



Федеральное государственное бюджетное образовательное учреждение  
высшего образования

САНКТ-ПЕТЕРБУРГСКИЙ ГОСУДАРСТВЕННЫЙ  
АРХИТЕКТУРНО-СТРОИТЕЛЬНЫЙ УНИВЕРСИТЕТ

Кафедра Информационных систем и технологий

УТВЕРЖДАЮ  
Начальник учебно-методического управления

«29» июня 2023 г.

## **РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)**

Объектно-ориентированное программирование

направление подготовки/специальность 01.03.02 Прикладная математика и информатика

направленность (профиль)/специализация образовательной программы Прикладная математика и информатика

Форма обучения очная

Санкт-Петербург, 2023

## 1. Цели и задачи освоения дисциплины (модуля)

Целями освоения дисциплины являются получение знаний, умений и навыков в области разработки объектно-ориентированных программ на языке С++ для решения прикладных задач в различных сферах деятельности на базе:

- системного изложения основ алгоритмического языка С++, его возможностей по разработке объектно-ориентированных программ и Windows-приложений;
- ознакомления студентов с технологиями и инструментальными средствами разработки программного обеспечения, основными структурами данных и методами работы с ними, в том числе с использованием стандартной библиотеки классов языка С++.

Задачами освоения дисциплины являются обучение студентов навыкам постановки задачи, разработки алгоритмов, выбора структуры данных, составления как процедурно-ориентированных, так и объектно-ориентированных программ на языке С++ для решения широкого круга практических задач в инженерных и экономических расчетах, обработки текстовой, графической и другой информации.

## 2. Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенных с индикаторами достижения компетенций

Код и наименование компетенции	Код и наименование индикатора достижения компетенции	Планируемые результаты обучения по дисциплине, обеспечивающие достижение планируемых результатов освоения ОПОП
--------------------------------	--	--

<p>ОПК-5                      Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения</p>	<p>ОПК-5.3                      Демонстрирует применение разработанного алгоритма и (или) компьютерной программы</p>	<p><b>знает</b></p> <ul style="list-style-type: none"> <li>- основные понятия и современные принципы разработки алгоритмов и компьютерных программ, пригодных для практического применения;</li> <li>- основные методы математического моделирования, классификации и условий применения моделей, методов и средств проектирования информационных и автоматизированных систем, инструментальные средства моделирования и проектирования;</li> <li>- методику составления алгоритмов для решения прикладных задач и оценки их эффективности;</li> <li>- возможности вычислительной техники, имеющегося программного обеспечения, особенности реализации методов вычислительной математики;</li> </ul> <p><b>умеет</b></p> <ul style="list-style-type: none"> <li>- разрабатывать алгоритмы и компьютерные программы для практического применения</li> <li>- осуществлять выбор программного обеспечения в зависимости от целей и видов решаемых задач;</li> <li>- выбирать подходящий метод, алгоритм для решения поставленной задачи</li> </ul> <p><b>владеет</b></p> <ul style="list-style-type: none"> <li>- навыками разработки алгоритмов для решения прикладных задач и их программных реализаций;</li> <li>- навыками демонстрации применения разработанного алгоритма и (или) компьютерной программы</li> </ul>
<p>ПК-1                      Способен проектировать и разрабатывать программное обеспечение</p>	<p>ПК-1.1                      Осуществляет выбор метода проектирования программного обеспечения, структуры данных, базы данных или программных интерфейсов из типовых решений и шаблонов</p>	<p><b>знает</b></p> <ul style="list-style-type: none"> <li>- методы проектирования программного обеспечения;</li> <li>- основные средства проектирования программного обеспечения;</li> <li>- основные структуры данных;</li> </ul> <p><b>умеет</b></p> <ul style="list-style-type: none"> <li>- проектировать программное обеспечение на базе типовых решений и шаблонов;</li> </ul> <p><b>владеет</b></p> <ul style="list-style-type: none"> <li>- навыками проектирования программного обеспечения на базе типовых решений и шаблонов;</li> </ul>

ПК-1 Способен проектировать и разрабатывать программное обеспечение	ПК-1.2 Осуществляет выбор типового решения и/или шаблона проектирования программного обеспечения, структуры данных, базы данных или программных интерфейсов в соответствии с поставленной задачей	<b>знает</b> - основные шаблоны проектирования; - основные структуры данных; <b>умеет</b> - осуществлять выбор типового решения и/или шаблона проектирования программного обеспечения в соответствии с поставленной задачей <b>владеет</b> - навыками осуществления выбор типового решения и/или шаблона проектирования программного обеспечения, структуры данных в соответствии с поставленной задачей
ПК-1 Способен проектировать и разрабатывать программное обеспечение	ПК-1.3 Разрабатывает программное обеспечение, структуры данных, базы данных или программные интерфейсы	<b>знает</b> - основные парадигмы объектно-ориентированного программирования; - основные функции и классы языка С++; - основные структуры данных; <b>умеет</b> - разрабатывать программное обеспечение в объектно-ориентированном стиле; - использовать при разработке программного обеспечения библиотечные функции, классы и структуры данных языка С++; - создавать свои собственные классы; - создавать свои собственные структуры данных; <b>владеет</b> - навыками разработки объектно-ориентированного программного обеспечения с использованием стандартных средств языка С++; - навыками разработки объектно-ориентированного программного обеспечения с использованием библиотек языка С++;

### 3. Указание места дисциплины (модуля) в структуре образовательной программы

Данная дисциплина (модуль) включена в Блок «Дисциплины, модули» Б1.О.31.02 основной профессиональной образовательной программы 01.03.02 Прикладная математика и информатика и относится к обязательной части учебного плана.

№ п/п	Предшествующие дисциплины	Код и наименование индикатора достижения компетенции
1	Программирование для ЭВМ	ОПК-5.1, ОПК-5.2
2	Информационные технологии	УК-1.1, УК-1.2, УК-1.6

знать:

- основные понятия информатики;
- основы разработки алгоритмов;
- основные типы данных;
- основные структуры данных;
- основные механизмы обработки информации в памяти компьютера.

уметь:

- работать на персональном компьютере;
- пользоваться операционной системой;
- использовать возможности вычислительной техники и программного обеспечения.

владеть:

- первичными навыками и основными методами решения математических задач;
- навыками работы с учебной литературой;
- основными приёмами работы на компьютере с прикладным программным обеспечением.

№ п/п	Последующие дисциплины	Код и наименование индикатора достижения компетенции
1	Программирование для Интернет	ПК-2.1, ПК-3.1, ПК-3.2, ПК-3.3
2	Программирование на языке Python	ПК-1.3, ПК-1.4

**4. Объем дисциплины (модуля) в зачетных единицах с указанием количества академических часов, выделенных на контактную работу обучающихся с преподавателем (по видам учебных занятий) и на самостоятельную работу обучающихся**

Вид учебной работы	Всего часов	Из них часы на практическую подготовку	Семестр
			5
<b>Контактная работа</b>	64		64
Практические занятия (Пр)	64	64	64
<b>Иная контактная работа, в том числе:</b>	1,5		1,5
консультации по курсовой работе (проекту), контрольным работам (РГР)	1		1
контактная работа на аттестацию (сдача зачета, зачета с оценкой; защита курсовой работы (проекта); сдача контрольных работ (РГР))	0,25		0,25
контактная работа на аттестацию в сессию (консультация перед экзаменом и сдача экзамена)	0,25		0,25
<b>Часы на контроль</b>	26,75		26,75
<b>Самостоятельная работа (СР)</b>	51,75		51,75
<b>Общая трудоемкость дисциплины (модуля)</b>			
<b>часы:</b>	144		144
<b>зачетные единицы:</b>	4		4

**5. Содержание дисциплины (модуля), структурированное по разделам (темам) с указанием отведенного на них количества академических часов и видов учебных занятий**

5.1. Тематический план дисциплины (модуля)

№	Разделы дисциплины	Семестр	Контактная работа (по учебным занятиям), час.						СР	Всего, час.	Код индикатора достижения компетенции
			лекции		ПЗ		ЛР				
			всего	из них на практическую подготовку	всего	из них на практическую подготовку	всего	из них на практическую подготовку			
1.	1 раздел. Основы программирования на языке Java										
1.1.	Введение в Java. Знакомство с JVM, JRE, JDK. Сравнительный анализ Java и других языков программирования.	5			4	4			3	7	ОПК-5.3, ПК-1.1, ПК-1.2, ПК-1.3
1.2.	Структура программы на языке Java. Импорт пакетов. Консольный ввод/вывод данных.	5			4	4			3	7	ОПК-5.3, ПК-1.1, ПК-1.2, ПК-1.3
1.3.	Организация вычислений. Циклы. Условные операторы. Использование ссылочных и примитивных типов данных.	5			4	4			3	7	ОПК-5.3, ПК-1.1, ПК-1.2, ПК-1.3
1.4.	Типы данных в Java.	5			4	4			3	7	ОПК-5.3, ПК-1.1, ПК-1.2, ПК-1.3
1.5.	Объектно-ориентированные возможности языка Java. Структура класса. Уровни доступа к полям класса. Принципы работы с объектами. Паттерн проектирования Singleton.	5			4	4			3	7	ОПК-5.3, ПК-1.1, ПК-1.2, ПК-1.3
1.6.	Одномерные и многомерные массивы. Метод clone. Класс Arrays.	5			4	4			3	7	ОПК-5.3, ПК-1.1, ПК-1.2, ПК-1.3
1.7.	Наследование. Использование конструктора родительского класса внутри дочернего класса.	5			4	4			3	7	ОПК-5.3, ПК-1.1, ПК-1.2, ПК-1.3
1.8.	Полиморфизм. Абстракции и интерфейсы. Абстрактные классы. Паттерн проектирования Абстрактная фабрика. Паттерн проектирования Фабричный метод.	5			4	4			3	7	ОПК-5.3, ПК-1.1, ПК-1.2, ПК-1.3



2.1.	Иная контактная работа	5								1,25	ОПК-5.3, ПК-1.1, ПК-1.2, ПК-1.3
3.	3 раздел. Контроль										
3.1.	Экзамен	5								27	ОПК-5.3, ПК-1.1, ПК-1.2, ПК-1.3

#### 5.1. Практические занятия

№ разд	Наименование раздела и темы практических занятий	Наименование и содержание практических занятий									
24	Введение в Java. Знакомство с JVM, JRE, JDK. Сравнительный анализ Java и других языков программирования.	Введение в Java. Знакомство с JVM, JRE, JDK. Сравнительный анализ Java и других языков программирования. Знакомство с JVM, JRE, JDK. Создание простейших программ на языке Java.									
25	Структура программы на языке Java. Импорт пакетов. Консольный ввод/вывод данных.	Структура программы на языке Java. Импорт пакетов. Консольный ввод/вывод данных.  Составление программ с использованием пакетов Java. Составление программ, использующих консольный ввод/вывод данных.									
26	Организация вычислений. Циклы. Условные операторы. Использование ссылочных и примитивных типов данных.	Организация вычислений. Циклы. Условные операторы. Использование ссылочных и примитивных типов данных. Знакомство с условными операторами в языке Java. Изучение грамотных способов сравнения ссылочных и примитивных типов данных. Понимание правильного выбора цикла для конкретной задачи на языке Java. Составление программ.									
27	Типы данных в Java.	Типы данных в Java. Изучение целочисленных, вещественных, символьных и логических примитивных типов данных. Повторение разницы между примитивными и ссылочными типами данных. Понимание принципов работы с объектами. Знакомство со String и Enum. Составление программ.									
28	Объектно-ориентированные возможности языка Java. Структура класса. Уровни доступа к полям класса. Принципы работы с объектами. Паттерн проектирования Singleton.	Объектно-ориентированные возможности языка Java. Структура класса. Уровни доступа к полям класса. Принципы работы с объектами. Паттерн проектирования Singleton. Знакомство со структурой класса на языке Java. Понимание разницы между публичными, приватными и защищенными секциями. Изучение правильного выбора полей класса и методов, достаточных для реализации решения задачи. Изучение паттерна проектирования Singleton. Составление программ.									
29	Одномерные и многомерные массивы. Метод clone. Класс Arrays.	Одномерные и многомерные массивы. Метод clone. Класс Arrays. Многомерные и одномерные массивы. Способы взаимодействия с одномерными и многомерными массивами. Изучение принципа работы метода clone. Знакомство с классом Arrays и способами его использования при работе с массивами.									

30	<p>Наследование. Использование конструктора родительского класса внутри дочернего класса.</p>	<p>Наследование. Использование конструктора родительского класса внутри дочернего класса. Наследование. Принципы грамотного наследования и способов его организации. Изучение методов использования конструктора родительского класса внутри дочернего при наследовании. Составление объектно-ориентированных программ с использованием механизма наследования.</p>
31	<p>Полиморфизм. Абстракции и интерфейсы. Абстрактные классы. Паттерн проектирования Абстрактная фабрика. Паттерн проектирования Фабричный метод.</p>	<p>Полиморфизм. Абстракции и интерфейсы. Абстрактные классы. Паттерн проектирования Абстрактная фабрика. Паттерн проектирования Фабричный метод. Составление объектно-ориентированных программ с использованием полиморфизма, абстрактных классов, паттернов проектирования Абстрактная фабрика и Фабричный метод.</p>
32	<p>Исключения. Использование исключений при разработке собственных библиотек. Stacktrace.</p>	<p>Исключения. Использование исключений при разработке собственных библиотек. Stacktrace. Исключения, Stacktrace. Изучение древа наследования исключений в Java. Понимание уместности и способов “выбрасывания” исключений при неправильной работе программы. Знакомство с принципами использования исключений при разработке собственных библиотек. Понимание устройства Stacktrace. Составление объектно-ориентированных программ с использованием исключений.</p>
33	<p>Generic классы. Способы организации. Методы Generic классов.</p>	<p>Generic классы. Способы организации. Методы Generic классов. Знакомство с Generic классами и способами их организации. Изучение структуры методов у Generic классов. Составление программ с использованием Generic классов.</p>
34	<p>Коллекции List. Основные методы. Организация хранения данных с помощью динамического массива и связанного списка. Классы ArrayList и LinkedList.</p>	<p>Коллекции List. Основные методы. Организация хранения данных с помощью динамического массива и связанного списка. Классы ArrayList и LinkedList. Коллекции List. Знакомство с методами коллекции List. Изучение организации хранения данных с помощью динамического массива и связанного списка. Понимание разницы между классами ArrayList и LinkedList. Составление объектно-ориентированных программ с использованием коллекции List.</p>
35	<p>Коллекции Queue и Stack. Основные методы. Организация хранения данных с помощью динамического массива и связанного списка на базе очереди и стека. Отличие интерфейсов Queue и Stack.</p>	<p>Коллекции Queue и Stack. Основные методы. Организация хранения данных с помощью динамического массива и связанного списка на базе очереди и стека. Отличие интерфейсов Queue и Stack. Коллекции Queue и Stack. Знакомство с методами коллекций Queue и Stack. Изучение организации хранения данных с помощью динамического массива и связанного списка на базе очереди и стека. Понимание разницы между интерфейсами Queue и Stack. Составление объектно-ориентированных программ с использованием коллекций Queue и Stack.</p>
36	<p>Коллекция HashSet. Коллекция Set. Основные принципы работы с Hash-кодом объекта. Взаимодействие объектов внутри</p>	<p>Коллекция HashSet. Коллекция Set. Основные принципы работы с Hash-кодом объекта. Взаимодействие объектов внутри коллекции. Переопределение метода hashCode. Коллекция HashSet. Знакомство с коллекцией Set. Изучение принципов работы с Hash-кодом объекта. Понимание внутреннего взаимодействия между объектами внутри коллекции. Знакомство с</p>

	коллекции. Переопределение метода hashCode.	переопределением метода hashCode. Составление объектно-ориентированных программ с использованием коллекции HashSet.
37	Коллекция HashMap. Коллекция Map и ее реализация на основе класса HashMap. Внутреннее устройство коллекции.	Коллекция HashMap. Коллекция Map и ее реализация на основе класса HashMap. Внутреннее устройство коллекции. Знакомство с коллекцией Map и ее реализацией на основе класса HashMap. Понимание внутреннего устройства коллекции в формате ключ-значение. Повторение важности переопределения метода hashCode при использовании собственных объектов. Составление программ с использованием коллекции HashMap.
38	Коллекции TreeSet. Методы взаимодействия с коллекцией Set. Структуры для хранения данных: бинарное дерево, красно-черное дерево. Внутреннее взаимодействие между объектами внутри коллекции.	Коллекции TreeSet. Методы взаимодействия с коллекцией Set. Структуры для хранения данных: бинарное дерево, красно-черное дерево. Внутреннее взаимодействие между объектами внутри коллекции. Коллекции TreeSet. Повторение методов взаимодействия с коллекцией Set. Изучение таких структур как бинарное и красно-черное дерево. Понимание внутреннего взаимодействия между объектами внутри коллекции. Составление объектно-ориентированных программ с использованием коллекции TreeSet.
39	Коллекция TreeMap. Методы взаимодействия с коллекцией Map. Реализация коллекции на основе класса TreeMap. Внутреннее устройство коллекции в формате ключ-значение.	Коллекция TreeMap. Методы взаимодействия с коллекцией Map. Реализация коллекции на основе класса TreeMap. Внутреннее устройство коллекции в формате ключ-значение. Коллекция TreeMap. Повторение методов взаимодействия с коллекцией Map. Изучение реализации коллекции на основе класса TreeMap. Повторение внутреннего устройства коллекции в формате ключ-значение. Составление объектно-ориентированных программ с использованием коллекции TreeMap.

## 5.2. Самостоятельная работа обучающихся

№ разд	Наименование раздела дисциплины и темы	Содержание самостоятельной работы
24	Введение в Java. Знакомство с JVM, JRE, JDK. Сравнительный анализ Java и других языков программирования.	Введение в Java. Знакомство с JVM, JRE, JDK. Сравнительный анализ Java и других языков программирования. Изучение материала. Подготовка к практическим занятиям и тестированию.
25	Структура программы на языке Java. Импорт пакетов. Консольный ввод/вывод данных.	Структура программы на языке Java. Импорт пакетов. Консольный ввод/вывод данных. Изучение материала. Подготовка к практическим занятиям и тестированию.
26	Организация вычислений. Циклы. Условные операторы. Использование ссылочных и примитивных типов	Организация вычислений. Циклы. Условные операторы. Использование ссылочных и примитивных типов данных. Изучение материала. Подготовка к практическим занятиям и тестированию.

	данных.	
27	Типы данных в Java.	Типы данных в Java. Изучение материала. Подготовка к практическим занятиям и тестированию.
28	Объектно-ориентированные возможности языка Java. Структура класса. Уровни доступа к полям класса. Принципы работы с объектами. Паттерн проектирования Singleton.	Объектно-ориентированные возможности языка Java. Структура класса. Уровни доступа к полям класса. Принципы работы с объектами. Паттерн проектирования Singleton. Изучение материала. Подготовка к практическим занятиям и тестированию.
29	Одномерные и многомерные массивы. Метод clone. Класс Arrays.	Одномерные и многомерные массивы. Метод clone. Класс Arrays. Изучение материала. Подготовка к практическим занятиям и тестированию.
30	Наследование. Использование конструктора родительского класса внутри дочернего класса.	Наследование. Использование конструктора родительского класса внутри дочернего класса. Изучение материала. Подготовка к практическим занятиям и тестированию.
31	Полиморфизм. Абстракции и интерфейсы. Абстрактные классы. Паттерн проектирования Абстрактная фабрика. Паттерн проектирования Фабричный метод.	Полиморфизм. Абстракции и интерфейсы. Абстрактные классы. Паттерн проектирования Абстрактная фабрика. Паттерн проектирования Фабричный метод. Изучение материала. Подготовка к практическим занятиям и тестированию.
32	Исключения. Использование исключений при разработке собственных библиотек. Stacktrace.	Исключения. Использование исключений при разработке собственных библиотек. Stacktrace. Изучение материала. Подготовка к практическим занятиям и тестированию.
33	Generic классы. Способы организации. Методы Generic классов.	Generic классы. Способы организации. Методы Generic классов. Изучение материала. Подготовка к практическим занятиям и тестированию.
34	Коллекции List. Основные методы. Организация хранения данных с помощью динамического массива и связанного списка. Классы ArrayList и LinkedList.	Коллекции List. Основные методы. Организация хранения данных с помощью динамического массива и связанного списка. Классы ArrayList и LinkedList. Изучение материала. Подготовка к практическим занятиям и тестированию.
35	Коллекции Queue и Stack. Основные	Коллекции Queue и Stack. Основные методы. Организация хранения данных с помощью динамического массива и

	<p>методы.          Организация хранения данных с помощью динамического массива и связного списка на базе очереди и стека.          Отличие интерфейсов Queue и Stack.</p>	<p>связного списка на базе очереди и стека. Отличие интерфейсов Queue и Stack.          Изучение материала. Подготовка к практическим занятиям и тестированию.</p>
36	<p>Коллекция HashSet.          Коллекция Set.          Основные принципы работы с Hash-кодом объекта.          Взаимодействие объектов внутри коллекции.          Переопределение метода hashCode.</p>	<p>Коллекция HashSet. Коллекция Set. Основные принципы работы с Hash-кодом объекта. Взаимодействие объектов внутри коллекции.          Переопределение метода hashCode.          Изучение материала. Подготовка к практическим занятиям и тестированию.</p>
37	<p>Коллекция HashMap.          Коллекция Map и ее реализация на основе класса HashMap.          Внутреннее устройство коллекции.</p>	<p>Коллекция HashMap. Коллекция Map и ее реализация на основе класса HashMap. Внутреннее устройство коллекции.          Изучение материала. Подготовка к практическим занятиям и тестированию.</p>
38	<p>Коллекции TreeSet.          Методы взаимодействия с коллекцией Set.          Структуры для хранения данных: бинарное дерево, красно-черное дерево.          Внутреннее взаимодействие между объектами внутри коллекции.</p>	<p>Коллекции TreeSet. Методы взаимодействия с коллекцией Set. Структуры для хранения данных: бинарное дерево, красно-черное дерево. Внутреннее взаимодействие между объектами внутри коллекции.          Изучение материала. Подготовка к практическим занятиям и тестированию.</p>
39	<p>Коллекция TreeMap.          Методы взаимодействия с коллекцией Map.          Реализация коллекции на основе класса TreeMap. Внутреннее устройство коллекции в формате ключ-значение.</p>	<p>Коллекция TreeMap. Методы взаимодействия с коллекцией Map. Реализация коллекции на основе класса TreeMap. Внутреннее устройство коллекции в формате ключ-значение.          Изучение материала. Подготовка к практическим занятиям и тестированию.</p>

## 6. Методические материалы для самостоятельной работы обучающихся по дисциплине (модулю)

Программой дисциплины предусмотрено проведение лекционных занятий, на которых дается основной систематизированный материал, и практических занятий, предполагающих закрепление изученного материала и формирование у обучающихся необходимых знаний, умений и навыков. Кроме того, важнейшим этапом изучения дисциплины является самостоятельная работа обучающихся с использованием всех средств и возможностей современных образовательных технологий.

В объем самостоятельной работы по дисциплине включается следующее:

- изучение теоретических вопросов по всем темам дисциплины;
- выполнение курсовой работы;
- подготовка к практическим занятиям;
- подготовка к текущему контролю успеваемости студентов;
- подготовка к зачету, зачету с оценкой и экзамену.

Залогом успешного освоения дисциплины является обязательное посещение лекционных и лабораторных занятий, так как пропуск одного (тем более, нескольких) занятий может осложнить освоение разделов курса. На лабораторных занятиях материал, изложенный на лекциях, закрепляется при выполнении лабораторных заданий.

Приступая к изучению дисциплины, необходимо в первую очередь ознакомиться с содержанием РПД, а также методическими указаниями по организации самостоятельной работы.

При подготовке к лекционным занятиям студенту необходимо:

- ознакомиться с соответствующей темой занятия;
- осмыслить круг изучаемых вопросов и логику их рассмотрения;
- изучить рекомендуемую рабочей программой литературу по данной теме.

При подготовке к практическим занятиям и в рамках самостоятельной работы по изучению дисциплины обучающимся необходимо:

- повторить законспектированный на лекционном занятии материал и дополнить его с учетом рекомендованной по данной теме литературы;
- при самостоятельном изучении теоретической темы сделать конспект, используя рекомендованные в РПД источники;
- выполнить практические задания в рамках изучаемой темы;
- ответить на контрольные вопросы по теме, используя материалы ФОС;
- подготовиться к проверочной работе, предусмотренной в контрольных точках;
- подготовиться к промежуточной аттестации.

Итогом изучения дисциплины является зачет, зачет с оценкой и экзамен. Зачет, зачет с оценкой и Экзамен проводится по расписанию. Форма проведения занятия может быть устная, письменная и в электронном виде. Студенты, не прошедшие аттестацию, должны ликвидировать задолженность в установленном порядке.

## 7. Оценочные материалы для проведения текущего контроля успеваемости и промежуточной аттестации обучающихся по дисциплине (модулю)

### 7.1. Перечень компетенций с указанием этапов их формирования в процессе освоения дисциплины

№ п/п	Контролируемые разделы дисциплины (модуля)	Код и наименование индикатора контролируемой компетенции	Вид оценочного средства
24	Введение в Java. Знакомство с JVM, JRE, JDK. Сравнительный анализ Java и других языков программирования.	ОПК-5.3, ПК-1.1, ПК-1.2, ПК-1.3	Практические задания, тесты
25	Структура программы на языке Java. Импорт пакетов. Консольный ввод/вывод данных.	ОПК-5.3, ПК-1.1, ПК-1.2, ПК-1.3	Практические задания, тесты
26	Организация вычислений. Циклы. Условные операторы. Использование ссылочных и примитивных типов данных.	ОПК-5.3, ПК-1.1, ПК-1.2, ПК-1.3	Практические задания, тесты

27	Типы данных в Java.	ОПК-5.3, ПК-1.1, ПК-1.2, ПК-1.3	Практические задания, тесты
28	Объектно-ориентированные возможности языка Java. Структура класса. Уровни доступа к полям класса. Принципы работы с объектами. Паттерн проектирования Singleton.	ОПК-5.3, ПК-1.1, ПК-1.2, ПК-1.3	Практические задания, тесты
29	Одномерные и многомерные массивы. Метод clone. Класс Arrays.	ОПК-5.3, ПК-1.1, ПК-1.2, ПК-1.3	Практические задания, тесты
30	Наследование. Использование конструктора родительского класса внутри дочернего класса.	ОПК-5.3, ПК-1.1, ПК-1.2, ПК-1.3	Практические задания, тесты
31	Полиморфизм. Абстракции и интерфейсы. Абстрактные классы. Паттерн проектирования Абстрактная фабрика. Паттерн проектирования Фабричный метод.	ОПК-5.3, ПК-1.1, ПК-1.2, ПК-1.3	Практические задания, тесты
32	Исключения. Использование исключений при разработке собственных библиотек. Stacktrace.	ОПК-5.3, ПК-1.1, ПК-1.2, ПК-1.3	Практические задания, тесты
33	Generic классы. Способы организации. Методы Generic классов.	ОПК-5.3, ПК-1.1, ПК-1.2, ПК-1.3	Практические задания, тесты
34	Коллекции List. Основные методы. Организация хранения данных с помощью динамического массива и связанного списка. Классы ArrayList и LinkedList.	ОПК-5.3, ПК-1.1, ПК-1.2, ПК-1.3	Практические задания, тесты
35	Коллекции Queue и Stack. Основные методы. Организация хранения данных с помощью динамического массива и связанного списка на базе очереди и стека. Отличие интерфейсов Queue и Stack.	ОПК-5.3, ПК-1.1, ПК-1.2, ПК-1.3	Практические задания, тесты
36	Коллекция HashSet. Коллекция Set. Основные принципы работы с Hash- кодом объекта. Взаимодействие объектов внутри коллекции. Переопределение метода hashCode.	ОПК-5.3, ПК-1.1, ПК-1.2, ПК-1.3	Практические задания, тесты
37	Коллекция HashMap. Коллекция Map и ее реализация на основе класса HashMap. Внутреннее устройство коллекции.	ОПК-5.3, ПК-1.1, ПК-1.2, ПК-1.3	Практические задания, тесты
38	Коллекции TreeSet. Методы взаимодействия с коллекцией Set. Структуры для хранения данных: бинарное дерево, красно-черное дерево. Внутреннее взаимодействие между объектами внутри коллекции.	ОПК-5.3, ПК-1.1, ПК-1.2, ПК-1.3	Практические задания, тесты
39	Коллекция TreeMap. Методы взаимодействия с коллекцией Map. Реализация коллекции на основе класса TreeMap. Внутреннее устройство коллекции в формате ключ-значение.	ОПК-5.3, ПК-1.1, ПК-1.2, ПК-1.3	Практические задания, тесты
40	Иная контактная работа	ОПК-5.3, ПК-1.1, ПК-1.2, ПК-1.3	
41	Экзамен	ОПК-5.3, ПК-1.1, ПК-1.2,	

7.2. Типовые контрольные задания или иные материалы текущего контроля успеваемости, необходимые для оценки знаний, умений и навыков и (или) опыта профессиональной деятельности, характеризующих этапы формирования компетенций в процессе освоения дисциплины

Комплект заданий для проверки сформированности индикаторов достижений компетенций ПК-1.1, ПК-1.2, ПК-1.3, ОПК-5.3 размещены в ЭИОС Moodle по адресу <https://moodle.spbgasu.ru/enrol/index.php?id=555>

7.3. Система оценивания результатов обучения по дисциплине (модулю) при проведении текущего контроля успеваемости

<p>Оценка «отлично» (зачтено)</p>	<p>знания:</p> <ul style="list-style-type: none"> <li>- систематизированные, глубокие и полные знания по всем разделам дисциплины, а также по основным вопросам, выходящим за пределы учебной программы;</li> <li>- точное использование научной терминологии, систематически грамотное и логически правильное изложение ответа на вопросы;</li> <li>- полное и глубокое усвоение основной и дополнительной литературы, рекомендованной рабочей программой по дисциплине (модулю)</li> </ul> <p>умения:</p> <ul style="list-style-type: none"> <li>- умеет ориентироваться в теориях, концепциях и направлениях дисциплины и давать им критическую оценку, используя научные достижения других дисциплин</li> </ul> <p>навыки:</p> <ul style="list-style-type: none"> <li>- высокий уровень сформированности заявленных в рабочей программе компетенций;</li> <li>- владеет навыками самостоятельно и творчески решать сложные проблемы и нестандартные ситуации;</li> <li>- применяет теоретические знания для выбора методики выполнения заданий;</li> <li>- грамотно обосновывает ход решения задач;</li> <li>- безупречно владеет инструментарием учебной дисциплины, умение его эффективно использовать в постановке научных и практических задач;</li> <li>- творческая самостоятельная работа на практических/семинарских/лабораторных занятиях, активно участвует в групповых обсуждениях, высокий уровень культуры исполнения заданий</li> </ul>
<p>Оценка «хорошо» (зачтено)</p>	<p>знания:</p> <ul style="list-style-type: none"> <li>- достаточно полные и систематизированные знания по дисциплине;</li> <li>- усвоение основной и дополнительной литературы, рекомендованной рабочей программой по дисциплине (модулю)</li> </ul> <p>умения:</p> <ul style="list-style-type: none"> <li>- умеет ориентироваться в основных теориях, концепциях и направлениях дисциплины и давать им критическую оценку;</li> <li>- использует научную терминологию, лингвистически и логически правильно излагает ответы на вопросы, умеет делать обоснованные выводы;</li> <li>- владеет инструментарием по дисциплине, умение его использовать в постановке и решении научных и профессиональных задач</li> </ul> <p>навыки:</p> <ul style="list-style-type: none"> <li>- самостоятельная работа на практических занятиях, участие в групповых обсуждениях, высокий уровень культуры исполнения заданий;</li> <li>- средний уровень сформированности заявленных в рабочей программе компетенций;</li> <li>- без затруднений выбирает стандартную методику выполнения заданий;</li> <li>- обосновывает ход решения задач без затруднений</li> </ul>

<p>Оценка «удовлетворительно» (зачтено)</p>	<p>знания: - достаточный минимальный объем знаний по дисциплине; - усвоение основной литературы, рекомендованной рабочей программой; - использование научной терминологии, стилистическое и логическое изложение ответа на вопросы, умение делать выводы без существенных ошибок умения: - умеет ориентироваться в основных теориях, концепциях и направлениях по дисциплине и давать им оценку; - владеет инструментарием учебной дисциплины, умение его использовать в решении типовых задач; - умеет под руководством преподавателя решать стандартные задачи навыки: - работа под руководством преподавателя на практических занятиях, допустимый уровень культуры исполнения заданий; - достаточный минимальный уровень сформированности заявленных в рабочей программе компетенций; - испытывает затруднения при обосновании алгоритма выполнения заданий</p>
<p>Оценка «неудовлетворительно» (не зачтено)</p>	<p>знания: - фрагментарные знания по дисциплине; - отказ от ответа (выполнения письменной работы); - знание отдельных источников, рекомендованных рабочей программой по дисциплине; умения: - не умеет использовать научную терминологию; - наличие грубых ошибок навыки: - низкий уровень культуры исполнения заданий; - низкий уровень сформированности заявленных в рабочей программе компетенций; - отсутствие навыков самостоятельной работы; - не может обосновать алгоритм выполнения заданий</p>

7.4. Теоретические вопросы и практические задания для проведения промежуточной аттестации обучающихся, необходимые для оценки знаний, умений и навыков и (или) опыта профессиональной деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы

7.4.1. Теоретические вопросы для проведения промежуточной аттестации обучающихся

Примерные теоретические вопросы для проведения промежуточной аттестации ( 5 семестр ):

1. Особенности языка Java. JVM, JRE, JDK. Сравнительный анализ Java и других языков программирования.
2. Структура программы на языке Java. Импорт пакетов. Консольный ввод/вывод.
3. Организация вычислений. Циклы. Условные операторы. Использование ссылочных и примитивных типов данных.
4. Типы данных языка Java.
5. Объектно-ориентированные возможности языка Java. Структура класса. Уровни доступа к полям класса. Принципы работы с объектами. Паттерн проектирования Singleton.
6. Одномерные и многомерные массивы. Метод clone. Класс Arrays и способы его использования при работе с массивами.
7. Наследование. Принципы грамотного наследования и способы его организации. Использование конструктора родительского класса внутри дочернего класса.
8. Полиморфизм. Абстракции и интерфейсы. Абстрактные классы. Абстрактные интерфейсы. Паттерн проектирования Абстрактная фабрика. Паттерн проектирования Фабричный метод.
9. Исключения. Дерево наследования исключений в Java. Использование исключений

при разработке собственных библиотек. Stacktrace.

10. Generic классы. Способы организации. Методы Generic классов.

11. Коллекции List. Основные методы. Организация хранения данных с помощью динамического массива и связанного списка. Классы ArrayList и LinkedList.

12. Коллекции Queue и Stack. Основные методы. Организация хранения данных с помощью динамического массива и связанного списка на базе очереди и стека. Отличие интерфейсов Queue и Stack.

13. Коллекция HashSet. Коллекция Set. Основные принципы работы с Hash-кодом объекта. Взаимодействие объектов внутри коллекции. Переопределение метода hashCode.

14. Коллекция HashMap. Коллекция Map и ее реализация на основе класса HashMap. Внутреннее устройство коллекции.

15. Коллекции TreeSet. Методы взаимодействия с коллекцией Set. Структуры для хранения данных: бинарное дерево, красно-черное дерево. Внутреннее взаимодействие между объектами внутри коллекции.

16. Коллекция TreeMap. Методы взаимодействия с коллекцией Map. Реализация коллекции на основе класса TreeMap. Внутреннее устройство коллекции в формате ключ-значение.

#### 7.4.2. Практические задания для проведения промежуточной аттестации обучающихся

Практические задания для проведения промежуточной аттестации обучающихся размещены по адресу ЭИОС Moodle (<https://moodle.spbgasu.ru/enrol/index.php?id=555>)

#### 7.4.3. Примерные темы курсовой работы (проекта) (при наличии)

Курсовая работа (КР):

Разработка собственной библиотеки

Описание:

Вам предстоит разработать систему классов (один или несколько) для реализации задач, связанных с темой вашего варианта. При разработке вам разрешено пользоваться такими готовыми классами, как:

- Имплементации интерфейсов Queue, Stack, List, Set и Map;
- Arrays;
- String ;
- StringBuilder и StringBuffer.

Требования:

• Библиотека должна решать большинство задач, связанных с вашим вариантом (Можете проконсультироваться с преподавателем).

• Классы должны иметь как статические, так и нестатические поля и методы.

• Система классов, по возможности, должна демонстрировать основные принципы объектно-ориентированного программирования и содержать Generic классы.

• Количество методов не ограничено, но минимум их должно быть 10.

Варианты заданий:

1. Библиотека для работы с матрицами;
2. Библиотека для работы с графами;
3. Библиотека для решения ЗЛП.
4. Библиотека для парсинга сайтов (по полученному html);
5. Библиотека для построения графиков

#### 7.5. Методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта профессиональной деятельности, характеризующие этапы формирования компетенций

Процедура проведения промежуточной аттестации и текущего контроля успеваемости регламентируется локальным нормативным актом, определяющим порядок организации и проведения текущего контроля успеваемости и промежуточной аттестации обучающихся.

Процедура оценивания формирования компетенций при проведении текущего контроля приведена в п. 7.2. Типовые контрольные задания или иные материалы текущего контроля приведены в п. 7.3. Промежуточная аттестация по дисциплине проводится в форме зачета, зачета с оценкой и экзамена.

7.6. Критерии оценивания сформированности компетенций при проведении промежуточной аттестации

Критерии оценивания	Уровень освоения и оценка			
	Оценка «неудовлетворительно»	Оценка «удовлетворительно»	Оценка «хорошо»	Оценка «отлично»
	«не зачтено»	«зачтено»		
	Уровень освоения компетенции «недостаточный». Компетенции не сформированы. Знания отсутствуют, умения и навыки не сформированы	Уровень освоения компетенции «пороговый». Компетенции сформированы. Сформированы базовые структуры знаний. Умения фрагментарны и носят репродуктивный характер. Демонстрируется низкий уровень самостоятельности практического навыка.	Уровень освоения компетенции «продвинутой». Компетенции сформированы. Знания обширные, системные. Умения носят репродуктивный характер, применяются к решению типовых заданий. Демонстрируется достаточный уровень самостоятельности устойчивого практического навыка.	Уровень освоения компетенции «высокий». Компетенции сформированы. Знания аргументированные, всесторонние. Умения успешно применяются к решению как типовых, так и нестандартных творческих заданий. Демонстрируется высокий уровень самостоятельности, высокая адаптивность практического навыка

знания	<p>Обучающийся демонстрирует:</p> <ul style="list-style-type: none"> <li>-существенные пробелы в знаниях учебного материала;</li> <li>-допускаются принципиальные ошибки при ответе на основные вопросы билета, отсутствует знание и понимание основных понятий и категорий;</li> <li>-непонимание сущности дополнительных вопросов в рамках заданий билета.</li> </ul>	<p>Обучающийся демонстрирует:</p> <ul style="list-style-type: none"> <li>-знания теоретического материала;</li> <li>-неполные ответы на основные вопросы, ошибки в ответе, недостаточное понимание сущности излагаемых вопросов;</li> <li>-неуверенные и неточные ответы на дополнительные вопросы.</li> </ul>	<p>Обучающийся демонстрирует:</p> <ul style="list-style-type: none"> <li>-знание и понимание основных вопросов контролируемого объема программного материала;</li> <li>- знания теоретического материала</li> <li>-способность устанавливать и объяснять связь практики и теории, выявлять противоречия, проблемы и тенденции развития;</li> <li>-правильные и конкретные, без грубых ошибок, ответы на поставленные вопросы.</li> </ul>	<p>Обучающийся демонстрирует:</p> <ul style="list-style-type: none"> <li>-глубокие, всесторонние и аргументированные знания программного материала;</li> <li>-полное понимание сущности и взаимосвязи рассматриваемых процессов и явлений, точное знание основных понятий, в рамках обсуждаемых заданий;</li> <li>-способность устанавливать и объяснять связь практики и теории,</li> <li>-логически последовательные, содержательные, конкретные и исчерпывающие ответы на все задания билета, а также дополнительные вопросы экзаменатора.</li> </ul>
умения	<p>При выполнении практического задания билета обучающийся продемонстрировал недостаточный уровень умений. Практические задания не выполнены. Обучающийся не отвечает на вопросы билета при дополнительных наводящих вопросах преподавателя.</p>	<p>Обучающийся выполнил практическое задание билета с существенными неточностями. Допускаются ошибки в содержании ответа и решении практических заданий. При ответах на дополнительные вопросы было допущено много неточностей.</p>	<p>Обучающийся выполнил практическое задание билета с небольшими неточностями. Показал хорошие умения в рамках освоенного учебного материала. Предложенные практические задания решены с небольшими неточностями. Ответил на большинство дополнительных вопросов.</p>	<p>Обучающийся правильно выполнил практическое задание билета. Показал отличные умения в рамках освоенного учебного материала. Решает предложенные практические задания без ошибок. Ответил на все дополнительные вопросы.</p>

владение навыками	Не может выбрать методику выполнения заданий. Допускает грубые ошибки при выполнении заданий, нарушающие логику решения задач. Делает некорректные выводы. Не может обосновать алгоритм выполнения заданий.	Испытывает затруднения по выбору методики выполнения заданий. Допускает ошибки при выполнении заданий, нарушения логики решения задач. Испытывает затруднения с формулированием корректных выводов. Испытывает затруднения при обосновании алгоритма выполнения заданий.	Без затруднений выбирает стандартную методику выполнения заданий. Допускает ошибки при выполнении заданий, не нарушающие логику решения задач. Делает корректные выводы по результатам решения задачи. Обосновывает ход решения задач без затруднений.	Применяет теоретические знания для выбора методики выполнения заданий. Не допускает ошибок при выполнении заданий. Самостоятельно анализирует результаты выполнения заданий. Грамотно обосновывает ход решения задач.
-------------------	---	--	--	---

Оценка по дисциплине зависит от уровня сформированности компетенций, закрепленных за дисциплиной, и представляет собой среднее арифметическое от выставленных оценок по отдельным результатам обучения (знания, умения, владение навыками).

Оценка «отлично»/«зачтено» выставляется, если среднее арифметическое находится в интервале от 4,5 до 5,0.

Оценка «хорошо»/«зачтено» выставляется, если среднее арифметическое находится в интервале от 3,5 до 4,4.

Оценка «удовлетворительно»/«зачтено» выставляется, если среднее арифметическое находится в интервале от 2,5 до 3,4.

Оценка «неудовлетворительно»/«не зачтено» выставляется, если среднее арифметическое находится в интервале от 0 до 2,4.

## 8. Учебно-методическое и материально-техническое обеспечение дисциплины (модуля)

### 8.1. Перечень основной и дополнительной литературы, необходимой для освоения дисциплины (модуля)

№ п/п	Автор, название, место издания, издательство, год издания учебной и учебно-методической литературы	Количество экземпляров/электронный адрес ЭБС
<b><u>Основная литература</u></b>		
1	Сорокин А. А., Объектно-ориентированное программирование, Ставрополь: Северо-Кавказский федеральный университет, 2014	<a href="http://www.iprbookshop.ru/63110.html">http://www.iprbookshop.ru/63110.html</a>
2	Джошуа Блох, Стрельцов В., Усманов Р., Java. Эффективное программирование, Саратов: Профобразование, 2017	<a href="http://www.iprbookshop.ru/64057.html">http://www.iprbookshop.ru/64057.html</a>
3	Вязовик Н. А., Программирование на Java, Москва: Интернет- Университет Информационных Технологий (ИНТУИТ), 2016	<a href="http://www.iprbookshop.ru/73710.html">http://www.iprbookshop.ru/73710.html</a>
4	Букунов С. В., Букунова О. В., Создание объектно-ориентированных приложений с использованием стандартных библиотек классов. Библиотека STL, СПб., 2018	<a href="http://ntb.spbgasu.ru/elib/00914/">http://ntb.spbgasu.ru/elib/00914/</a>
5	Букунов С. В., Букунова О. В., Основы объектно-ориентированного программирования, СПб., 2017	<a href="http://ntb.spbgasu.ru/elib/00799/">http://ntb.spbgasu.ru/elib/00799/</a>
<b><u>Дополнительная литература</u></b>		

1	Лисицин Д. В., Объектно-ориентированное программирование, Новосибирск: Новосибирский государственный технический университет, 2010	<a href="http://www.iprbookshop.ru/44970.html">http://www.iprbookshop.ru/44970.html</a>
1	Новиков П. В., Объектно-ориентированное программирование, Саратов: Вузовское образование, 2017	<a href="http://www.iprbookshop.ru/64650.html">http://www.iprbookshop.ru/64650.html</a>

Обучающиеся из числа инвалидов и лиц с ОВЗ обеспечиваются печатными и (или) электронными образовательными ресурсами в формах, адаптированных к ограничениям их здоровья.

8.2. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины (модуля)

Наименование ресурса сети «Интернет»	Электронный адрес ресурса
Система дистанционного обучения СПбГАСУ Moodle	<a href="https://moodle.spbgasu.ru/">https://moodle.spbgasu.ru/</a>

8.3. Перечень современных профессиональных баз данных и информационных справочных систем

Наименование	Электронный адрес ресурса
Электронно-библиотечная система издательства "Лань"	<a href="https://e.lanbook.com/">https://e.lanbook.com/</a>
Электронно-библиотечная система издательства "ЮРАЙТ"	<a href="https://www.biblio-online.ru/">https://www.biblio-online.ru/</a>
Электронно-библиотечная система издательства "IPRsmart"	<a href="http://www.iprbookshop.ru/">http://www.iprbookshop.ru/</a>

8.4. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень лицензионного и свободно распространяемого программного обеспечения

Наименование	Способ распространения (лицензионное или свободно распространяемое)
Microsoft Visual Studio Community Edition	Свободно распространяемое
Qt Creator	Свободно распространяемое

8.5. Материально-техническое обеспечение дисциплины

Сведения об оснащённости учебных аудиторий и помещений для самостоятельной работы

Наименование учебных аудиторий и помещений для самостоятельной работы	Оснащённость оборудованием и техническими средствами обучения
73. Учебные аудитории для проведения практических занятий, курсового проектирования (выполнения курсовых работ), групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации	Учебная аудитория для проведения практических занятий, курсового проектирования (выполнения курсовых работ), групповых и индивидуальных консультаций, текущего контроля и промежуточной аттестации – комплект мультимедийного оборудования (персональный компьютер, мультимедийный проектор, экран, аудиосистема), доска маркерная белая эмалевая, комплект учебной мебели, подключение к компьютерной сети СПбГАСУ, выход в Интернет.
73. Помещения для самостоятельной работы	Помещение для самостоятельной работы (читальный зал библиотеки, ауд. 217): ПК-23 шт., в т.ч. 1 шт.- ПК для лиц с ОВЗ (системный блок, монитор, клавиатура, мышь) с подключением к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду СПбГАСУ. ПО Microsoft Windows 10

73. Учебные аудитории для проведения лекционных занятий	Учебная аудитория для проведения занятий лекционного типа, комплект мультимедийного оборудования (персональный компьютер, мультимедийный проектор, экран, аудиосистема), доска маркерная белая эмалевая, экран, комплект учебной мебели, подключение к компьютерной сети СПбГАСУ, выход в Интернет.
73. Компьютерный класс	Рабочие места с ПК (стол компьютерный, системный блок, монитор, клавиатура, мышь), стол рабочий, подключение к компьютерной сети СПбГАСУ, выход в Internet.

Для инвалидов и лиц с ОВЗ обеспечиваются специальные условия для получения образования в соответствии с требованиями нормативно-правовых документов.