательного аппарата	Решение дистанционных тестов, контрольные работы, письменные самостоятельные работы, вопросы к зачету	Преимущественно дистанционными методами
С ограничениями по общемедицинским показаниям	Тесты, письменные самостоятельные работы, вопросы к зачету, контрольные работы, устные ответы	Преимущественно проверка методами, определяющимися исходя из состояния обучающегося на момент проверки

9.4. Методические рекомендации по оценочным материалам для лиц с ограниченными возможностями здоровья и инвалидов

Для лиц с ограниченными возможностями здоровья и инвалидов предусматривается доступная форма предоставления заданий оценочных средств, а именно:

– в печатной форме;
– в печатной форме с увеличенным шрифтом;
– в форме электронного документа;
– методом чтения ассистентом задания вслух;
– предоставление задания с использованием сурдоперевода.

Лицам с ограниченными возможностями здоровья и инвалидам увеличивается время на подготовку ответов на контрольные вопросы. Для таких обучающихся предусматривается доступная форма предоставления ответов на задания, а именно:

– письменно на бумаге;
– набор ответов на компьютере;
– набор ответов с использованием услуг ассистента;
– представление ответов устно.

Процедура оценивания результатов обучения лиц с ограниченными возможностями здоровья и инвалидов по дисциплине предусматривает предоставление информации в формах, адаптированных к ограничениям их здоровья и восприятия информации:

Для лиц с нарушениями зрения:
– в форме электронного документа;
– в печатной форме увеличенным шрифтом.

Для лиц с нарушениями слуха:
– в форме электронного документа;
– в печатной форме.

Для лиц с нарушениями опорно-двигательного аппарата:
– в форме электронного документа;
– в печатной форме.

При необходимости для лиц с ограниченными возможностями здоровья и инвалидов процедура оценивания результатов обучения может проводиться в несколько этапов.

Согласована на портале № 72083

ЛИСТ СОГЛАСОВАНИЯ

Рассмотрена и одобрена на заседании кафедры ТУ
протокол № 23 от «15» 2 2023 г.

СОГЛАСОВАНО:

Должность	Инициалы, фамилия	Подпись
Заведующий выпускающей каф. ТУ	Т.Р. Газизов	Согласовано, dccabe2f-73cc-455a-90f8-2fcc230a841e
Заведующий обеспечивающей каф. ТУ	Т.Р. Газизов	Согласовано, dccabe2f-73cc-455a-90f8-2fcc230a841e
И.О. начальника учебного управления	И.А. Лариошина	Согласовано, c3195437-a02f-4972-a7c6-ab6ee1f21e73

ЭКСПЕРТЫ:

Доцент, каф. ТУ	А.Н. Булдаков	Согласовано, d65c269c-f546-4509-b920-73aeef59fee4
Старший преподаватель, каф. ТУ	А.В. Бусыгина	Согласовано, 7d0bdef1-6f57-4269-9fbe-4beb03053805

РАЗРАБОТАНО:

Ассистент, каф. ТУ	А.В. Осинцев	Разработано, eaca1dbf-3450-4b0d-8d0c-8f642316f32b
Профессор, каф. ТУ	С.П. Куксенко	Разработано, 51277244-a412-430d-9479-a52ed425b1e9

Согласована на портале № 72083