

Документ подписан простой электронной подписью  
Информация о владельце:  
ФИО: Галунин Сергей Александрович  
Должность: проректор по учебной работе  
Дата подписания: 23.12.2025 13:42:26  
Уникальный программный ключ:  
08ef34338325bdb0ac5a47baa5472ce36cc3fc3b

Приложение к ОПОП  
«Разработка программно-  
информационных систем»



**СПбГЭТУ «ЛЭТИ»**  
ПЕРВЫЙ ЭЛЕКТРОТЕХНИЧЕСКИЙ

МИНОБРНАУКИ РОССИИ

федеральное государственное автономное образовательное учреждение высшего образования  
**«Санкт-Петербургский государственный электротехнический университет  
«ЛЭТИ» им. В.И.Ульянова (Ленина)»  
(СПбГЭТУ «ЛЭТИ»)**

---

**РАБОЧАЯ ПРОГРАММА**

дисциплины

**«ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ»**

для подготовки бакалавров

по направлению

09.03.04 «Программная инженерия»

по профилю

**«Разработка программно-информационных систем»**

Санкт-Петербург

2025

## **ЛИСТ СОГЛАСОВАНИЯ**

Разработчики:

к.т.н., доцент Кринкин К.В.

ассистент Жангиров Т.Р.

Рабочая программа рассмотрена и одобрена на заседании кафедры МОЭВМ  
20.01.2025, протокол № 1

Рабочая программа рассмотрена и одобрена учебно-методической комиссией  
ФКТИ, 28.01.2025, протокол № 1

Согласовано в ИС ИОТ

Начальник ОМОЛА Загороднюк О.В.

## **1 СТРУКТУРА ДИСЦИПЛИНЫ**

Обеспечивающий факультет	ФКТИ
Обеспечивающая кафедра	МОЭВМ
Общая трудоемкость (ЗЕТ)	5
Курс	2
Семестр	3

### **Виды занятий**

Лекции (академ. часов)	34
Лабораторные занятия (академ. часов)	34
Иная контактная работа (академ. часов)	1
Все контактные часы (академ. часов)	69
Самостоятельная работа, включая часы на контроль (академ. часов)	111
Всего (академ. часов)	180

### **Вид промежуточной аттестации**

Экзамен (курс)	2
----------------	---

## **2 АННОТАЦИЯ ДИСЦИПЛИНЫ**

### **«ОБЪЕКТНО-ОРИЕНТИРОВАННОЕ ПРОГРАММИРОВАНИЕ»**

Дисциплина предназначена для приобретения студентами теоретических знаний и практических навыков профессионального программирования на основе объектно-ориентированного представления сущностей предметной области, задаваемой решаемой задачей. Рассматриваются основные элементы объектно-ориентированного программирования – декомпозиция задачи на объекты, инкапсуляция внутреннего состояния и поведения объекта, описываемого классом, построение иерархии классов, полиморфизм, простое и множественное наследование, параметрический полиморфизм, механизм обработки исключений.

#### **SUBJECT SUMMARY**

#### **«OBJECT-ORIENTED PROGRAMMING»**

Discipline is designed for students of theoretical knowledge and practical skills for professional programming based on object-oriented representation of the domain entity specified by current task. Discusses the fundamentals of object-oriented programming (OOP) is a decomposition of the task objects encapsulating the internal State and object behaviors described by the class build a hierarchy of classes, polymorphism, simple and multiple inheritance, parametric polymorphism, exception handling mechanism.

## **3 ОБЩИЕ ПОЛОЖЕНИЯ**

### **3.1 Цели и задачи дисциплины**

1. Целью дисциплины является формирование знаний о современных методах и приемах объектно-ориентированного подхода к программированию, получение навыков составления программ с использованием объектно-ориентированного подхода на языке программирования C++.
2. Задачи изучения дисциплины: формирование умения применять основные принципы современного подхода к программированию на основе объектно-ориентированной парадигмы -инкапсуляции, наследования, полиморфизма.
3. Изучение базовых понятий; получение знаний о методах и приемах объектно-ориентированного подхода к программированию; способов составления объектно-ориентированных программ на языке программирования C++.
4. Формирование умений применять основные принципы современного подхода к программированию на основе объектно-ориентированной парадигмы: инкапсуляции, наследования, полиморфизма.
5. Освоение навыков применения методов и инструментальных средств разработки объектно-ориентированных программ для решения прикладных задач.

### **3.2 Место дисциплины в структуре ОПОП**

Дисциплина изучается на основе ранее освоенных дисциплин учебного плана:

1. «Информатика»
2. «Программирование»
3. «Информационные технологии»

и обеспечивает изучение последующих дисциплин:

1. «Операционные системы»
2. «Построение и анализ алгоритмов»
3. «Тестирование программного обеспечения»
4. «Учебная практика (технологическая (проектно-технологическая) практика)»
5. «Web-технологии»
6. «Базы данных»
7. «Введение в методы машинного обучения ч.1»
8. «Компьютерная графика»
9. «Введение в методы машинного обучения ч.2»
10. «Логическое программирование»
11. «Основы промышленной разработки программного обеспечения»
12. «Производственная практика (технологическая (проектно-технологическая) практика)»
13. «Распределенные алгоритмы»
14. «Спецификация, проектирование и архитектура программных систем»
15. «Интеллектуальные системы»
16. «Разработка приложений для мобильных платформ»
17. «Управление разработкой и экономика программного проекта»
18. «Цифровая обработка изображений»

### **3.3 Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы**

В результате освоения образовательной программы обучающийся должен достичь следующие результаты обучения по дисциплине:

<b>Код компетенции/ индикатора компетенции</b>	<b>Наименование компетенции/индикатора компетенции</b>
ОПК-7	Способен применять в практической деятельности основные концепции, принципы, теории и факты, связанные с информатикой;
<i>ОПК-7.2</i>	<i>Применяет знания о теориях и фактах, связанных с информатикой, в контексте задач практической деятельности</i>

## **4 СОДЕРЖАНИЕ ДИСЦИПЛИНЫ**

### **4.1 Содержание разделов дисциплины**

#### **4.1.1 Наименование тем и часы на все виды нагрузки**

<b>№ п/п</b>	<b>Наименование темы дисциплины</b>	<b>Лек, ач</b>	<b>Лаб, ач</b>	<b>ИКР, ач</b>	<b>СР, ач</b>
1	Введение	1			
2	Тема 1. Основы объектно-ориентированного подхода	2	2		5
3	Тема 2. Классы и методы	2	4		5
4	Тема 3. Создание и уничтожение объектов класса. Конструкторы и деструкторы	4	4		8
5	Тема 4. Наследование классов	4	4		8
6	Тема 5. Перегрузка операторов	4	4		10
7	Тема 6. Принципы проектирования программ	4	4		10
8	Тема 7. Шаблоны	4	4		10
9	Тема 8. Исключительные ситуации	4	4		10
10	Тема 9. Шаблоны проектирования GoF	4	4		10
11	Заключение	1		1	35
	Итого, ач	34	34	1	111
	Из них ач на контроль	0	0	0	35
	Общая трудоемкость освоения, ач/зе				180/5

#### **4.1.2 Содержание**

<b>№ п/п</b>	<b>Наименование темы дисциплины</b>	<b>Содержание</b>
1	Введение	Предмет дисциплины, ее структура и содержание. Цели и задачи дисциплины. Структура и план учебной деятельности студентов. Рекомендуемая литература. Связь дисциплины с предшествующими и последующими дисциплинами
2	Тема 1. Основы объектно-ориентированного подхода	Виды парадигм программирования. Основные причины и предпосылки создания объектно-ориентированного подхода. Основные принципы объектно-ориентированного подхода. Пользовательские типы данных: enum, struct, class. Различия struct и класса. Поля класса. Доступ к полям класса. Модификаторы доступа к полям класса.

<b>№ п/п</b>	<b>Наименование темы дисциплины</b>	<b>Содержание</b>
3	Тема 2. Классы и методы	Понятие метода класса. Неявный указатель <code>this</code> . Объявление и определение методов. Значения аргументов методов по умолчанию. Перегрузка методов. Константные методы. <code>Mutable</code> поля класса. <code>Static</code> поля классов. <code>Static</code> методы классов. Недостатки <code>static</code> методов в парадигме ООП. UML: отношения классов.
4	Тема 3. Создание и уничтожение объектов класса. Конструкторы и деструкторы	Понятие конструктора класса. Списки инициализации. Конструктор от одного аргумента, особенности. Делегирующий конструктор и конструктор по умолчанию. Ключевое слово <code>explicit</code> и <code>default</code> . Деструктор и уничтожение объекта. Понятие <code>lvalue</code> и <code>rvalue</code> . Конструктор копирования. Конструктор перемещения. Операторы присваивания. Методы, генерируемые автоматически. Ситуации явного и неявного вызова конструкторов и деструкторов. Принцип RAII
5	Тема 4. Наследование классов	Понятие наследования классов. Типы наследования, управление уровня доступа. Порядок конструирования объектов классов. Переопределение методов. Ключевое слово <code>using</code> при наследовании. Виртуальные методы классов. Полиморфный класс и ситуации использования виртуальных методов. Таблица виртуальных функций. Ключевое слово <code>final</code> . Механизм RTTI и определения типа данных. Абстрактный класс и интерфейса. Виртуальное наследование
6	Тема 5. Перегрузка операторов	Понятие оператора и его виды. Специальные операторы в языке C++. Особенности перегрузки унарных и бинарных операторов. Перегрузка инкремента и декремента. Перегрузка логических операторов. Перегрузка оператора приведения к типу. Перегрузка операторов ввода и вывода в поток. Пользовательские литералы. Функтор и его применение. Лямбда-выражения. Связь функторов, лямбда-выражений и обычных функций
7	Тема 6. Принципы проектирования программ	Что такое принципе проектирования. Понятие SOLID. Принцип единственной ответственности. Принцип открытости/закрытости. Принцип подстановки Барбары Лисков. Принцип разделения интерфейсов. Принцип инверсии зависимости
8	Тема 7. Шаблоны	Понятие шаблона и связь с полиморфизмом. Шаблонные функции и классы. Инстанцирование и двухэтапная трансляция шаблонов. Выведение типов в шаблонах. Шаблоны для возвращаемого из функции типа. Перегрузка шаблонных функций. Специализация шаблонов. Применение шаблонов, СРТР и нешаблонная база. Шаблоны с переменным количеством параметров. Распаковка параметров

<b>№ п/п</b>	<b>Наименование темы дисциплины</b>	<b>Содержание</b>
9	Тема 8. Исключительные ситуации	Понятие исключительной ситуации. Подходы к обработке исключительных ситуаций. Исключительные ситуации в конструкторах и деструкторах. Механизм обработки исключений. Раскрутка стека при исключениях. Перехват исключений. Спецификатор поехсерт. Гарантии механизма исключений. Умные указатели
10	Тема 9. Шаблоны проектирования GoF	Шаблоны проектирования. Порождающие шаблоны. Структурные шаблоны. Поведенческие шаблоны
11	Заключение	Основные тенденции и направления развития объектно-ориентированных технологий создания программ в условиях распределения процесса разработки в пространстве и во времени и необходимости обеспечения качества программных продуктов

## **4.2 Перечень лабораторных работ**

<b>Наименование лабораторной работы</b>	<b>Количество ауд. часов</b>
1. Создание классов, конструкторов и методов	4
2. Интерфейс, динамический полиморфизм	6
3. Логирование, перегрузка операций	6
4. Уровни абстракции, управление программой	6
5. Шаблонные классы	6
6. СерIALIZАЦИЯ, исключения	6
<b>Итого</b>	<b>34</b>

## **4.3 Перечень практических занятий**

Практические занятия не предусмотрены.

## **4.4 Курсовое проектирование**

Курсовая работа (проект) не предусмотрены.

## **4.5 Реферат**

Реферат не предусмотрен.

## **4.6 Индивидуальное домашнее задание**

Индивидуальное домашнее задание не предусмотрено.

#### **4.7 Доклад**

Доклад не предусмотрен.

#### **4.8 Кейс**

Кейс не предусмотрен.

#### **4.9 Организация и учебно-методическое обеспечение самостоятельной работы**

Изучение дисциплины сопровождается самостоятельной работой студентов с рекомендованными преподавателем литературными источниками и информационными ресурсами сети Интернет.

Планирование времени для изучения дисциплины осуществляется на весь период обучения, предусматривая при этом регулярное повторение пройденного материала. Обучающимся, в рамках внеаудиторной самостоятельной работы, необходимо регулярно дополнять сведениями из литературных источников материал, законспектированный на лекциях. При этом на основе изучения рекомендованной литературы целесообразно составить конспект основных положений, терминов и определений, необходимых для освоения разделов учебной дисциплины.

Особое место уделяется консультированию, как одной из форм обучения и контроля самостоятельной работы. Консультирование предполагает особым образом организованное взаимодействие между преподавателем и студентами, при этом предполагается, что консультант либо знает готовое решение, которое он может предписать консультируемому, либо он владеет способами деятельности, которые указывают путь решения проблемы.

Текущая СРС	Примерная трудоемкость, ач
Работа с лекционным материалом, с учебной литературой	32

<b>Текущая СРС</b>	<b>Примерная трудоемкость, ач</b>
Опережающая самостоятельная работа (изучение нового материала до его изложения на занятиях)	0
Самостоятельное изучение разделов дисциплины	0
Выполнение домашних заданий, домашних контрольных работ	0
Подготовка к лабораторным работам, к практическим и семинарским занятиям	24
Подготовка к контрольным работам, коллоквиумам	20
Выполнение расчетно-графических работ	0
Выполнение курсового проекта или курсовой работы	0
Поиск, изучение и презентация информации по заданной проблеме, анализ научных публикаций по заданной теме	0
Работа над междисциплинарным проектом	0
Анализ данных по заданной теме, выполнение расчетов, составление схем и моделей, на основе собранных данных	0
Подготовка к зачету, дифференцированному зачету, экзамену	35
<b>ИТОГО СРС</b>	<b>111</b>

## **5 Учебно-методическое обеспечение дисциплины**

### **5.1 Перечень основной и дополнительной литературы, необходимой для освоения дисциплины**

<b>№ п/п</b>	<b>Название, библиографическое описание</b>	<b>К-во экз. в библ.</b>
<b>Основная литература</b>		
1	Мейер, Берtrand. Объектно-ориентированное конструирование программных систем [Текст] / Б. Мейер, 2005. -XXVIII, 1198 с.	50
2	Клюшин, Дмитрий Анатольевич. Полный курс C++ [Текст] / Д.А. Клюшин, 2004. -667 с	50
3	Зыков, Сергей Викторович. Программирование. Объектно-ориентированный подход [Электронный ресурс] : Учебник и практикум для вузов / Зыков С. В., 2020. -155 с	неогр
4	Объектно-ориентированное программирование [Текст] : метод. указания к лаб. работам / Санкт-Петербургский государственный электротехнический университет им. В.И. Ульянова (Ленина) "ЛЭТИ", 2007. -32 с.	128
<b>Дополнительная литература</b>		
1	Пол, Айра. Объектно-ориентированное программирование на C++ [Текст] : научное издание / А. Пол; Пер. с англ. Д. Ковальчука, 2001. -461 с	17
2	Лафоре, Роберт. Объектно-ориентированное программирование в C++ [Текст] : перевод. изд. / Р.Лафоре ; [Пер. с англ. А.Кузнецов и др.], 2008. - 923 с.	4
3	Кубенский А. Структуры и алгоритмы обработки данных: объектно ориентированный подход и реализация на C++ [Электронный ресурс] / А. Кубенский, 2010. -464 с.	неогр

### **5.2 Перечень ресурсов информационно-телекоммуникационной сети «Интернет», используемых при освоении дисциплины**

<b>№ п/п</b>	<b>Электронный адрес</b>
1	Бьерн Страуструп. Язык программирования C++ <a href="http://lib.ru/CPPHB/cpptut.txt">http://lib.ru/CPPHB/cpptut.txt</a>
2	Основы программирования, Автор: В.В. Борисенко (ИНТУИТ.ru:Интернет-Университет Информационных Технологий) <a href="http://www.intuit.ru/department/se/pbmsu/">http://www.intuit.ru/department/se/pbmsu/</a>

№ п/п	Электронный адрес
3	Бьерн Страуструп. Программирование: Принципы и практика использования C++ (на англ.) = Programming --Principles and Practice Using C++ <a href="https://e-library.namdu.uz/30%20%D0%A2%D0%B5%D1%85%D0%BD%D0%B8%D0%BA%D0%B0%20%D1%84%D0%B0%D0%BD%D0%BB%D0%B0%D1%80%D0%A1%D1%82%D1%80%D0%B0%D1%83%D1%81%D1%82%D1%80%D1%83%D0%BF%20%D0%91.%20-%20%D0%9F%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5.%20%D0%9F%D1%80%D0%B8%D0%BD%D1%86%D0%B8%D0%BF%D1%8B%20%D0%B8%20%D0%BF%D1%80%D0%BA%D1%82%D0%B8%D0%BA%D0%B0%20%D0%B8%D1%81%D0%BF%D0%BE%D0%BB%D1%8C%D0%B7%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F%20C++%20-%2020211.pdf">https://e-library.namdu.uz/30%20%D0%A2%D0%B5%D1%85%D0%BD%D0%B8%D0%BA%D0%B0%20%D1%84%D0%B0%D0%BD%D0%BB%D0%B0%D1%80%D0%A1%D1%82%D1%80%D0%B0%D1%83%D1%81%D1%82%D1%80%D1%83%D0%BF%20%D0%91.%20-%20%D0%9F%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5.%20%D0%9F%D1%80%D0%B8%D0%BD%D1%86%D0%B8%D0%BF%D1%8B%20%D0%B8%20%D0%BF%D1%80%D0%BA%D1%82%D0%B8%D0%BA%D0%B0%20%D0%B8%D1%81%D0%BF%D0%BE%D0%BB%D1%8C%D0%B7%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F%20C++%20-%2020211.pdf</a>

### 5.3 Адрес сайта курса

Адрес сайта курса: <https://vec.etu.ru/moodle/course/view.php?id=10632>

## **6 Критерии оценивания и оценочные материалы**

### **6.1 Критерии оценивания**

Для дисциплины «Объектно-ориентированное программирование» формой промежуточной аттестации является экзамен. Оценивание качества освоения дисциплины производится с использованием рейтинговой системы.

#### **Экзамен**

<b>Оценка</b>	<b>Количество баллов</b>	<b>Описание</b>
Неудовлетворительно	0 – 36	теоретическое содержание курса не освоено, необходимые практически навыки и умения не сформированы, выполненные учебные задания содержат грубые ошибки, дополнительная самостоятельная работа над курсом не приведет к существенному повышению качества выполнения учебных заданий
Удовлетворительно	37 – 48	теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые практические навыки и умения работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий содержат ошибки
Хорошо	49 – 72	теоретическое содержание курса освоено полностью, без пробелов, некоторые практические навыки и умения сформированы недостаточно, все предусмотренные программой обучения учебные задания выполнены, качество выполнения ни одного из них не оценено минимальным числом баллов, некоторые виды заданий выполнены с ошибками
Отлично	73 – 100	теоретическое содержание курса освоено полностью, без пробелов, необходимые практические навыки и умения сформированы, все предусмотренные программой обучения учебные задания выполнены, качество их выполнения оценено количеством баллов, близким к максимальному

## **Особенности допуска**

Допуск к экзамену: выполнение и защита 6 лабораторных работ, итоговый тест. Оценка за промежуточную аттестацию по выбору студента может быть выставлена на основании оценки за итоговый тест (более 36 баллов), либо студент отвечает на вопросы билета на экзамене.

## **6.2 Оценочные материалы для проведения текущего контроля и промежуточной аттестации обучающихся по дисциплине**

### **Вопросы к экзамену**

<b>№ п/п</b>	<b>Описание</b>
1	Виды парадигм программирования
2	Поля класса. Доступ к полям класса.
3	Неявный указатель this. Объявление и определение методов
4	Mutable поля класса
5	Понятие конструктора класса
6	Ключевое слово explicit и default
7	Переопределение методов
8	Механизм RTTI и определения типа данных
9	Специальные операторы в языке C++
10	Перегрузка оператора приведения к типу
11	Понятие SOLID
12	Принцип разделения интерфейсов
13	Понятие шаблона и связь с полиморфизмом
14	Перегрузка шаблонных функций
15	Подходы к обработке исключительных ситуаций
16	Механизм обработки исключений
17	Порождающие шаблоны
18	Поведенческие шаблоны
19	Делегирующий конструктор и конструктор по умолчанию
20	Ключевое слово explicit и default
21	Деструктор и уничтожение объекта

### **Форма билета**

Министерство науки и высшего образования Российской Федерации

ФГАОУ ВО «Санкт-Петербургский государственный электротехнический

## ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 1

Дисциплина **Объектно-ориентированное программирование** ФКТИ

1. Неявный указатель this. Объявление и определение методов.
2. Перегрузка шаблонных функций.
3. Задача.

УТВЕРЖДАЮ

Заведующий кафедрой

**Образцы задач (заданий) для контрольных (проверочных) работ**

### **Пример задачи к билету на экзамене**

#### *Задача к билету №1*

Написать программу на языке C++ с использованием объектно-ориентированного подхода, решающую следующую задачу.

Дана строка, содержащая несколько слов, разделенных пробелами. Необходимо из введенной строки удалить слова, состоящие из n букв. Число n задается пользователем.

### **Примеры вопросов для формирования итогового теста**

1. Основными предпосылками создания объектно-ориентированного подхода к проектированию и созданию программного обеспечению являются:
  1. рост сложности программного обеспечения;
  2. изменение архитектур вычислительных средств (ВС) в целях повышения производительности, надежности и коммуникативности;

3. упрощение взаимодействия пользователей с ВС и интеллектуализация ВС;

**4. все перечисленные выше факторы**

2. Инкапсуляция - это:

1. размещение объявления класса в отдельном файле;

**2. объединение и локализация в рамках объекта, как единого целого, данных и функций, обрабатывающих эти данные;**

3. свойство классов порождать своих потомков и наследовать свойства (данные и методы) своих родителей;

4. свойство объектов-родственников по-разному осуществлять однотипные (и даже одинаково поименованные) действия;

5. возможность создания и уничтожения объектов

3. Какие из следующих типов данных C++ относятся к структурным

1. void

2. enum

3. union

4. unsigned char

5. double &

6. все перечисленные

4. Какая из приведенных конструкций объявляет массив указателей на объект типа t:

1. t \* vpt[];

2. t \* (vpt[]);

3. t (\* vpt)[];

5. Обязательно ли в функции должен быть оператор return?

1. Да

2. Нет, если не указан тип возвращаемого значения

**3. Нет, если тип возвращаемого значения – void**

Весь комплект контрольно-измерительных материалов для проверки сформированности компетенции (индикатора компетенции) размещен в закрытой части по адресу, указанному в п. 5.3

### **6.3 График текущего контроля успеваемости**

<b>Неделя</b>	<b>Темы занятий</b>	<b>Вид контроля</b>
1	Тема 1. Основы объектно-ориентированного подхода Тема 2. Классы и методы	
2		
3		Отчет по лаб. работе
4	Тема 3. Создание и уничтожение объектов класса. Конструкторы и деструкторы Тема 4. Наследование классов	
5		
6		Отчет по лаб. работе
7	Тема 5. Перегрузка операторов Тема 6. Принципы проектирования программ	
8		
9		Отчет по лаб. работе
10	Тема 7. Шаблоны	
11		
12		Отчет по лаб. работе
13	Тема 8. Исключительные ситуации	
14		
15		Отчет по лаб. работе
16	Тема 9. Шаблоны проектирования GoF	Отчет по лаб. работе
17	Заключение	Тест

### **6.4 Методика текущего контроля**

#### **на лекционных занятиях**

Текущий контроль включает в себя:

- контроль посещаемости (не менее 80% занятий).

#### **на лабораторных занятиях**

В процессе обучения по дисциплине студент обязан выполнить и успешно защитить 6 лабораторных работ. Под выполнением лабораторных работ подразумевается подготовка к работе, проведение экспериментальных исследований, подготовка отчета и его защита. Отчет оформляется после выполнения экспериментальных исследований и представляется преподавателю на проверку. После проверки отчет либо возвращается (при наличии замечаний) на доработку, либо допускается к защите.

Лабораторные работы защищаются студентами индивидуально, в часы,

отведенные для лабораторных работ. На защите лабораторной работы студент должен показать: понимание постановки задачи, подхода к ее решению, умение объяснять ход решения, выбор тех или иных методик решения задачи. Преподаватель задает вопросы, позволяющие определить глубину понимания теоретического материала, который лежит в основе решения задачи лабораторной работы, а также самостоятельность ее выполнения.

Текущий контроль включает в себя контроль выполнения лабораторной работы и сдачи в срок отчета по лабораторной работе.

Критерии оценивания: «не зачтено» - ставится, если основное содержание материала работы не раскрыто, не даны ответы на вопросы преподавателя, допущены грубые ошибки в определении понятий и в использовании терминологии; «зачтено» ставится, если продемонстрировано усвоение основного содержания материала, работа выполнена полностью, самостоятельно и оформлена в соответствии с требованиями.

**Итоговый тест** состоит из 25 вопросов с возможностью выбора вариантов ответа и оценивается следующим образом: за каждый полностью правильный ответ начисляется 4 балла, если выбраны не все правильные варианты ответа, либо выбраны неправильные, баллы не начисляются.

Результирующая оценка за дисциплину формируется по результатам итогового теста или при сдаче экзамена по билетам с 3 вопросами в них.

### **самостоятельной работы студентов**

Контроль самостоятельной работы студентов осуществляется на осуществляется на лекционных, лабораторных занятиях по методикам, описанным выше.

## **7 Описание информационных технологий и материально-технической базы**

<b>Тип занятий</b>	<b>Тип помещения</b>	<b>Требования к помещению</b>	<b>Требования к программному обеспечению</b>
Лекция	Лекционная аудитория	Количество посадочных мест – в соответствии с контингентом, рабочее место преподавателя, IBM-совместимый компьютер Pentium или выше, проектор, экран, меловая или маркерная доска	1) Windows 7 и выше; 2) Microsoft Office 2007 и выше
Лабораторные работы	Лаборатория	Количество посадочных мест – в соответствии с контингентом. Оснащено компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду университета. Рабочее место преподавателя.	1) Windows XP и выше; 2) Microsoft Office 2007 и выше
Самостоятельная работа	Помещение для самостоятельной работы	Оснащено компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду университета.	1) Windows XP и выше; 2) Microsoft Office 2007 и выше

## **8 Адаптация рабочей программы для лиц с ОВЗ**

Адаптированная программа разрабатывается при наличии заявления со стороны обучающегося (родителей, законных представителей) и медицинских показаний (рекомендациями психолого-медико-педагогической комиссии). Для инвалидов адаптированная образовательная программа разрабатывается в соответствии с индивидуальной программой реабилитации.

## **ЛИСТ РЕГИСТРАЦИИ ИЗМЕНЕНИЙ**

<b>№ п/п</b>	<b>Дата</b>	<b>Изменение</b>	<b>Дата и номер протокола заседания УМК</b>	<b>Автор</b>	<b>Начальник ОМОЛА</b>