

**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  
09.03.02 Информационные системы и технологии,  
утвержденной первым проректором РУТ (МИИТ)  
Тимониным В.С.

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

### **Программирование**

Направление подготовки: 09.03.02 Информационные системы и технологии

Направленность (профиль): Информационные системы и технологии на транспорте

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: 11.04.2024

## 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, принят язык C, реализованный в интегрированной среде Microsoft Visual Studio (включая 2010 и выше), для разработки используется интегрированная среда JetBrains IntelliJ IDEA, изучение и использование которой рассматривается в данном курсе.

В завершении курса студенты изучают современные подходы к написанию современных программных продуктов с использованием объектно-ориентированного подхода. Изучается разработка на объектно-ориентированном языке программирования Java.

Задачами дисциплины (модуля) являются:

- овладение структурной и объектно-ориентированной парадигмами программирования;
- формирование навыков использования формализации, декомпозиции, алгоритмизации, кодирования, тестирования и отладки при разработке и создании программного обеспечения.

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

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

**ОПК-2** - Способен понимать принципы работы современных информационных технологий и программных средств, в том числе отечественного производства, и использовать их при решении задач профессиональной деятельности;

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

**ОПК-8** - Способен применять математические модели, методы и средства проектирования информационных и автоматизированных систем.

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):	144	64	80
В том числе:			
Занятия лекционного типа	64	32	32
Занятия семинарского типа	80	32	48

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 180 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;

№ п/п	Тематика лекционных занятий / краткое содержание
	<ul style="list-style-type: none"> <li>- основные понятия алгоритмизации;</li> <li>- формы записи алгоритмов;</li> <li>- основные алгоритмические конструкции;</li> <li>- блок-схемы;</li> <li>- псевдокод;</li> <li>- виды алгоритмов;</li> <li>- стратегии;</li> <li>- сортировка;</li> <li>- поиск;</li> <li>- графы.</li> </ul>
2	<p><b>Языки программирования.</b></p> <p>Рассматриваемые вопросы:</p> <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>- 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> </ul>

№ п/п	Тематика лекционных занятий / краткое содержание
	<ul style="list-style-type: none"> <li>- наследование;</li> <li>- полиморфизм;</li> <li>- инкапсуляция.</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> <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	<p><b>Возможности Java.</b></p> <p>Рассматриваемые вопросы:</p> <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>
8	<p><b>Коллекции Java.</b></p> <p>Рассматриваемые вопросы:</p> <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>
9	<p><b>Многопоточность в Java.</b></p> <p>Рассматриваемые вопросы:</p> <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>
10	<p><b>Функциональное программирование в Java.</b></p> <p>Рассматриваемые вопросы:</p>

№ п/п	Тематика лекционных занятий / краткое содержание
	<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>
11	<p><b>Современная Java.</b></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>
12	<p><b>Введение в объектно-ориентированное программирование.</b></p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> <li>- объектно-ориентированная парадигма программирования;</li> <li>- объектно-ориентированная формализация и декомпозиция;</li> <li>- определение класса объектов, как понятия предметной области.</li> </ul>
13	<p><b>Основные особенности ООП.</b></p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> <li>- абстрагирование, существенные свойства объектов;</li> <li>- свойства класса объектов;</li> <li>- отношения между классами объектов: ассоциация, наследование, агрегация, зависимость;</li> <li>- описание состояния и поведение в классе объектов;</li> <li>- поля и методы класса объектов;</li> <li>- состояние и индивидуальность объекта.</li> </ul>
14	<p><b>Объектно-ориентированные языки программирования.</b></p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> <li>- описание классов объектов на языке C++;</li> <li>- описание классов объектов на языке Java.</li> </ul>
15	<p><b>Абстракция.</b></p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> <li>- понятие - как абстракция, используемая для описания классов объектов;</li> <li>- содержание и объём понятия;</li> <li>- выделение существенных свойств класса объектов.</li> </ul>
16	<p><b>Наследование.</b></p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> <li>- определение наследования классов объектов;</li> <li>- иерархия классов объектов на основе наследования;</li> <li>- распределение сложности по иерархии классов объектов;</li> <li>- исключение дублирования в иерархии классов объектов;</li> <li>- интерфейс, абстрактный класс объектов, конкретный класс объектов.</li> </ul>
17	<p><b>Полиморфизм.</b></p> <p>Рассматриваемые вопросы:</p> <ul style="list-style-type: none"> <li>- определение полиморфизма классов объектов;</li> <li>- полиморфные методы;</li> <li>- переопределение и перекрытие методов;</li> <li>- виртуальные методы.</li> </ul>

№ п/п	Тематика лекционных занятий / краткое содержание
18	<b>Инкапсуляция.</b> Рассматриваемые вопросы: <ul style="list-style-type: none"> <li>- определение инкапсуляции классов объектов;</li> <li>- время жизни и область видимости;</li> <li>- локальность и глобальность;</li> <li>- модульность, разбиение модуля на части, открытая и закрытая части модуля;</li> <li>- интерфейс и реализация класса объектов;</li> <li>- свобода в изменении реализации;</li> <li>- обеспечение разработки несколькими людьми;</li> <li>- сокрытие ноу-хао.</li> </ul>
19	<b>Модификаторы доступа.</b> Рассматриваемые вопросы: <ul style="list-style-type: none"> <li>- разбиение класса объектов на области доступа;</li> <li>- приватная область;</li> <li>- защищённая область;</li> <li>- общедоступная область.</li> </ul>
20	<b>Специальные методы в классе объектов.</b> Рассматриваемые вопросы: <ul style="list-style-type: none"> <li>- конструкторы и деструкторы;</li> <li>- создание конструкторов по умолчанию;</li> <li>- виртуализация деструкторов;</li> <li>- последовательность вызовов конструкторов и деструкторов;</li> <li>- гетеры и сетеры;</li> <li>- дружественные функции.</li> </ul>
21	<b>Перегрузка операций.</b> Рассматриваемые вопросы: <ul style="list-style-type: none"> <li>- копирование и присваивание;</li> <li>- перегрузка и приоритетность операций;</li> <li>- перегрузка внешним образом от перегрузки как метода класса.</li> </ul>
22	<b>Исключения.</b> Рассматриваемые вопросы: <ul style="list-style-type: none"> <li>- генерация исключений;</li> <li>- типы исключений;</li> <li>- перехват и обработка исключений.</li> </ul>
23	<b>Шаблоны.</b> Рассматриваемые вопросы: <ul style="list-style-type: none"> <li>- способы сортировки и поиска;</li> <li>- коллекции: очереди, стэки, дэки, списки, множества, отображения;</li> <li>- параметризация типа данных;</li> <li>- обобщенный алгоритм, шаблоны функций;</li> <li>- обобщённый класс, шаблоны классов;</li> <li>- реализация сортировки и поиска шаблонными функциями;</li> <li>- реализация коллекций объектов шаблонными классами.</li> </ul>
24	<b>Паттерны программирования.</b> Рассматриваемые вопросы: <ul style="list-style-type: none"> <li>- понятие паттерна программирования;</li> <li>- структура и принцип действия паттерна Singleton (Одиночка);</li> <li>- структура и принцип действия паттерна Abstract Factory (Абстрактная фабрика);</li> <li>- структура и принцип действия паттерна Observer (Наблюдатель);</li> <li>- структура и принцип действия паттерна Composite (Компоновщик);</li> </ul>



№ п/п	Тематика лекционных занятий / краткое содержание
	- структура и принцип действия паттерна Decorator (Декоратор); - структура и принцип действия паттерна Strategy (Стратегия).

## 4.2. Занятия семинарского типа.

### Лабораторные работы

№ п/п	Наименование лабораторных работ / краткое содержание
1	Java. Основы языка. В результате выполнения лабораторных работ студент получает навыки работы с базовыми конструкциями, типами данных и синтаксическими особенностями языка Java.
2	Java. Логические конструкции. В результате выполнения лабораторных работ студент получает навыки работы с логическими конструкциями и вложенными логическими конструкциями на языке Java.
3	Java. Циклы. В результате выполнения лабораторных работ студент получает навыки работы с операторами циклов (while, for, foreacj) и их отличительными особенностями на языке Java.
4	Java. Вложенные циклы. В результате выполнения лабораторных работ студент получает навыки работы с построением алгоритмов с использованием вложенных циклов и использование бесконечных циклов на языке Java.
5	Java. Массивы. В результате выполнения лабораторных работ студент получает навыки работы с массивами на языке Java.
6	Java. Строки. В результате выполнения лабораторных работ студент получает навыки работы со строками и регулярными выражениями на языке Java.
7	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.
8	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).
9	Java. Обработка исключений. В результате выполнения лабораторных работ студент получает навыки разработки алгоритмов программ с использованием технологий обработки исключений на языке Java.
10	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.
11	Java. Сериализация и десериализация. В результате выполнения лабораторных работ студент получает навыки разработки алгоритмов программ с использованием технологий сереализации объекта в поток ввывода данных и десериализации из потока ввода данных в объект на языке Java.
12	Структура и класс объектов. В результате выполнения лабораторной работы студент получает навык создания класса объектов,

№ п/п	Наименование лабораторных работ / краткое содержание
	описание полей и методов класса объектов и знание об отличии структуры от класса объектов и отличии функции от метода класса.
13	<b>Конструкторы и деструкторы.</b> В результате выполнения лабораторной работы студент получает навык описания конструкторов и деструкторов в классе объектов, приобретает понятие о перегрузки методов, о типах конструкторов и виртуализации деструкторов.
14	<b>Перегрузка операций.</b> В результате выполнения лабораторной работы студент получает навык перегрузки операций внешним образом и перегрузки как метода класса, определения дружественных функций.
15	<b>Наследование.</b> В результате выполнения лабораторной работы студент получает навык наследование одного класса от другого.
16	<b>Полиморфизм.</b> В результате выполнения лабораторной работы студент получает навык переопределение методов в наследуемых классах.
17	<b>Исключения.</b> В результате выполнения лабораторной работы студент получает навык генерации исключений, перехвата и обработки исключений.

### Практические занятия

№ п/п	Тематика практических занятий/краткое содержание
1	<b>Java. Коллекции.</b> В результате выполнения практических работ студент получает навыки работы с коллекциями, реализующими интерфейс List на языке Java. В результате выполнения практических работ студент получает навыки работы с коллекциями, реализующими интерфейс Set на языке Java. В результате выполнения практических работ студент получает навыки работы с коллекциями, реализующими интерфейс Map на языке Java. В результате выполнения практических работ студент получает навыки работы с коллекциями, реализующими интерфейс Queue на языке Java.
2	<b>Java. Функциональное программирование.</b> В результате выполнения практических работ студент получает навыки разработки алгоритмов программ с применением функциональных возможностей (лямбда-выражений) языка Java.
3	<b>Java. Многопоточное программирование.</b> В результате выполнения практических работ студент получает навыки разработки алгоритмов программ с применением технологий многопоточного программирования и коллекциями из пакета java.util.concurrent на языке Java.
4	<b>Разработка класса объектов.</b> В результате работы на практическом занятии студент получает навыки описания класса объектов посредством содержания или объёма понятия предметной области, выделения существенных свойств для описания класса объектов, описание состояния и поведения классов объектов.
5	<b>Разработка иерархии классов объектов.</b> В результате работы на практическом занятии студент получает навыки обобщения и специализации классов объектов, выделения интерфейса и определения абстрактных классов, переопределения методов классов.
6	<b>Использование агрегации для создания классов.</b> В результате работы на практическом занятии студент получает навыки сборки классов из нескольких других классов, понятие об альтернативе наследованию в порождении новых классов объектов.

### 4.3. Самостоятельная работа обучающихся.

№ п/п	Вид самостоятельной работы
1	Изучение рекомендованной литературы.
2	Подготовка к лабораторным и практическим занятиям.
3	Рассмотрение отличий объектно-ориентированного программирование от структурного программирования.
4	Изучения создания классов объектов на разных языках программирования.
5	Изучение использования наследования, инкапсуляции и полиморфизма в разных языках программирования.
6	Подготовка к промежуточной аттестации.
7	Подготовка к текущему контролю.

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

№ п/п	Библиографическое описание	Место доступа
1	JAVA тм 2 П. Ноутон, Г. Шилдт; Пер. с англ. Однотомное издание БХВ-Петербург , 2000, ISBN: 5-94157-012-0, 1072 с.	НТБ (ЭЭ); НТБ (фб.)
2	Объектно ориентированное программирование в Java : учебное пособие Гуськова, О. И. Учебное пособие 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).
3	Современный Java: рецепты программирования К. Коузен Москва : ДМК Пресс , 2018, 274 с.	<a href="https://e.lanbook.com/book/116121">https://e.lanbook.com/book/116121</a> (дата обращения: 04.02.2022)
4	Объектно-ориентированное программирование : учебник И. А. Барков. Учебник Санкт-Петербург : Лань , 2019, ISBN 978-5-8114-3586-9, 698 с.	<a href="https://e.lanbook.com/book/119661">https://e.lanbook.com/book/119661</a> (дата обращения: 04.02.2022)
5	Объектно-ориентированное программирование : учебное пособие А. П. Долгинцев Учебное пособие 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)
6	Самоучитель Java с примерами и программами : учебное пособие А. Н. Васильев Учебное пособие Санкт-Петербург : Наука и Техника , 2017, ISBN 978-5-94387-745-2, 365 с.	<a href="https://e.lanbook.com/book/101548">https://e.lanbook.com/book/101548</a> (дата обращения: 04.02.2022)
7	Программирование на языке высокого уровня C/C++ Зоткин С.П. Книга Москва: МИСИ - МГСУ , 2018, ISBN 978-5-7264-1810-0, 140 с.	<a href="https://e.lanbook.com/book/108512">https://e.lanbook.com/book/108512</a>
8	Программирование на языке C++ Борисов С.В., Пащенко О.Б., Серебрякова И.Л., Степанов В.П.	<a href="https://e.lanbook.com/book/103498">https://e.lanbook.com/book/103498</a>

	Книга Москва: МГТУ им. Н.Э. Баумана , 2017, ISBN: 978-5-7038-4726-8, 74 с.	
9	Решение задач на языках программирования Си и Си++ Быков А.Ю. Книга Москва: МГТУ им Н.Э. Баумана , 2017, ISBN 978-5-7038-4577-6, 244 с.	<a href="https://e.lanbook.com/book/103505">https://e.lanbook.com/book/103505</a>
10	Курс программирования на языке Си Подбельский В.В., Фомин С.С. Книга Москва: ДМК Пресс , 2012, ISBN 978-5-94074-449-8, 384 с.	<a href="https://e.lanbook.com/book/4148">https://e.lanbook.com/book/4148</a>
11	Программирование на языке С++ Павловская Т.А, Книга Москва , 2016, 154 с.	<a href="https://e.lanbook.com/book/100409">https://e.lanbook.com/book/100409</a>
12	Программирование на языке С в Microsoft Visual Studio 2010 Александров Э.Э., Афонин В.В. Книга Москва , 2010, ISBN 978-5-7103-2341-0 , 570 с.	<a href="https://e.lanbook.com/book/100410">https://e.lanbook.com/book/100410</a>

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/>).

Электронно-библиотечная система [ibooks.ru](http://ibooks.ru) (<http://ibooks.ru/>).

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/>).

C/C++ tutorial (<http://www.cplusplus.com/doc/tutorial>).

C/C++ practice (<https://www.geeksforgeeks.org/c-plus-plus>).

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

Microsoft Visual Studio.

JetBrains IntelliJ IDEA.

Java 17.

Microsoft Internet Explorer (или другой браузер).

Операционная система Microsoft Windows.

Microsoft Office.

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»

Е.А. Заманов

старший преподаватель кафедры  
«Цифровые технологии управления  
транспортными процессами»

И.С. Разживайкин

А.Ю. Павлов

Согласовано:

Заведующий кафедрой ЦТУТП

В.Е. Нутович

Председатель учебно-методической  
комиссии

Н.А. Андриянова