

ПЕРВОЕ ВЫСШЕЕ ТЕХНИЧЕСКОЕ УЧЕБНОЕ ЗАВЕДЕНИЕ РОССИИ



МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение
высшего образования
САНКТ-ПЕТЕРБУРГСКИЙ ГОРНЫЙ УНИВЕРСИТЕТ

СОГЛАСОВАНО

УТВЕРЖДАЮ

Руководитель ОПОП ВО
доцент Е.Б. Мазаков

Проректор по образовательной
деятельности
Д. Г. Петраков

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

ПРОГРАММИРОВАНИЕ

Уровень высшего образования	Бакалавриат
Направление подготовки	09.03.01 Информатика и вычислительная техника
Направленность (профиль):	Автоматизированные системы обработки информации и управления
Квалификация выпускника:	Бакалавр
Форма обучения:	очная
Составитель:	доц. Гурко А. В.

Санкт-Петербург

Рабочая программа дисциплины «Программирование» разработана:

- в соответствии с требованиями ФГОС ВО – бакалавриат по направлению подготовки «09.03.01 Информатика и вычислительная техника», утвержденного приказом Минобрнауки России № 929 от 19 сентября 2017 г.;

- на основании учебного плана бакалавриата по направлению подготовки «09.03.01 Информатика и вычислительная техника», направленность (профиль) «Автоматизированные системы обработки информации и управления».

Составитель: _____ к.т.н., доц. А. В. Гурко

Рабочая программа рассмотрена и одобрена на заседании кафедры Информационных систем и вычислительной техники от 25.01.2021, протокол № 7.

Заведующий кафедрой _____ к.т.н., доцент Е.Б. Мазаков

Рабочая программа согласована:

Начальник отдела лицензирования, аккредитации и контроля качества образования _____ Ю. А. Дубровская

Начальник отдела методического обеспечения учебного процесса _____ А. Ю. Романчиков

1. ЦЕЛИ И ЗАДАЧИ ДИСЦИПЛИНЫ

Дисциплина «Программирование» предназначена для бакалавров, специализирующихся в области проектирования и исследования автоматизированных систем обработки информации и управления.

Цель дисциплины – овладение студентами приемами программирования на алгоритмическом языке и приобретение практических навыков в решении задач на персональных ЭВМ.

Основными задачами изучения дисциплины являются:

Формирование понимания принципов работы современных информационных технологий и программных средств, в том числе отечественного производства, умения их использовать их при решении задач профессиональной деятельности.

Формирование навыка решать стандартные задачи профессиональной деятельности с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности.

Формирование навыка разрабатывать алгоритмы и программы, пригодные для практического применения.

Способность осваивать методики использования программных средств для решения практических задач.

Способность разрабатывать компоненты программно-аппаратных комплексов, ПО, базы данных, используя современные инструментальные средства и технологии программирования, включая проектирование, отладку, проверку работоспособности и модификацию ПО

2. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОПОП ВО

Дисциплина «Программирование» входит в состав обязательной части Блока 1 «Дисциплины (модули)» основной профессиональной образовательной программы бакалавриата по направлению подготовки «09.03.01 Информатика и вычислительная техника», направленность (профиль) программы «Автоматизированные системы обработки информации и управления» и изучается в 1 и 2 семестрах.

Дисциплина «Программирование» является основополагающей для изучения дисциплины «Объектно-ориентированное программирование».

Особенностью дисциплины является изучение теоретического материала и приобретение практических навыков на языке программирования С++ с дополнительным изучением языка программирования С.

3. ПЛАНИРУЕМЫЕ РЕЗУЛЬТАТЫ ОБУЧЕНИЯ ПО ДИСЦИПЛИНЕ, СООТНЕСЕННЫЕ С ПЛАНИРУЕМЫМИ РЕЗУЛЬТАТАМИ ОСВОЕНИЯ ОСНОВНОЙ ОБРАЗОВАТЕЛЬНОЙ ПРОГРАММЫ

Процесс изучения дисциплины «Программирование» направлен на формирование следующих компетенций:

Формируемые компетенции по ФГОС		Основные показатели освоения программы дисциплины
Содержание компетенции	Код компетенции	
Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач	ОПК-2	ОПК-2.4. Уметь: анализировать профессиональные задачи, выбирать и использовать подходящие ИТ-решения ОПК-2.5. Владеть: навыками работы с лежащими в основе ИТ-решений данными ОПК-2.6. Владеть: навыками применения современных информационно-коммуникационных и интеллектуальных технологий, инструментальных сред, программно-технических платформ и программных средств, в том числе отечественного производства, для решения задач профессиональной деятельности

Формируемые компетенции по ФГОС		Основные показатели освоения программы дисциплины
Содержание компетенции	Код компетенции	
профессиональной деятельности		
Способен решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности	ОПК-3	ОПК-3.1. Знать: принципы, методы и средства решения стандартных задач профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности
Способен разрабатывать алгоритмы и программы, пригодные для практического применения	ОПК-8	<p>ОПК-8.1. Знать: процессы, методы поиска, сбора, хранения, обработки, предоставления, распространения информации и способы осуществления таких процессов и методов (информационные технологии)</p> <p>ОПК-8.2. Знать: логику построения и принципы функционирования современных языков программирования и языков работы с базами данных, сред разработки информационных систем и технологий, принципы разработки алгоритмов и компьютерных программ</p> <p>ОПК-8.3. Знать: современные языки программирования и языки работы с базами данных, среды разработки информационных систем и технологий</p> <p>ОПК-8.4. Уметь: выбирать языки программирования и языки работы с базами данных, среды разработки информационных систем и технологий, исходя из имеющихся задач</p> <p>ОПК-8.5. Уметь: применять современные языки программирования для разработки оригинальных алгоритмов и компьютерных программ, пригодных для практического применения, вести базы данных и информационные хранилища, применять современные программные среды разработки информационных систем и технологий;</p> <p>ОПК-8.6. Уметь: читать коды программных продуктов, написанных на освоенных языках программирования, и вносить требуемые изменения</p> <p>ОПК-8.7. Уметь: анализировать профессиональные задачи, разрабатывать подходящие ИТ-решения</p> <p>ОПК-8.8. Уметь: самостоятельно осваивать новые для себя современные языки программирования и языки работы с базами данных, среды разработки информационных систем и технологий</p>

Формируемые компетенции по ФГОС		Основные показатели освоения программы дисциплины
Содержание компетенции	Код компетенции	
Способен осваивать методики использования программных средств для решения практических задач	ОПК-9	ОПК-9.1. Знать: классификацию программных средств и возможности их применения для решения практических задач ОПК-9.2. Уметь: находить и анализировать техническую документацию по использованию программного средства, выбирать и использовать необходимые функции программных средств для решения конкретной задачи ОПК-9.3. Владеть: способами описания методики использования программного средства для решения конкретной задачи в виде документа, презентации или видеоролика
Способен разрабатывать компоненты программно-аппаратных комплексов, ПО, базы данных, используя современные инструментальные средства и технологии программирования, включая проектирование, отладку, проверку работоспособности и модификацию ПО	ПКС-5	ПКС-5.1. Знать: принципы, базовые концепции технологий программирования, основные этапы и принципы создания программного продукта, абстракция, различие между спецификацией и реализацией, рекурсия, конфиденциальность информации, повторное использование, проблема сложности, масштабирование, проектирование с учетом изменений, классификация, типизация, соглашения, обработка исключений, ошибки и отладка; основные положения теории баз данных, хранилищ данных, витрин данных, баз знаний, концептуальные, логические и физические модели данных ПКС-5.2. Уметь: устанавливать, тестировать, испытывать и использовать программные компоненты аппаратно-программных комплексов и баз данных, осуществлять их сертификацию по стандартам качества, разрабатывать информационно-логическую, функциональную и объектно-ориентированную модели информационной системы, модели данных информационных систем ПКС-5.3. Владеть: языками процедурного и объектно-ориентированного программирования, навыками владения одной из технологий программирования; методами и средствами представления данных и знаний о предметной области, методами и средствами анализа информационных систем, технологиями реализации, внедрения проекта аппаратно-программных комплексов и баз данных

4. СТРУКТУРА И СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

4.1. Объем дисциплины и виды учебной работы

Общая трудоёмкость дисциплины составляет 7 зачётных единиц или 252 часа.

Вид учебной работы	Всего ак. часов	Ак. часы по семестрам	
		1	2
Аудиторная работа, в том числе:	108	54	54
Лекции (Л)	36	18	18
Практические занятия (ПЗ)	72	36	36
Самостоятельная работа студентов (СРС), в том числе:	72	36	36
Выполнение курсового проекта	36	-	36
Подготовка к практическим занятиям	36	36	-
Промежуточная аттестация – зачет (З) / экзамен (Э)	72	36 (Э)	36 (Э)

Вид учебной работы	Всего ак. часов	Ак. часы по семестрам	
		1	2
Общая трудоёмкость час	252	126	126
зач. ед.	7	3.5	3.5

4.2. Содержание дисциплины

Учебным планом предусмотрены: лекции, практические занятия, самостоятельная работа, курсовой проект.

4.2.1. Разделы дисциплины и виды занятий

Наименование разделов	Всего ак. часов	Лекции	Практические занятия	Самостоятельная работа студента, в том числе курсовая работа (проект)
1-й семестр				
Раздел 1. Простые типы данных и конструкции языка	20	4	8	8
Раздел 2. Агрегатные типы данных	20	4	8	8
Раздел 3. Функции	30	6	12	12
Раздел 4. Символьные типы данных	20	4	8	8
Всего в семестре	90	18	36	36
2-й семестр				
Раздел 5. Классы	30	8	18	4
Раздел 6. Исключения	14	2	4	8
Раздел 7. Перегрузка операторов	18	2	4	12
Раздел 8. Стандартная библиотека STL. Шаблоны	28	6	10	12
Всего в семестре	90	18	36	36
Итого:	180	36	72	72
Подготовка к экзамену	72			
Всего:	252			

4.2.2. Содержание разделов дисциплины

№	Разделы	Содержание лекционных занятий	Трудоёмкость в ак. часах
1-й семестр			
1	Раздел 1	Простые типы данных и конструкции языка. Введение в программирование. Первая программа. Среда разработки. Компиляция. Сборка. Выполнение. Простые типы данных. Литералы. Числа целые, с плавающей точкой. Логический тип. Преобразование типов данных. Научный формат. Манипуляторы. Переменные. Ключевые слова. Комментарии.	4

№	Разделы	Содержание лекционных занятий	Трудоемкость в ак. часах
		Операции. Операторы умножения, деления, сложения, вычитания, деление по модулю, сравнения. Короткие операции. Унарный минус, Унарный плюс, Приоритет операций. Ассоциативность операций. Управляющие конструкции. Условный оператор if, else-if. Тернарный оператор. Оператор switch. Циклы while, do-while, for. Преобразование типов данных. Неявные, явные преобразования. Продвижение. Примеры анализа преобразований. Препроцессор. Директивы. Макросы. Работа с файлами заголовков.	
2	Раздел 2	Агрегатные данные. Массивы []. Инициализация массива. Операции над массивами. Многомерные массивы. Структуры (struct). Объявление переменной типа struct. Инициализация структур. Операции над структурами. Перечисление (enum). Объявление переменной типа enum. Инициализация перечислений. Операции над перечислениями. Объединение (union). Объявление переменной типа union. Инициализация объединений. Операции над объединениями. Указатели. Область применения указателей. Объявление указателей. Инициализация. Оператор sizeof. Указатели и массивы. Арифметика указателей. Динамическая память. Динамическая память new, delete, malloc, calloc, realloc, free. Массивы указателей. Динамические двумерные массивы.	4
3	Раздел 3	Функции. Глобальные переменные. Передача параметров по значению. Передача параметров по адресу. Передача параметров по ссылке. Параметры по умолчанию. функции inline. Перегрузка функций. Тип void. Рекурсивные функции. Организация, правило остановки. Достоинства, недостатки. Примеры Библиотеки. Библиотека math, работа с математическими функциями, Библиотека bits (работа с битовым типом данных). Таблицы истинности. Законы Де Моргана. Таблицы истинности битовых операций. Короткие битовые операции. Битовые операции: проверить состояние бита, сбросить бит, установить бит, изменить бит. Операции сдвига. Знаковый бит. Библиотека iostream (Ввод-вывод данных. Операторы cin, cout).	6
4	Раздел 4.	Символьные типы данных. Тип данных char. Объявление, инициализация. Таблица ASCII, литералы. escape-символы, операции над символами. Функции обработки символов. Тип данных string. Инициализация строки. Операции. Функция getline. Сравнение строк. Методы обработки строк – compare, substr, size, length, find, capacity, max_size, reserve, clear, resize, empty, Строки, как массивы, Функции – append, push_back, insert, assign, replace, erase, swap. Регулярные выражения. Пространство имен. Представление, определение, использование, расширение. Использование объекта. Пространство без имени. Переименование.	4
Итого по 1 семестру:			18
2-й семестр			
5	Раздел 5	Основы объектно-ориентированного программирования. Инкапсуляция, наследование, полиморфизм. Модификаторы доступа класса и при наследовании (private public, protected)	8

№	Разделы	Содержание лекционных занятий	Трудоемкость в ак. часах
		Внешнее и внутреннее определение методов. Конструктор. Деструктор. Обращение к методам через объект. Точки и стрелки. Вызов конструктора суперкласса. Геттеры и сеттеры. Создание объекта. Указатель this. Перегрузка методов и конструкторов. Копирование объектов. Утечка памяти. Массивы указателей на объекты. Статические компоненты (переменные, функции). Ключевое слово auto. Взаимодействие статических и динамических компонентов. Объекты внутри объектов. Применение форм и обработка событий (MFC, CLR).	
6	Раздел 6	Исключения. Описание проблемы. Деление на ноль. Контроль деления на ноль. Использование функции. Операторы try, catch, throw. Утечка памяти. Списки исключений в заголовке. Соответствие типов данных throw и catch. Спецификация throw. Распределение catch в программе. Иерархия catch. Иерархия классов исключений. Стандартные классы исключений – exception, logic_error, domain_error, invalid_argument, length_error, out_of_range, runtime_error, range_error, overflow_error, underflow_error. Исключение bad_alloc, bad_exception. Неопределенные исключения. Пользовательские исключения. Использование посредников. Обработка на разных уровнях.	2
7	Раздел 7	Перегрузка операторов. Реализация перегруженных операторы. Переопределение оператора индексации. Перегрузка операторов и тип enum. Арифметические операторы. Битовые операторы. Оператор присваивания. Реляционные операторы. Логические операторы. Операторы сложного присваивания. Префиксные операторы увеличения и уменьшения. Постфиксные операторы увеличения и уменьшения. Оператор []. Оператор (). Операторы указателя и ссылки. Операторы без перегрузки.	2
8	Раздел 8	Библиотека Standard Template Library (STL). Контейнеры vector, list, deque, set, map, multiset, multimap, stack, queue, priority_queue. Функции библиотек: algorithm, functional. Итераторы. Шаблоны. Шаблоны функций и классов.	6
Итого по 2 семестру:			18
Итого:			36

4.2.3. Практические занятия

№	Раздел	Темы практических занятий	Трудоёмкость в ак. часах
1-й семестр			
1	Раздел 1	Программирование с использованием простых типов данных и конструкций языка	8
2	Раздел 2	Программирование с использованием агрегатных типов данных	8
3	Раздел 3	Программирование с использованием функций	12
4	Раздел 4	Программирование с использованием символьных типов данных	8
Итого по 1 семестру:			36
2-й семестр			
5	Раздел 5	Программирование с использованием классов	18
6	Раздел 6	Программирование с использованием исключений	4
7	Раздел 7	Программирование с использованием перегрузки операторов	4
8	Раздел 8	Программирование с использованием стандартной библиотеки STL и шаблонов	10

№	Раздел	Темы практических занятий	Трудоёмкость в ак. часах
Итого по 2 семестру:			36
Итого:			72

4.2.4. Лабораторные работы

Лабораторные работы не предусмотрены

4.2.5. Курсовой проект

№	Тематика курсового проекта
1.	Разработка программного обеспечения с применением интеллектуальных технологий
2.	Разработка программного обеспечения с использованием интернет технологий

4.2.6. Самостоятельная работа

№	Раздел	Тематика Самостоятельной работы	Трудоёмкость в ак. часах
1-й семестр			
1	Раздел 1	Программирование с использованием простых типов данных и конструкций языка	8
2	Раздел 2	Программирование с использованием агрегатных типов данных	8
3	Раздел 3	Программирование с использованием функций	12
4	Раздел 4	Программирование с использованием символьных типов данных	8
Итого по 1 семестру:			36
2-й семестр			
5	Раздел 5-8	Разработка курсового проекта	36
Итого по 2 семестру:			36
Итого:			72

5. ОБРАЗОВАТЕЛЬНЫЕ ТЕХНОЛОГИИ

В ходе обучения применяются:

Лекции, которые являются одним из важнейших видов учебных занятий и составляют основу теоретической подготовки обучающихся. Цели лекционных занятий:

- дать систематизированные научные знания по дисциплине, акцентировать внимание на наиболее сложных вопросах дисциплины;
- стимулировать активную познавательную деятельность обучающихся, способствовать формированию их творческого мышления.

Практические занятия. Цели практических занятий:

- совершенствовать умения и навыки решения практических задач.

Главным содержанием этого вида учебных занятий является работа каждого обучающегося по овладению практическими умениями и навыками профессиональной деятельности.

Консультации (текущая консультация, накануне экзамена) является одной из форм руководства учебной работой обучающихся и оказания им помощи в самостоятельном изучении материала дисциплины, в ликвидации имеющихся пробелов в знаниях, задолженностей по текущим занятиям, в подготовке письменных работ (проектов).

Текущие консультации проводятся преподавателем, ведущим занятия в учебной группе, научным руководителем и носят как индивидуальный, так и групповой характер.

Самостоятельная работа обучающихся направлена на углубление и закрепление знаний, полученных на лекциях и других занятиях, выработку навыков самостоятельного активного приобретения новых, дополнительных знаний, подготовку к предстоящим учебным занятиям и промежуточному контролю.

Курсовое проектирование формирует навыки самостоятельного профессионального творчества.

6. ОЦЕНОЧНЫЕ СРЕДСТВА ДЛЯ ТЕКУЩЕГО КОНТРОЛЯ УСПЕВАЕМОСТИ, ПРОМЕЖУТОЧНОЙ АТТЕСТАЦИИ ПО ИТОГАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ

6.1. Оценочные средства для самостоятельной работы и текущего контроля успеваемости

Раздел 1. Простые типы данных и конструкции языка.

1. Введение в программирование. Первая программа. Среда разработки. Компиляция. Сборка. Выполнение.
2. Простые типы данных. Литералы. Числа целые, с плавающей точкой. Логический тип. Преобразование типов данных. Научный формат. Манипуляторы. Переменные. Ключевые слова. Комментарии.
3. Операции. Операторы умножения, деления, сложения, вычитания, деление по модулю, сравнения. Короткие операции. Унарный минус, Унарный плюс, Приоритет операций. Ассоциативность операций.
4. Управляющие конструкции. Условный оператор if, else-if. Тернарный оператор. Оператор switch. Циклы while, do-while, for.
5. Преобразование типов данных. Неявные, явные преобразования. Продвижение. Примеры анализа преобразований.
6. Препроцессор. Директивы. Макросы. Работа с файлами заголовков.

Раздел 2. Агрегатные данные.

1. Массивы []. Инициализация массива. Операции над массивами. Многомерные массивы.
2. Перечисление (enum). Объявление переменной типа enum. Инициализация перечислений. Операции над перечислениями.
3. Объединение (union). Объявление переменной типа union. Инициализация объединений. Операции над объединениями.
4. Указатели. Область применения указателей. Объявление указателей. Инициализация. Оператор sizeof. Указатели и массивы. Арифметика указателей. Динамическая память.
5. Динамическая память new, delete, malloc, calloc, realloc, free. Массивы указателей. Динамические двумерные массивы.

Раздел 3. Функции.

1. Глобальные переменные. Передача параметров по значению. Передача параметров по адресу. Передача параметров по ссылке. Параметры по умолчанию. функции inline. Перегрузка функций. Тип void.
2. Рекурсивные функции. Организация, правило остановки. Достоинства, недостатки. Примеры.
3. Библиотеки. Библиотека math, работа с математическими функциями,
4. Библиотека bits (работа с битовым типом данных). Таблицы истинности. Законы Де Моргана. Таблицы истинности битовых операций. Короткие битовые операции. Битовые операции: проверить состояние бита, сбросить бит, установить бит, изменить бит. Операции сдвига. Знаковый бит.
5. Библиотека iostream (Ввод-вывод данных. Операторы cin, cout).

Раздел 4. Символьные типы данных.

1. Тип данных char. Объявление, инициализация. Таблица ASCII, литералы. escape-символы, операции над символами. Функции обработки символов.
2. Тип данных string. Инициализация строки. Операции. Функция getline. Сравнение строк. Методы обработки строк – compare, substr, size, length, find, capacity, max_size, reserve, clear, resize, empty.
3. Строки, как массивы, Функции – append, push_back, insert, assign, replace, erase, swap. Регулярные выражения.
4. Пространство имен. Представление, определение, использование, расширение. Использование объекта. Пространство без имени. Переименование..

Раздел 5. Классы.

1. **Объектно-ориентированное программирование (ООП).** Преимущества ООП. Инкапсуляция. Внешнее и внутреннее определение методов. Конструктор. Обращение к методам через объект. Точки и стрелки. Наследование. Полиморфизм. Вызов конструктора суперкласса. Геттеры и сеттеры. Создание объекта Указатель this.
2. Перегрузка функций. Перегрузка конструкторов.
3. Копирование объектов. Утечка памяти. Деструкторы. Статические компоненты. Ключевое слово «auto».
4. Экземпляры класса. Статические компоненты класса. Статические переменные. Статические функции. Взаимодействие статических и нестатических компонентов. Динамические объекты.
5. Массивы указателей на объекты. Объекты внутри объектов. Формы. Обработка событий. Стек. Односвязный и двухсвязный список. Рекурсия. Введение в разработку форм и обработку событий.

Раздел 6. Исключения.

1. **Исключения.** Преимущества применения исключений. Класс exception. Операторы – try, catch, throw. Иерархия классов исключений.
2. Ключевое слово explicit.
3. Классы – exception, logic_error, domain_error, invalid_argument, length_error, out_of_range, runtime_error, range_error, overflow_error, underflow_error.
4. Расширение классов, Исключения - bad_alloc, bad_exception.

Раздел 7. Перегрузка операторов.

1. **Перегрузка операторов.** Перегрузка операторов на примере стека (операторы для методов push, pop, Цепочки переменных <<, >>). Реализация перегруженных операторы, при недоступном исходном коде.
2. Переопределение оператора индексации. Перегрузка операторов и тип enum. Арифметические операторы + - * /%. Битовые операторы ^ | & ~ << >>. Оператор присваивания.
3. Реляционные операторы ==! ==> = <<=. Логические операторы ! && ||.
4. Операторы сложного присваивания += -= *= %= /= &= |= ^= ==> = <<=.
5. Префиксные операторы увеличения и уменьшения ++ -.
6. Постфиксные операторы увеличения и уменьшения ++ -.
7. Оператор квадратные скобки - [].
8. Оператор круглые скобки - (). Операторы указателя & *.
9. Операторы без перегрузки. Шаблоны.

Раздел 8. Стандартная библиотека STL. Шаблоны.

1. **Библиотека Standard Template Library (STL).**
2. Контейнеры vector, list, deque, set, map, multiset, multimap, stack, queue, priority_queue.
3. Функции библиотек: algorithm, functional.
4. Итераторы.

6.2. Оценочные средства для проведения промежуточной аттестации

6.2.1. Примерный перечень вопросов к экзамену

1. Литералы. Переменные. Ключевые слова. Комментарии.
2. Операторы (умножение, деление, сложение, вычитание, унарный минус, унарный плюс, деление по модулю, сравнение, короткие операции). Приоритет и ассоциативность операций.
3. Символьный тип. Таблица ASCII, escape-символы, операции над символами. Целые числа. Числа с плавающей точкой. Логический тип. Преобразование типов данных.
4. Ввод-вывод (консоль, файлы). Научный формат. Манипуляторы.
5. Условный оператор (if, if-else) Вложенные условия. Каскадные условия. Тернарный оператор условия. Оператор switch. Циклы (while, do-while, for).
6. Массивы. Инициализация массива. Многомерные массивы. Операции над массивами.

7. Структуры (struct). Объявление переменной типа struct. Инициализация структур. Операции над структурами.
8. Перечисление (enum). Объявление переменной типа enum. Инициализация перечислений. Операции над перечислениями.
9. Объединение (union). Объявление переменной типа union. Инициализация объединений. Операции над объединениями.
10. Логические операции.
11. Работа с битами. Таблицы истинности битовых операций. Короткие битовые операции.
12. Битовые операции: проверка состояния бита, сброс бит, установка бита, изменение бита. Операции сдвига. Знаковый бит.
13. Указатели. Область применения указателей. Объявление указателей. Инициализация.
14. Оператор sizeof.
15. Указатели и массивы.
16. Арифметика указателей. Динамическая память.
17. Динамическое выделение памяти (new, delete).
18. Массивы указателей. Динамические двумерные массивы.
19. Функции. Глобальные переменные.
20. Передача параметров по значению.
21. Передача параметров по адресу.
22. Передача параметров по ссылке.
23. Параметры по умолчанию.
24. Рекурсия. Функции inline. Перегрузка функций. Тип void.
25. Тип данных string. Инициализация строки. Операция конкатенации. Короткие операции. Функция getline. Сравнение строк.
26. Функции – compare, substr, size, length, find, capacity, max_size, reserve, resize, empty.
27. Строки, как массивы. Методы – append, push_back, insert, assign, replace, erase, swap.
28. Преобразование типов данных. Неявные преобразования. Явные преобразования. Разбор случаев преобразования типов. Целочисленные типы. Значения с плавающей точкой.
29. Продвижение. Пример анализа преобразований.
30. Пространство имен. Представление пространства имен. Определение пространства имен.
31. Использование пространства имен. Расширение пространства имен. Пространство без имени. Переименование пространства имен.
32. ООП. Инкапсуляция. Внешнее и внутреннее определение методов. Конструктор. Обращение к методам через объект. Точки и стрелки. Наследование. Полиморфизм. Вызов конструктора суперкласса. Геттеры и сеттеры. Создание объекта Указатель this. Перегрузка функций.
33. Перегрузка конструкторов. Копирование объектов. Утечка памяти. Деструкторы.
34. Статические компоненты. Ключевое слово auto. Экземпляры класса. Статические компоненты класса.
35. Динамические Массивы указателей на объекты. Объекты внутри объектов. Стек.
36. Односвязный и двухсвязный список. Введение в разработку форм и обработку событий.
37. Технология исключений. Класс exception.
38. Операторы – try, catch, throw. Иерархия классов исключений.
39. Ключевое слово explicit. Классы – exception, logic_error, domain_error, invalid_argument, length_error, out_of_range, runtime_error, range_error, overflow_error, underflow_error. Расширение классов,
40. Исключения - bad_alloc, bad_exception.

6.2.2. Примерные тестовые задания к экзамену

Вариант № 1

№	Вопрос	Варианты ответа
1.	Автор языка С...	1. Кен Томпсон 2. Денис Ритчи 3. Бил Гейтс 4. Джеймс Гослинг
2.	Автор языка С++...	1. Расмус Лердорф 2. Бьерн Страуструп 3. Тим Бернерс-Ли 4. Джеймс Гослинг
3.	Назначение технологии - пространство имен.	1. объявление классов и имен 2. устранение длинных имен 3. указание видимости имен 4. устранение конфликтов имен
4.	Пространство имен задается с помощью ключевого слова ...	1. space 2. system 3. blank 4. namespace
5.	Каким символом должен заканчиваться оператор?	1. символом - точка с запятой 2. символом - точка 3. символом - запятая 4. символом - двоеточие.
6.	Правильный комментарий -	1. {comment} 2. /* comment */ 3. *comment* 4. [*comment]*
7.	С какого места начинает выполняться программа на Си++?	1. #include 2. do 3. class 4. main ()
8.	Дробные числа в С++ обозначаются ...	1.float 2.short 3.char 4.bool
9.	Укажите правильный идентификатор.	1. FA_Ф12; 2. _aaa; 3. int; 4. перем12.
10.	Какая операция в С++ обозначается знаком «%»?	1. вычисление процента; 2. вычисление остатка от деления; 3. целочисленное деление; 4. нет такого знака.
11.	Что выполняется в первую очередь при компоновке программы?	1. функция main; 2. приведение типов; 3. директивы define, include; 4. выделение памяти.
12.	Сколько функций может быть в программе на С++?	1. ни одной 2. как минимум одна 3. 100 4. 256
13.	С помощью какой директивы происходит подключение других модулей программы?	1. using namespace std 2. #define 3. #ifndef 4. #include
14.	Не относится к логическим операциям -	1. && 2. = 3.

№	Вопрос	Варианты ответа
		4. !
15.	Для получения адреса переменной используется операция ...	1. *; 2. &; 3. &&; 4. нет правильного ответа.
16.	Строка: 12_Angry_Men Это - ...	1. Не переменная 2. Переменная 3. Это константа 4. Это макрос
17.	C++ программа может содержать	1. Только одну функцию main 2. Ни одной функции main 3. Любое количество функций main 4. Две функции main
18.	Число, записанное как 007 означает	1. Целое в шестнадцатеричное 2. Целое в восьмеричное 3. Целое в десятичное 4. Целое в двоичное
19.	Какого значение следующего литерала 0E1	1. 0.01 2. Ошибочный литерал. 3. 0 4. 0.1
20.	Какого значение следующего литерала -1e-1	1. Ошибочный литерал. 2. -0.1 3. -1.0 4. -10.0

Вариант № 2

№	Вопрос	Варианты ответа
1.	Чему равен результат вычисления выражения $x + 3 * b / 2$ при $x = 12$ и $b = 8$?	1. 20; 2. 24; 3. 60; 4. 120.
2.	Какой результат получится int var; var = 10; /* var = var + 10; */ var = var + var; // var = var + var;	1. 20 2. 10 3. 40 4. 80
3.	Какой результат получится float var; var = .1; var = var + 1.0; var = var + 1e1; var = var + 1e-2;	1. 111.1 2. 1.111 3. 11.11 4. 0.1111
4.	Какое значение примет переменная k после выполнения кода: int i, j, k; i = 10; j = 3; k = (i % j * i / 3) / (j % i - j / i);	1. 1 2. 2.5 3. 2 4. 1.5
5.	Чему равно значение целочисленной переменной при вычислении выражения $21/5*3$?	1. 13.02; 2. 1.47; 3. 12; 4. 1.
6.	Какое значение примет переменная y? $y = (5 > 3) ? 1 : 2;$	1. 4; 2. 3; 3. 2; 4. 1.
7.	Какое значение примет переменная x? $x = (5 > = 5) ? 1 : 2;$	1. 4; 2. 3;

№	Вопрос	Варианты ответа
		3. 2; 4. 1.
8.	Какое значение примет переменная z? z = (4 < 3)? 1:2;	1. 1; 2. 2; 3. 3; 4. 4.
9.	Сколько раз будет выполняться операция 1? for (i=0; i<=9; i++) { операция 1; }	1.10; 2. 9; 3. 1; 4. ни разу.
10.	Сколько раз будет выполняться операция 1? for (i=3; i<9; i=i+3) { операция 1; }	1. ни разу; 2. 1; 3. 2; 4. 3.
11.	Сколько раз будет выполняться операция 1? for (i=3; i<10; i=i+3) {операция 1; }	1. ни разу; 2. 1; 3. 2; 4. 3.
12.	Сколько раз выполнится тело цикла? int i=0; while (i<9) { операция 1; i++;}	1. 8; 2. 9; 3. 10; 4. ни разу.
13.	Сколько раз выполнится тело цикла? int i=9; while (i<9) { операция 1; i++;}	1. 8; 2. 9; 3. 10; 4. ни разу.
14.	Сколько раз выполнится тело цикла? int i=0; while (i) { операция 1; i++;}	1. 8; 2. 9; 3. 10; 4. ни разу.
15.	Укажите результат после выполнения фрагмента программы. char operation = '+' ; int c=2, d=3, result; switch (operation) { case '+': result=c+d; break; case '-': result=c-d; break; default: result=2; }	1. 1; 2. 2; 3. 5; 4. нет правильного ответа.
16.	Укажите результат после выполнения фрагмента программы. char operation = '*' ; int c=2, d=1, result; switch (operation) { case '+': result=c+d; break; case '-': result=c-d; break; default: result=5; }	1. 1; 2. 2; 3. 5; 4. нет правильного ответа.
17.	Что выведет код? char operation = '-' ; int c=2, d=1, result; switch (operation) { case '-': result=c+d; break; case '+': result=c-d; break; default: result=2; }	1. 1; 2. 2; 3. 3; 4. нет правильного ответа.
18.	Что выведет код, если ввести значение 3.0 float a, b; cin >> a;	1.10.5 2.4.5

№	Вопрос	Варианты ответа
	<pre>b = a / 2; if (b >= 1) a += b; if (a > b) b += a; cout <<a + b;</pre>	3.8.5 4.Невозможно определить.
19.	<p>Что выведет код, если ввести значение 2.0?</p> <pre>float a; cin >> a; a *= a; a *= a; a = sqrtf(a) * sqrtf(a); cout <<a;</pre>	1.16 2.8 3.32 4.4
20.	<p>Что выведет код, если ввести значение 10?</p> <pre>int i,j; cin >> i; j = i; if (i >= 9) i += 9 - i; j /= i; cout << j;</pre>	1.2 2.Невозможно определить 3.0 4.1

Вариант № 3

№	Вопрос	Варианты ответа
1.	Укажите стандартный поток ввода данных?	1.cin; 2. cout; 3. cerr; 4. std.
2.	Укажите стандартный поток ввода данных?	1.cin; 2. cout; 3. cerr; 4. std.
3.	Укажите оператор ввода данных?	1.cin; 2. cout; 3. >> ; 4. << .
4.	Укажите оператор вывода данных?	1.cin; 2. cout; 3. << ; 4. >> .
5.	Если int a, b ; то будет ли выполнена инструкция? cout <<a, b;	1.да; 2.да, в некоторых средах программирования; 3.в зависимости от компилятора; 4. нет.
6.	В результате выполнения оператора cout <<hex <<100; на экран будет выведено...	1. hex; 2. hex100; 3. 100; 4. 64.
7.	Сколько параметров у функции int buf (int a)?	1. нет; 2. 1; 3. 2; 4. 3.
8.	Переменные внутри функции называются...	1.локальные; 2.встроенные; 3.глобальные; 4.функциональные.
9.	Укажите правильный вызов функции.	1.int beta(); 2.beta(char a); 3.beta(7); 4.void beta(7);
10.	Возвращаемым значением у функции не может	1.переменная;

№	Вопрос	Варианты ответа
	быть...	2.массив; 3.указатель; 4.нет правильного ответа.
11.	Функция не может быть ...	1.объявлена; 2.закрыта; 3.определена; 4.вызвана.
12.	Функция может возвращать ...	1.одно значение; 2.два значения; 3.три значения; 4.много значений.
13.	Оператор return применяется для ...	1.возврат из цикла; 2.обмен данными между функциями; 3.выход из оператора switch; 4.нет правильного ответа.
14.	Что выведет код? int t[4] = { 8, 4, 2, 1 }; int *p1 = t + 2, *p2 = p1 - 1; p1++; cout << *p1 - t[p1 - p2] << endl;	1.2 2.-2 3.-1 4.1
15.	Что выведет код? char f1(char c) { return c == 'z' ? 'a' : c + 1; } char f2(char &c) { c = f1(c); return c; } int main(void) { char x = 'x'; cout << f2(x); cout << f2(x); cout << f2(x) << endl; return 0;}	1.XYZ 2.YZA 3.yza 4.xyz
16.	Что выведет код? #include <iostream> using namespace std; int *fun(void) { return new int[2]; } int fun(int *p) { delete [] p; return 0; } void fun(int *p, int q) { p[q] *= 2; } void fun(int *p, int q, int r) { p[q] = r; } int main(void) { int *v = fun(); fun(v,0,1); fun(v,1,2); fun(v,0); cout << v[1] + v[0] << endl; fun(v); return 0;}	1.2 2.3 3.4 4.1
17.	Что выведет код? #include <iostream> using namespace std; int fun(void) { return 1; } int fun(int p) { return p > 1 ? 1 : 0; } int fun(int p, int q) { return q > p ? q-p : p-q; } int main(void) { cout << fun() + fun(1) + fun(1,2) << endl; return 0;}	1.9 2.2 3.4 4.5
18.	Что выведет код? #include <iostream> using namespace std; int fun(void) { return 1; } int fun(int p) { return 1 + p; }	1.12 2.10 3.6 4.8

№	Вопрос	Варианты ответа
	<pre>int fun(int p, int q) { return 2 + p + q; } int main(void) { cout << fun() + fun(1) + fun(1,2) << endl; return 0;}</pre>	
19.	<p>Что выведет код?</p> <pre>#include <iostream> using namespace std; int fun(int p = 2, int q = 3) { return p + q;} int main(void) { cout << fun() + fun(1) + fun(1,2) << endl; return 0;}</pre>	<p>1.6 2.12 3.8 4.10</p>
20.	<p>Что выведет код?</p> <pre>#include <iostream> using namespace std; int fun1(int p) { ++p; return p++;} int fun2(int &p) { ++p; return p++;} int main(void) { int a = 1, b, c; b = fun1(a); c = fun2(b); cout << a + b + c << endl; return 0;}</pre>	<p>1.8 2.4 3.6 4.10</p>

6.3. Описание показателей и критериев контроля успеваемости, описание шкал оценивания

6.3.1. Критерии оценок промежуточной аттестации (экзамен)

Оценка			
«2» (неудовлетворительно)	Пороговый уровень освоения	Углубленный уровень освоения	Продвинутый уровень освоения
	«3» (удовлетворительно)	«4» (хорошо)	«5» (отлично)
Студент не знает значительной части материала, допускает существенные ошибки в ответах на вопросы	Студент поверхностно знает материал основных разделов и тем учебной дисциплины, допускает неточности в ответе на вопрос	Студент хорошо знает материал, грамотно и по существу излагает его, допуская некоторые неточности в ответе на вопрос.	Студент в полном объеме знает материал, грамотно и по существу излагает его, не допуская существенных неточностей в ответе на вопрос
Не умеет находить решения большинства предусмотренных программой обучения заданий	Иногда находит решения, предусмотренные программой обучения заданий	Уверенно находит решения, предусмотренные программой обучения заданий	Безошибочно находит решения, предусмотренные программой обучения заданий
Большинство предусмотренных программой обучения заданий не выполнено	Предусмотренные программой обучения задания выполнены удовлетворительно	Предусмотренные программой обучения задания успешно выполнены	Предусмотренные программой обучения задания успешно выполнены

Примерная шкала оценивания знаний в тестовой форме:

Количество правильных ответов, %	Оценка
0-49	Неудовлетворительно
50-65	Удовлетворительно
66-85	Хорошо
86-100	Отлично

6.3.2. Описание шкалы и критериев оценивания для проведения промежуточной аттестации в форме защиты курсового проекта

Студент выполняет курсовой проект в соответствии с графиком, принятым на заседании кафедры. Оценка может быть снижена за несоблюдение установленного кафедрой графика.

Оценка			
«2» (неудовлетворительно)	Пороговый уровень освоения	Углубленный уровень освоения	Продвинутый уровень освоения
	«3» (удовлетворительно)	«4» (хорошо)	«5» (отлично)
Студент не выполнил курсовой проект в соответствии с заданием. Не владеет теоретическими знаниями по изучаемой дисциплине. Необходимые практические компетенции не сформированы	Студент выполнил курсовой проект с существенными ошибками. При защите курсового проекта демонстрирует слабую теоретическую подготовку. При решении задач, предусмотренных программой учебной дисциплины, допускает неточности, существенные ошибки	Студент выполнил курсовой проект с некоторыми незначительными ошибками и неточностями. При защите курсового проекта демонстрирует хорошую теоретическую подготовку. Хорошо справляется с решением задач, предусмотренных программой учебной дисциплины	Студент выполнил курсовой проект полностью в соответствии с заданием. При защите курсового проекта демонстрирует высокую теоретическую подготовку. Успешно справляется с решением задач, предусмотренных программой учебной дисциплины

7. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ

7.1. Рекомендуемая литература

7.1.1. Основная литература

1. Шилдт, Герберт «С++: базовый курс, 3-е издание» Пер. с англ. — М.: Издательский дом “Вильямс”, 2016. — 624 с. URL: https://www.bsuir.by/m/12_119786_1_98220.pdf

7.1.2. Дополнительная литература

1. Государственные стандарты: ГОСТ 19781–90, ГОСТ Р ИСО/МЭК 12207-99, ГОСТ Р ИСО/МЭК 14764-2002, ГОСТ 34.601-90.

7.1.3. Учебно-методическое обеспечение

1. Полный справочник по С++, 4-е издание. . Пер. с англ. — М. : Издательский дом “Вильямс”, 2006. — 800 с. — URL: https://codernet.ru/books/c_plus/polnyj_spravochnik_po_c_4-e_izdanie/

7.2. Базы данных, электронно-библиотечные системы, информационно-справочные и поисковые системы

1. Европейская цифровая библиотека Europeana: <http://www.europeana.eu/portal>
2. Консультант Плюс: справочно-поисковая система - www.consultant.ru
3. Информационно-издательский центр "ГЕОИНФОРММАРК": <http://www.geoinform.ru/>
4. Информационно-аналитический центр «Минерал»: <http://www.mineral.ru/>
5. Мировая цифровая библиотека: <http://wdl.org/ru>
6. Научная электронная библиотека «Scopus»: <https://www.scopus.com>
7. Научная электронная библиотека ScienceDirect: <http://www.sciencedirect.com>
8. Научная электронная библиотека «eLIBRARY»: <https://elibrary.ru/>
9. Портал «Гуманитарное образование» <http://www.humanities.edu.ru/>
10. Федеральный портал «Российское образование» <http://www.edu.ru/>

11. «Единая коллекция цифровых образовательных ресурсов» <http://school-collection.edu.ru/>
12. Поисковые системы Yandex, Rambler, Yahoo и др.
13. Электронно-библиотечная система «Лань»: <https://e.lanbook.com/books>
14. Электронная библиотека РГБ: <http://elibrary.rsl.ru/>
15. Электронная библиотека учебников: <http://studentam.net>
16. Электронно-библиотечная система «ЭБС ЮРАЙТ»: www.biblio-online.ru
17. Национальный цифровой ресурс «Руконт»: <http://rucont.ru/>
18. Электронно-библиотечная система <http://www.sciteclibrary.ru/>

8. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

8.1. Материально-техническое оснащение аудиторий

Специализированные аудитории, используемые при проведении занятий лекционного типа, оснащены мультимедийными проекторами и комплектом аппаратуры, позволяющей демонстрировать текстовые и графические материалы.

Для проведения практических занятий используются компьютерные классы, оборудованные техникой из расчета один компьютер на одного обучающегося, с обустроенным рабочим местом преподавателя. В учебном процессе используется комплект демонстрационных стендовых материалов по темам курса.

Аудитории для проведения лекционных занятий

56 посадочных мест. Оснащенность: Мультимедийный проектор – 1 шт., проекционная аппаратура: источник бесперебойного питания – 1 шт., экран – 1 шт., ноутбук – 1 шт., (возможность доступа к сети «Интернет»), стойка мобильная – 1 шт., стул для студентов – 56 шт., кресло преподавателя – 1 шт., стол – 29 шт., переносная настольная трибуна – 1 шт., доска настенная магнитно-маркерная – 1 шт., плакат в рамке настенный – 5 шт.

Перечень лицензионного программного обеспечения: Microsoft Windows 8 Professional, Microsoft Office 2007 Professional Plus, CorelDRAW Graphics Suite X5, Autodesk product: Building Design Suite Ultimate 2016, product Key: 766H1, антивирусное программное обеспечение: Kaspersky Endpoint Security, 7-zip (свободно распространяемое ПО), Foxit Reader (свободно распространяемое ПО), SeaMonkey (свободно распространяемое ПО), Chromium (свободно распространяемое ПО), Java Runtime Environment (свободно распространяемое ПО), doPDF (свободно распространяемое ПО), GNU Image Manipulation Program (свободно распространяемое ПО), Inkscape (свободно распространяемое ПО), XnView (свободно распространяемое ПО), K-Lite Codec Pack (свободно распространяемое ПО), FAR Manager (свободно распространяемое ПО).

Аудитории для проведения практических занятий *16 посадочных мест*

Оснащенность: Стол компьютерный для студентов (тип 4) - 3 шт., стол компьютерный для студентов (тип 6) - 2 шт., стол компьютерный для студентов (тип 7) - 1 шт., кресло преподавателя (сетка, цвет черный) - 17 шт., доска напольная мобильная белая магнитно-маркерная «Magnetoplan» 1800мм×1200мм - 1 шт., моноблок Lenovo M93Z Intel Q87 - 17 шт., (возможность доступа к сети «Интернет»), плакат - 5 шт.

Перечень лицензионного программного обеспечения: Microsoft Windows 7 Professional, Microsoft Office 2007 Professional Plus; CorelDRAW Graphics Suite X5, Autodesk product: Building Design Suite Ultimate 2016, product Key: 766H1, антивирусное программное обеспечение: Kaspersky Endpoint Security, 7-zip (свободно распространяемое ПО), Foxit Reader (свободно распространяемое ПО), SeaMonkey (свободно распространяемое ПО), Chromium (свободно распространяемое ПО), Java Runtime Environment (свободно распространяемое ПО), doPDF (свободно распространяемое ПО), GNU Image Manipulation Program (свободно распространяемое ПО), Inkscape (свободно распространяемое ПО), XnView (свободно распространяемое ПО), K-Lite Codec Pack (свободно распространяемое ПО), FAR Manager (свободно распространяемое ПО), Cisco Packet Tracer 7.1 (свободно распространяемое ПО), Quantum GIS (свободно распространяемое ПО), Python (свободно распространяемое ПО), R (свободно распространяемое ПО), Rstudio (свободно распространяемое ПО), SMath Studio (свободно распространяемое ПО), GNU Octave (свободно распространяемое ПО), Scilab (свободно распространяемое ПО).

8.2. Помещение для самостоятельной работы

1. Оснащенность помещения для самостоятельной работы: 16 посадочных мест. Стол компьютерный для студентов (тип 4) - 3 шт., стол компьютерный для студентов (тип 6) - 2 шт., стол компьютерный для студентов (тип 7) - 1 шт., кресло преподавателя (сетка, цвет черный) - 17 шт., доска напольная мобильная белая магнитно-маркерная «Magnetoplan» 1800мм×1200мм - 1 шт., моноблок Lenovo M93Z Intel Q87 - 17 шт., плакат - 5 шт. Доступ к сети «Интернет», в электронную информационно-образовательную среду Университета.

Перечень лицензионного программного обеспечения: Microsoft Windows 7 Professional; Microsoft Office 2007 Professional Plus; CorelDRAW Graphics Suite X5, Autodesk product: Building Design Suite Ultimate 2016, product Key: 766H1, антивирусное программное обеспечение: Kaspersky Endpoint Security, 7-zip (свободно распространяемое ПО), Foxit Reader (свободно распространяемое ПО), SeaMonkey (свободно распространяемое ПО), Chromium (свободно распространяемое ПО), Java Runtime Environment (свободно распространяемое ПО), doPDF (свободно распространяемое ПО), GNU Image Manipulation Program (свободно распространяемое ПО), Inkscape (свободно распространяемое ПО), XnView (свободно распространяемое ПО), K-Lite Codec Pack (свободно распространяемое ПО), FAR Manager (свободно распространяемое ПО), Cisco Packet Tracer 7.1 (свободно распространяемое ПО), Quantum GIS (свободно распространяемое ПО), Python (свободно распространяемое ПО), R (свободно распространяемое ПО), Rstudio (свободно распространяемое ПО), SMath Studio (свободно распространяемое ПО), GNU Octave (свободно распространяемое ПО), Scilab (свободно распространяемое ПО).

8.3. Помещения для хранения и профилактического обслуживания

Оснащенность: персональный компьютер - 2 шт. (доступ к сети «Интернет»), монитор - 4 шт., сетевой накопитель - 1 шт., источник бесперебойного питания - 2 шт., телевизор плазменный Panasonic - 1 шт., точка Wi-Fi - 1 шт., паяльная станция - 2 шт., дрель - 5 шт., перфоратор - 3 шт., набор инструмента - 4 шт., тестер компьютерной сети - 3 шт., баллон со сжатым газом - 1 шт., паста теплопроводная - 1 шт., пылесос - 1 шт., радиостанция - 2 шт., стол - 4 шт., тумба на колесиках - 1 шт., подставка на колесиках - 1 шт., шкаф - 5 шт., кресло - 2 шт., лестница Alve - 1 шт.

Перечень лицензионного программного обеспечения: Microsoft Windows 7 Professional, Microsoft Office 2010 Professional Plus, антивирусное программное обеспечение: Kaspersky Endpoint Security, 7-zip (свободно распространяемое ПО), Foxit Reader (свободно распространяемое ПО), SeaMonkey (свободно распространяемое ПО), Chromium (свободно распространяемое ПО), Java Runtime Environment (свободно распространяемое ПО), doPDF (свободно распространяемое ПО), GNU Image Manipulation Program (свободно распространяемое ПО), Inkscape (свободно распространяемое ПО), XnView (свободно распространяемое ПО), K-Lite Codec Pack (свободно распространяемое ПО), FAR Manager (свободно распространяемое ПО).

8.4. Лицензионное программное обеспечение

1. Microsoft Windows 7 Professional.
2. Microsoft Windows 8 Professional.
3. Microsoft Office 2007 Professional Plus.

ЛИСТ АКТУАЛИЗАЦИИ РАБОЧЕЙ ПРОГРАММЫ ДИСЦИПЛИНЫ

Рабочая программа дисциплины рассмотрена и переутверждена на заседании кафедры ИСиВТ от 25.01.2021 года, протокол № 7.

Заведующий кафедрой _____ к.т.н., доцент Е.Б. Мазаков

Рабочая программа дисциплины рассмотрена и переутверждена на заседании кафедры ИСиВТ от _____ года, протокол № _____.

Заведующий кафедрой _____ к.т.н., доцент Е.Б. Мазаков

Рабочая программа дисциплины рассмотрена и переутверждена на заседании кафедры ИСиВТ от _____ года, протокол № _____.

Заведующий кафедрой _____ к.т.н., доцент Е.Б. Мазаков

Рабочая программа дисциплины рассмотрена и переутверждена на заседании кафедры ИСиВТ от _____ года, протокол № _____.

Заведующий кафедрой _____ к.т.н., доцент Е.Б. Мазаков

Рабочая программа дисциплины рассмотрена и переутверждена на заседании кафедры ИСиВТ от _____ года, протокол № _____.

Заведующий кафедрой _____ к.т.н., доцент Е.Б. Мазаков