МИНИСТЕРСТВО ТРАНСПОРТА РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«РОССИЙСКИЙ УНИВЕРСИТЕТ ТРАНСПОРТА»
(РУТ (МИИТ)

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

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

Алгоритмы и структура данных

Направление подготовки: 09.03.01 Информатика и вычислительная техника

Направленность (профиль): Цифровая инженерия транспортных процессов

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 ФГАОУ ВО РУТ (МИИТ)
ID подписи: 937226
Подписал: руководитель образовательной программы
Проневич Ольга Борисовна
Дата: 09.09.2026

1. Общие сведения о дисциплине (модуле).

Целями освоения учебной дисциплины являются формирование у обучающихся системных знаний и практических навыков в области алгоритмов и структур данных для разработки и отладки программного кода при решении прикладных задач в транспортной отрасли. В рамках курса рассматриваются основные алгоритмы решения типичных задач, структуры данных (массивы, списки, стеки, очереди, деревья, графы, хеш-таблицы), методы оценки эффективности алгоритмов (временная и пространственная сложность), алгоритмы сортировки и поиска, графовые алгоритмы, а также нотации и программное обеспечение для графического отображения алгоритмов. Особое внимание уделяется практическим аспектам реализации алгоритмов на языке Python, адаптации методов под специфику задач транспортной отрасли, обоснованию выбора алгоритмов для программной реализации моделей и оценке их эффективности.

Задачами освоения дисциплины являются:

Формирование теоретических знаний об основных алгоритмах решения типичных задач, областях и способах их применения.

Освоение структур данных, их свойств, областей применения и влияния на производительность алгоритмов.

Формирование практических навыков реализации алгоритмов на языке Python с использованием стандартных библиотек.

Освоение методов оценки эффективности алгоритмов (Big O нотация) и выбора оптимальных решений для конкретных задач.

Приобретение навыков описания задач предметной области в виде формальных математических моделей, пригодных для последующей алгоритмизации и программной реализации.

Развитие навыков отладки программного кода, выявления и исправления ошибок.

Формирование навыков работы с системами управления версиями (Git) и оформления программного кода в соответствии со стандартами.

Дисциплина предназначена для формирования компетенций, необходимых для разработки, отладки и сопровождения алгоритмических решений в транспортных организациях, и предназначена для студентов всех ролевых траекторий: «Инженер данных», «Аналитик данных», «MLOps / Data Scientist».

2. Планируемые результаты обучения по дисциплине (модулю).

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

ППК-Р1 - Способен разрабатывать и отлаживать программный код.

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

Знать:

Алгоритмы решения типичных задач, области и способы их применения; нотации и программное обеспечение для графического отображения алгоритмов.

Синтаксис выбранного языка программирования, особенности программирования на этом языке, стандартные библиотеки языка программирования; методологии разработки программного обеспечения.

Уметь:

Выбирать адекватный математический аппарат для формализованного описания сущностей и отношений на основе ТЗ и бизнес-требований; проводить сравнительный анализ и обоснование выбора алгоритмов для программной реализации модели.

Применять нормативно-технические документы к оформлению программного кода; регистрировать изменения в системе управления версиями; выявлять и отлаживать ошибки в программном коде.

Владеть:

Навыками описания задач предметной области в виде формальных математических моделей, пригодных для последующей алгоритмизации и программной реализации.

Методами адаптации методов и алгоритмов под специфику задачи; навыками оценки эффективности выбранных алгоритмов.

3. Объем дисциплины (модуля).

3.1. Общая трудоемкость дисциплины (модуля).

Общая трудоемкость дисциплины (модуля) составляет 2 з.е. (72 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 №7
Контактная работа при проведении учебных занятий (всего):	32	32
В том числе:		
Занятия лекционного типа	16	16
Занятия семинарского типа	16	16

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 40 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	Тема 1. Введение в алгоритмы и структуры данных. Рассматриваемые вопросы: - Понятие алгоритма. Свойства алгоритмов. - Классификация алгоритмов по областям применения. - Нотации для графического отображения алгоритмов: блок-схемы, псевдокод, UML-диаграммы. - Программное обеспечение для моделирования алгоритмов. - Понятие структуры данных. Связь алгоритмов и структур данных. - Примеры алгоритмов в транспортной отрасли (оптимизация маршрутов, управление потоками данных).
2	Тема 2. Оценка эффективности алгоритмов. Рассматриваемые вопросы: - Временная и пространственная сложность алгоритмов. - Асимптотический анализ: O-нотация (Big O), ?-нотация, ?-нотация. - Анализ сложности базовых операций. - Сравнительный анализ алгоритмов по эффективности. - Факторы, влияющие на производительность алгоритмов. - Выбор алгоритма на основе оценки эффективности.
3	Тема 3. Базовые структуры данных. Рассматриваемые вопросы: - Массивы: статические и динамические. Операции и сложность.

№ п/п	Тематика лекционных занятий / краткое содержание
	- Связные списки: односвязные, двусвязные, кольцевые. - Стек: принцип LIFO, реализация, применение. - Очередь: принцип FIFO, реализация, применение. - Дек: двусторонняя очередь. - Сравнение структур данных и выбор для конкретных задач транспортной отрасли.
4	Тема 4. Алгоритмы сортировки и поиска. Рассматриваемые вопросы: - Алгоритмы сортировки: пузырьковая, вставками, выбором, слиянием, быстрая. - Сравнительный анализ алгоритмов сортировки по сложности и стабильности. - Алгоритмы поиска: линейный поиск, бинарный поиск. - Интерполяционный поиск. - Применение алгоритмов сортировки и поиска в транспортных задачах. - Выбор оптимального алгоритма для конкретной задачи.
5	Тема 5. Хеш-таблицы и словари. Рассматриваемые вопросы: - Понятие хеш-функции. Требования к хеш-функциям. - Разрешение коллизий: метод цепочек, открытая адресация. - Реализация хеш-таблиц в Python (dict, set). - Операции и их сложность. - Применение хеш-таблиц для оптимизации поиска и обработки данных. - Использование хеш-таблиц в транспортных системах (кэширование, индексация).
6	Тема 6. Деревья и графы. Рассматриваемые вопросы: - Двоичные деревья: основные понятия, обходы (в глубину, в ширину). - Двоичные деревья поиска: свойства, операции, сложность. - Сбалансированные деревья: AVL-деревья, красно-черные деревья. - Куча: двоичная куча, пирамидальная сортировка. - Очередь с приоритетом. - Применение деревьев в транспортных задачах (иерархические структуры, маршрутизация).
7	Тема 7. Алгоритмы на графах. Рассматриваемые вопросы: - Основные понятия теории графов. Способы представления графов. - Алгоритмы обхода графов: BFS, DFS. - Алгоритм Дейкстры для поиска кратчайшего пути. - Алгоритм A* и его применение в навигации. - Алгоритм Флойда-Уоршелла. - Применение графовых алгоритмов в транспортной отрасли (построение маршрутов, логистика, оптимизация сетей).
8	Тема 8. Современные алгоритмы и структуры данных для больших данных. Рассматриваемые вопросы: - Внешняя сортировка и обработка данных, не помещающихся в память. - Структуры данных для потоковой обработки (Bloom filter, HyperLogLog). - Алгоритмы сжатия данных. - Алгоритмы для обработки графов больших размеров. - Алгоритмы машинного обучения и их связь с классическими алгоритмами. - Перспективные направления развития алгоритмов и структур данных.

4.2. Занятия семинарского типа.

Практические занятия

№ п/п	Тематика практических занятий/краткое содержание
1	Тема 1. Установка и настройка среды разработки. Введение в Python. Рассматриваемые вопросы: - Установка Python и менеджера пакетов pip. - Установка и настройка IDE (PyCharm / VS Code). - Настройка виртуального окружения (venv). - Основы синтаксиса Python: переменные, типы данных, условные операторы, циклы. - Создание первого проекта и написание простейших алгоритмов. - Основы работы с отладчиком (Debugger).
2	Тема 2. Реализация базовых структур данных. Рассматриваемые вопросы: - Реализация стека и очереди на Python. - Реализация односвязного и двусвязного списка. - Сравнение эффективности различных структур данных. - Применение структур данных для решения транспортной задачи (моделирование очереди транспортных средств). - Оценка эффективности реализованных решений.
3	Тема 3. Реализация алгоритмов сортировки. Рассматриваемые вопросы: - Реализация алгоритмов сортировки: пузырьковая, вставками, слиянием, быстрая. - Сравнение производительности алгоритмов на различных наборах данных. - Измерение времени выполнения и анализ сложности. - Применение сортировки для обработки транспортных данных (сортировка по времени, расстоянию и т.д.). - Обоснование выбора алгоритма для конкретной задачи.
4	Тема 4. Реализация алгоритмов поиска. Рассматриваемые вопросы: - Реализация линейного и бинарного поиска. - Реализация интерполяционного поиска. - Сравнительный анализ производительности алгоритмов поиска. - Применение бинарного поиска в отсортированных транспортных данных. - Оценка эффективности и обоснование выбора алгоритма.
5	Тема 5. Работа с хеш-таблицами. Рассматриваемые вопросы: - Реализация хеш-таблицы с разрешением коллизий методом цепочек. - Использование встроенного типа dict в Python. - Решение задачи поиска и обработки данных с использованием хеш-таблиц. - Применение хеш-таблиц в транспортной отрасли (кэширование маршрутов, хранение данных о поездках). - Сравнение производительности хеш-таблиц с другими структурами данных.
6	Тема 6. Реализация деревьев и алгоритмов обхода. Рассматриваемые вопросы: - Реализация двоичного дерева поиска и его операций (вставка, удаление, поиск). - Реализация обходов дерева: прямой, симметричный, обратный. - Реализация обхода в глубину (DFS) и в ширину (BFS). - Применение деревьев для иерархических данных в транспортных системах. - Оценка эффективности операций с деревьями.
7	Тема 7. Реализация графовых алгоритмов. Рассматриваемые вопросы: - Представление графа в Python (матрица смежности, список смежности). - Реализация обхода графа: BFS, DFS.

№ п/п	Тематика практических занятий/краткое содержание
	- Реализация алгоритма Дейкстры. - Реализация алгоритма A*. - Применение графовых алгоритмов для поиска кратчайшего пути в транспортной сети. - Сравнительный анализ алгоритмов и выбор оптимального решения.
8	Тема 8. Комплексный проект. Рассматриваемые вопросы: - Разработка алгоритмического решения для транспортной задачи с использованием изученных алгоритмов и структур данных. - Реализация кода с соблюдением стандартов оформления. - Отладка и тестирование разработанного модуля. - Оценка эффективности и обоснование выбора алгоритмов. - Оформление технической документации. - Защита проекта перед преподавателем.

4.3. Самостоятельная работа обучающихся.

№ п/п	Вид самостоятельной работы
1	Изучение теоретического материала по алгоритмам и структурам данных.
2	Подготовка к практическим занятиям и выполнение лабораторных работ.
3	Подготовка презентации по результатам проекта для защиты.
4	Подготовка к промежуточной аттестации.
5	Подготовка к промежуточной аттестации.

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

№ п/п	Библиографическое описание	Место доступа
1	Скворцова, Л. А. Структуры и алгоритмы обработки данных : учебное пособие / Л. А. Скворцова, К. В. Гусев, А. С. Филатов. — Москва : РТУ МИРЭА, 2023 — Часть 1 : Линейные структуры данных в алгоритмах — 2023. — 255 с. — ISBN 978-5-7339-1971-3.	https://e.lanbook.com/book/386186
2	Куликов, А. А. Алгоритмы и структуры данных с использованием компилируемых языков : учебное пособие / А. А. Куликов. — Москва : РТУ МИРЭА, 2026. — 220 с. — ISBN 978-5-7339-2906-4.	https://e.lanbook.com/book/523272

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

MS Office (Excel, Word)

Notepad++

Браузер Chrome

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

Python 3.11/3.12 и JupyterLab / Jupyter Notebook

pandas, NumPy, SciPy, statsmodels и Pingouin

Matplotlib, Seaborn и Plotly

PostgreSQL и DBeaver Community Edition

Microsoft Excel и Power BI Desktop / LibreOffice Calc

Yandex DataLens, Metabase или Apache Superset

Git и GitHub / GitLab

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

Компьютер преподавателя с доступом к сети Интернет

Компьютеры студентов с установленным Python-окружением и табличным процессором

Доступ к учебной базе событийных данных и экспериментальным выборкам

Доступ к PostgreSQL и BI-системе для визуализации результатов

Доступ к облачной виртуальной машине или Docker-контейнерам для воспроизводимого анализа

Проектор и экран для проектора

Маркерная доска

9. Форма промежуточной аттестации:

Зачет в 7 семестре.

10. Оценочные материалы.

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

Авторы:

руководитель образовательной программы О.Б. Проневич

Согласовано:

Директор Д.В. Паринов

Руководитель образовательной программы О.Б. Проневич

Председатель учебно-методической комиссии Д.В. Паринов