

**МИНИСТЕРСТВО ТРАНСПОРТА РОССИЙСКОЙ ФЕДЕРАЦИИ**  
**ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ**  
**УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ**  
**«РОССИЙСКИЙ УНИВЕРСИТЕТ ТРАНСПОРТА»**  
**(РУТ (МИИТ))**



Рабочая программа дисциплины (модуля),  
как компонент образовательной программы  
высшего образования - программы бакалавриата  
по направлению подготовки  
09.03.01 Информатика и вычислительная техника,  
утвержденной первым проректором РУТ (МИИТ)  
Тимониным В.С.

**РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)**

**Программирование**

Направление подготовки: 09.03.01 Информатика и вычислительная техника

Направленность (профиль): Технологии разработки программного обеспечения

Форма обучения: Очная

Рабочая программа дисциплины (модуля) в виде  
электронного документа выгружена из единой  
корпоративной информационной системы управления  
университетом и соответствует оригиналу

Простая электронная подпись, выданная РУТ (МИИТ)  
ID подписи: 5665  
Подписал: заведующий кафедрой Нутович Вероника  
Евгеньевна  
Дата: 01.09.2024

## 1. Общие сведения о дисциплине (модуле).

Целями освоения дисциплины «Программирование» являются изучение основ прикладного и системного программирования, средств создания программного обеспечения для решения множества научных, прикладных, деловых, административных, математических и др. задач. В качестве языка программирования принят язык Java версии 17, для разработки используется интегрированная среда JetBrains IntelliJ IDEA, изучение и использование которой рассматривается в данном курсе.

Задачей дисциплины является обучить студентов современным подходам к написанию программных продуктов с использованием объекто-ориентированного подхода, а также современные особенности языка и различные технологии и фреймворки, актуальные при разработке современных программных продуктов на данном языке.

## 2. Планируемые результаты обучения по дисциплине (модулю).

Перечень формируемых результатов освоения образовательной программы (компетенций) в результате обучения по дисциплине (модулю):

**ОПК-5** - Способен устанавливать программное и аппаратное обеспечение для информационных и автоматизированных систем;

**ОПК-8** - Способен разрабатывать алгоритмы и программы, пригодные для практического применения;

**ОПК-9** - Способен осваивать методики использования программных средств для решения практических задач.

Обучение по дисциплине (модулю) предполагает, что по его результатам обучающийся будет:

### **Знать:**

- синтаксис, особенности программирования и стандартные библиотеки выбранного языка программирования;
- конструкции параллельного программирования;
- отличия языков программирования;
- основные типы данных и базовые языковые конструкции;
- принципы объектно-ориентированного программирования;
- основные методы программирования;
- основные парадигмы программирования.

### **Уметь:**

- применять языки программирования, для написания программного кода;
- создавать блок-схемы алгоритмов функционирования разрабатываемых программных продуктов;
- осуществлять отладку программных продуктов;
- применять принципы объектно-ориентированного программирования с помощью выбранного языка программирования;
- использовать стандартные библиотеки для написания программного обеспечения;
- применять технологии и языковые конструкции параллельного программирования.

#### **Владеть:**

- навыками изучения технической документации по языку программирования;
- навыками применения информации из технической документации по языку программирования при написании программного обеспечения;
- навыками разработки блок-схемы алгоритма разрабатываемого программного обеспечения в соответствии с техническим заданием;
- навыками написания исходного кода программного обеспечения;
- навыками отладки разработанного кода программного обеспечения;
- навыками применения различных методов программирования при реализации программного обеспечения;
- навыками применения различных парадигм программирования при разработке программного обеспечения;
- навыками применения различных стандартных библиотек для реализации алгоритма программного обеспечения;
- навыками внедрения технологий и языковых конструкций параллельного программирования при реализации программного обеспечения.

### **3. Объем дисциплины (модуля).**

#### **3.1. Общая трудоемкость дисциплины (модуля).**

Общая трудоемкость дисциплины (модуля) составляет 9 з.е. (324 академических часа(ов)).

3.2. Объем дисциплины (модуля) в форме контактной работы обучающихся с педагогическими работниками и (или) лицами,

привлекаемыми к реализации образовательной программы на иных условиях, при проведении учебных занятий:

Тип учебных занятий	Количество часов		
	Всего	Семестр	
		№1	№2
Контактная работа при проведении учебных занятий (всего):	128	64	64
В том числе:			
Занятия лекционного типа	64	32	32
Занятия семинарского типа	64	32	32

3.3. Объем дисциплины (модуля) в форме самостоятельной работы обучающихся, а также в форме контактной работы обучающихся с педагогическими работниками и (или) лицами, привлекаемыми к реализации образовательной программы на иных условиях, при проведении промежуточной аттестации составляет 196 академических часа (ов).

3.4. При обучении по индивидуальному учебному плану, в том числе при ускоренном обучении, объем дисциплины (модуля) может быть реализован полностью в форме самостоятельной работы обучающихся, а также в форме контактной работы обучающихся с педагогическими работниками и (или) лицами, привлекаемыми к реализации образовательной программы на иных условиях, при проведении промежуточной аттестации.

#### 4. Содержание дисциплины (модуля).

##### 4.1. Занятия лекционного типа.

№ п/п	Тематика лекционных занятий / краткое содержание
1	Алгоритмизация. Рассматриваемые вопросы: - понятие алгоритма; - основные понятия алгоритмизации; - формы записи алгоритмов; - основные алгоритмические конструкции; - блок-схемы; - псевдокод; - виды алгоритмов; - стратегии; - сортировка; - поиск; - графы.
2	Языки программирования. Рассматриваемые вопросы:

№ п/п	Тематика лекционных занятий / краткое содержание
	<ul style="list-style-type: none"> <li>- типы языков программирования;</li> <li>- поколения языков программирования;</li> <li>- отличия языков программирования;</li> <li>- методы программирования;</li> <li>- языки программирования и задачи профессиональной деятельности;</li> <li>- типы приложений.</li> </ul>
3	<p><b>Основы программирования.</b></p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> <li>- лексика языка;</li> <li>- переменные и константы;</li> <li>- типы данных;</li> <li>- выражения и операции;</li> <li>- логические конструкции и ветвления;</li> <li>- конструкция if/else;</li> <li>- вложенные условия;</li> <li>- конструкция switch/case;</li> <li>- циклы;</li> <li>- цикл for;</li> <li>- цикл while;</li> <li>- цикл foreach;</li> <li>- бесконечный цикл;</li> <li>- вложенные циклы.</li> </ul>
4	<p><b>Процедурное программирование.</b></p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> <li>- процедуры и функции;</li> <li>- рекурсия и рекурсивные функции;</li> <li>- массивы;</li> <li>- массивы в памяти;</li> <li>- работа с массивами;</li> <li>- строки;</li> <li>- разбор строк и регулярные выражения;</li> <li>- работа с вводом/выводом;</li> <li>- методы файлового ввода/вывода.</li> </ul>
5	<p><b>Объектно-ориентированное программирование.</b></p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> <li>- введение в объектно-ориентированное программирование;</li> <li>- основные особенности ООП;</li> <li>- объектно-ориентированные языки программирования;</li> <li>- принципы объектно-ориентированного программирования;</li> <li>- абстракция;</li> <li>- наследование;</li> <li>- полиморфизм;</li> <li>- инкапсуляция;</li> <li>- отношения между классами.</li> </ul>
6	<p><b>ООП Java.</b></p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> <li>- объекты и классы;</li> <li>- класс Object;</li> <li>- поля;</li> <li>- геттеры и сеттеры;</li> <li>- свойства;</li> </ul>

№ п/п	Тематика лекционных занятий / краткое содержание
	<ul style="list-style-type: none"> <li>- конструкторы;</li> <li>- типы конструкторов;</li> <li>- методы;</li> <li>- модификаторы доступа;</li> <li>- области видимости;</li> <li>- пакеты;</li> <li>- интерфейсы;</li> <li>- модификаторы в интерфейсах;</li> <li>- обратные вызовы;</li> <li>- вложенные классы и интерфейсы.</li> </ul>
7	<b>Исключения.</b> Рассматриваемые вопросы: <ul style="list-style-type: none"> <li>- исключения;</li> <li>- иерархия исключений;</li> <li>- конструкция try/catch/finally;</li> <li>- выброс и обработка исключений.</li> </ul>
8	<b>Возможности Java.</b> Рассматриваемые вопросы: <ul style="list-style-type: none"> <li>- Enum;</li> <li>- Record;</li> <li>- обобщения (Generics);</li> <li>- работа с датами и большими числами.</li> </ul>
9	<b>Коллекции Java.</b> Рассматриваемые вопросы: <ul style="list-style-type: none"> <li>- типы коллекций;</li> <li>- структура Java Collection Framework;</li> <li>- интерфейс Collection;</li> <li>- интерфейс Map;</li> <li>- коллекции реализующие интерфейс Map;</li> <li>- интерфейс Set;</li> <li>- коллекции реализующие интерфейс Set;</li> <li>- интерфейс Queue;</li> <li>- коллекции реализующие интерфейс Queue;</li> <li>- интерфейс List;</li> <li>- коллекции реализующие интерфейс List;</li> <li>- интерфейсы Comparable и Comparator;</li> <li>- сортировка;</li> <li>- итераторы.</li> </ul>
10	<b>Многопоточность в Java.</b> Рассматриваемые вопросы: <ul style="list-style-type: none"> <li>- потоки;</li> <li>- создание и запуск потоков;</li> <li>- завершение и прерывание потоков;</li> <li>- синхронизация;</li> <li>- коллекции пакета java.util.concurrent.</li> </ul>
11	<b>Функциональное программирование в Java.</b> Рассматриваемые вопросы: <ul style="list-style-type: none"> <li>- императивная и декларативная парадигмы;</li> <li>- лямбда-выражения;</li> <li>- функциональные интерфейсы;</li> <li>- сильные и слабые стороны функционального подхода.</li> </ul>

№ п/п	Тематика лекционных занятий / краткое содержание
12	<p>Современная Java.</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> <li>- сериализация и десериализация;</li> <li>- Stream API;</li> <li>- модульность;</li> <li>- многострочные строки;</li> <li>- sealed-классы;</li> <li>- sealed-интерфейсы;</li> <li>- обновления в конструкции switch/case;</li> <li>- паттерн-матчинг;</li> <li>- будущее языка.</li> </ul>
13	<p>Spring.</p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> <li>- основы web;</li> <li>- протокол http;</li> <li>- URL;</li> <li>- MIME;</li> <li>- управление сессией;</li> <li>- cookies;</li> <li>- MVC;</li> <li>- компоненты Spring;</li> <li>- Spring Boot;</li> <li>- стартеры;</li> <li>- Bean;</li> <li>- Spring Inversion of Control;</li> <li>- инъекция зависимости;</li> <li>- области видимости bean;</li> <li>- жизненный цикл bean;</li> <li>- web-сервис;</li> <li>- JSON;</li> <li>- REST API.</li> </ul>

## 4.2. Занятия семинарского типа.

### Лабораторные работы

№ п/п	Наименование лабораторных работ / краткое содержание
1	<p>Java. Основы языка.</p> <p>В результате выполнения лабораторных работ студент получает навыки работы с базовыми конструкциями, типами данных и синтаксическими особенностями языка Java.</p>
2	<p>Java. Циклы.</p> <p>В результате выполнения лабораторных работ студент получает навыки работы с операторами циклов (while, for, foreach) и их отличительными особенностями на языке Java.</p>
3	<p>Java. Вложенные циклы.</p> <p>В результате выполнения лабораторных работ студент получает навыки работы с построением алгоритмов с использованием вложенных циклов и использование бесконечных циклов на языке Java.</p>

№ п/п	Наименование лабораторных работ / краткое содержание
4	Java. Массивы. В результате выполнения лабораторных работ студент получает навыки работы с массивами на языке Java.
5	Java. Работа с файловым вводом-выводом. В результате выполнения лабораторных работ студент получает навыки работы с файловой системой и операциями файлового ввода-вывода, работы с различными типами файлов на языке Java.
6	Java. ООП. В результате выполнения лабораторных работ студент получает навыки разработки алгоритмов программ с применением объектно-ориентированной парадигмы программирования на языке Java, применяя теоретические знания в области принципов объектно-ориентированного подхода (абстракция, наследование, полиморфизм, инкапсуляция).
7	Java. Коллекции. В результате выполнения лабораторных работ студент получает навыки работы с коллекциями, реализующими интерфейс List на языке Java. В результате выполнения лабораторных работ студент получает навыки работы с коллекциями, реализующими интерфейс Set на языке Java. В результате выполнения лабораторных работ студент получает навыки работы с коллекциями, реализующими интерфейс Map на языке Java. В результате выполнения лабораторных работ студент получает навыки работы с коллекциями, реализующими интерфейс Queue на языке Java.
8	Java. Сериализация и десериализация. В результате выполнения лабораторных работ студент получает навыки разработки алгоритмов программ с использованием технологий сериализации объекта в поток вывода данных и десериализации из потока ввода данных в объект на языке Java.
9	Java. Stream API. В результате выполнения лабораторных работ студент получает навыки разработки алгоритмов программ с использованием технологий Stream API при работе с коллекциями на языке Java.
10	Java. Spring. В результате выполнения лабораторных работ студент получит навыки разработки веб-приложений с помощью библиотеки Spring Boot на языке Java.

#### 4.3. Самостоятельная работа обучающихся.

№ п/п	Вид самостоятельной работы
1	Изучение рекомендованной литературы.
2	Подготовка к лабораторным и практическим занятиям.
3	Подготовка к промежуточной аттестации.
4	Подготовка к текущему контролю.

5. Перечень изданий, которые рекомендуется использовать при освоении дисциплины (модуля).

№ п/п	Библиографическое описание	Место доступа
----------	----------------------------	---------------



1	Гуськова, О. И. Объектно ориентированное программирование в Java : учебное пособие / О. И. Гуськова. — Москва : МПГУ, 2018. — 240 с. — ISBN 978-5-4263-0648-6.	<a href="https://e.lanbook.com/book/122311">https://e.lanbook.com/book/122311</a> (дата обращения: 04.02.2022).
2	Коузен, К. Современный Java: рецепты программирования / К. Коузен. — Москва : ДМК Пресс, 2018. — 275 с. — ISBN 978-5-97060-134-1.	<a href="https://e.lanbook.com/book/116121">https://e.lanbook.com/book/116121</a> (дата обращения: 04.02.2022)
3	Барков, И. А. Объектно-ориентированное программирование : учебник / И. А. Барков. — Санкт-Петербург : Лань, 2019. — 700 с. — ISBN 978-5-8114-3586-9	<a href="https://e.lanbook.com/book/119661">https://e.lanbook.com/book/119661</a> (дата обращения: 04.02.2022)
4	Долгинцев, А. П. Объектно-ориентированное программирование : учебное пособие / А. П. Долгинцев. — Самара : СамГУПС, 2011. — 31 с.	<a href="https://e.lanbook.com/book/130277">https://e.lanbook.com/book/130277</a> (дата обращения: 04.02.2022)
5	Васильев, А. Н. Самоучитель Java с примерами и программами : учебное пособие / А. Н. Васильев. — 4-е, изд. — Санкт-Петербург : Наука и Техника, 2017. — 368 с. — ISBN 978-5-94387-745-2.	<a href="https://e.lanbook.com/book/101548">https://e.lanbook.com/book/101548</a> (дата обращения: 04.02.2022)

6. Перечень современных профессиональных баз данных и информационных справочных систем, которые могут использоваться при освоении дисциплины (модуля).

Электронно-библиотечная система издательства «Лань» (<http://e.lanbook.com/>).

Научно-техническая библиотека РУТ (МИИТ) (<http://library.miit.ru>).

Образовательная платформа «юрайт» (<https://urait.ru/>).

Учебные курсы microsoft (<https://www.microsoft.com/ru-ru/learning/training.aspx>)

Справочник по языку Java (<https://metanit.com/java/tutorial/>)

7. Перечень лицензионного и свободно распространяемого программного обеспечения, в том числе отечественного производства, необходимого для освоения дисциплины (модуля).

Java 17

Пакет офисных приложений

Браузер с доступом в интернет

JetBrains IntelliJ IDEA

8. Описание материально-технической базы, необходимой для осуществления образовательного процесса по дисциплине (модулю).

Учебные аудитории для проведения учебных занятий, оснащенные компьютерной техникой и наборами демонстрационного оборудования.

Для практических занятий – наличие персональных компьютеров вычислительного класса.

9. Форма промежуточной аттестации:

Зачет в 1 семестре.

Экзамен во 2 семестре.

10. Оценочные материалы.

Оценочные материалы, применяемые при проведении промежуточной аттестации, разрабатываются в соответствии с локальным нормативным актом РУТ (МИИТ).

Авторы:

старший преподаватель кафедры  
«Цифровые технологии управления  
транспортными процессами»

И.С. Разживайкин

Согласовано:

Заведующий кафедрой ЦТУТП

В.Е. Нутович

Председатель учебно-методической  
комиссии

Н.А. Андриянова