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 директором ФГАОУ ВО РУТ
(МИИТ) Париновым Д.В.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ (МОДУЛЯ)

Теория графов и комбинаторика

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 ФГАОУ ВО РУТ (МИИТ)
ID подписи: 937226
Подписал: директор центра Проневич Ольга Борисовна
Дата: 12.11.2026

1. Общие сведения о дисциплине (модуле).

Дисциплина формирует способность представлять прикладные задачи транспортной отрасли графовыми и комбинаторными моделями, обоснованно выбирать и адаптировать алгоритмы, оценивать их корректность, сложность и эффективность на воспроизводимом эксперименте. Востребованность этих умений подтверждается цифровизацией транспортной логистики – внедрением систем автоматической маршрутизации, которым нужны специалисты, умеющие переводить предметную задачу в формальную модель и доказательно выбирать алгоритм. В сквозном кейсе обучающийся строит графовую модель транспортной сети, сравнивает алгоритмы поиска пути, реализует два варианта на Java, проводит сравнительное испытание и проектирует перестроение маршрутов при закрытии участка. Студент узнает классы графов, способы их представления, алгоритмы обхода и поиска кратчайших путей, асимптотический анализ и комбинаторные модели, научится формализовывать задачу, обосновывать выбор и оформлять инженерное решение.

Целью освоения дисциплины является формирование способности представлять прикладные задачи графовыми и комбинаторными моделями, обоснованно выбирать и адаптировать алгоритмы, оценивать их корректность, вычислительную сложность и эффективность на воспроизводимом эксперименте.

Для достижения поставленной цели в рамках дисциплины решается комплекс задач, направленных на формирование у обучающихся способности: формализовать прикладную задачу в виде графа, определяя вершины, рёбра, направления, веса и ограничения. Сравнивать классы графовых и комбинаторных методов по корректности, вычислительной сложности, памяти и характеру входных данных. Обосновывать выбор алгоритма для конкретного режима запросов и изменений модели. Адаптировать алгоритм к ограничениям предметной задачи без нарушения его предпосылок. Реализовывать выбранные варианты на языке Java и экспериментально оценивать их эффективность. Формулировать границы применимости решения и безопасный вариант поведения при нарушении предпосылок.

2. Планируемые результаты обучения по дисциплине (модулю).

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

ОПК-1 - Способен применять методы математического анализа и моделирования, теоретического и экспериментального исследования в профессиональной деятельности;

ППК-Р1 - Способен разрабатывать и отлаживать программный код.

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

Знать:

- разделы математики, применяемые для анализа и моделирования в инженерных задачах;
- методы теоретического и экспериментального исследования процессов и систем;
- инструменты математического моделирования и вычислительного эксперимента;
- основные классы методов программной реализации моделей и критерии выбора алгоритмов.

Уметь:

- строить математические модели процессов и систем по формализованной постановке задачи;
- применять математический аппарат для анализа и решения профессиональных задач;
- проводить экспериментальную проверку и оценку адекватности моделей;
- проводить сравнительный анализ и обоснование выбора алгоритмов для программной реализации модели.

Владеть:

- навыками доказательной проверки инженерных решений на основе математических моделей;
- 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 3 з.е. (108 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 №4
Контактная работа при проведении учебных занятий (всего):	64	64
В том числе:		
Занятия лекционного типа	32	32
Занятия семинарского типа	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 44 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	Основные определения теории графов и их классификация Рассматриваемые вопросы: - понятие графа как упорядоченной пары множеств вершин и рёбер; - ориентированные и неориентированные графы, пути, циклы, связность; - взвешенные графы и интерпретация веса в транспортных задачах; - базовые классы графов по структурным признакам – деревья, двудольные, планарные, полные; - смежность, инцидентность, степень вершины и их роль в моделировании транспортных узлов.
2	Способы представления графов в памяти компьютера Рассматриваемые вопросы: - матрица смежности – структура, сложность операций, область применимости; - списки смежности – структура, сложность операций, преимущества для разреженных графов; - матрица инцидентности и её использование в специализированных задачах; - критерии выбора представления под профиль операций и плотность графа; - оценка затрат памяти для различных представлений в контексте транспортных сетей.

№ п/п	Тематика лекционных занятий / краткое содержание
3	Формализация предметных задач транспортной логистики в графовые модели Рассматриваемые вопросы: - этапы перевода неформального описания транспортной сети в формальную модель; - определение множества вершин как станций, узлов и терминалов; - определение множества рёбер как участков пути с направлением и весом; - формализация ограничений – тип состава, пропускная способность, временные окна; - словарь данных как инструмент обеспечения однозначной интерпретации модели.
4	Алгоритмы обхода графов Рассматриваемые вопросы: - поиск в ширину (BFS) – идея, инварианты, сложность и свойства; - поиск в глубину (DFS) – идея, рекурсивная и итеративная реализации, инварианты; - применение обходов для проверки связности и поиска компонент; - топологическая сортировка на основе DFS и её использование в планировании; - обнаружение циклов в ориентированных графах как необходимое условие корректности маршрута.
5	Алгоритм Дейкстры для поиска кратчайших путей Рассматриваемые вопросы: - математическая постановка задачи поиска кратчайшего пути; - идея жадного алгоритма Дейкстры и условие неотрицательности весов; - структуры данных для эффективной реализации – массив, двоичная куча; - доказательство корректности и оценка временной сложности; - ограничения применимости и поведение при нарушении предпосылки о неотрицательности.
6	Алгоритмы Флойда–Уоршелла и Беллмана–Форда Рассматриваемые вопросы: - алгоритм Флойда–Уоршелла – динамическое программирование для всех пар вершин; - временная и пространственная сложность алгоритма Флойда–Уоршелла; - алгоритм Беллмана–Форда – обнаружение отрицательных циклов; - сравнительный анализ трёх алгоритмов по критериям сложности, памяти и режима запросов; - выбор алгоритма для статического планирования и оперативного перестроения маршрута.
7	Алгоритмы построения остовных деревьев Рассматриваемые вопросы: - понятие остовного дерева и его роль в проектировании транспортных сетей; - алгоритм Прима – жадная стратегия наращивания дерева из стартовой вершины; - алгоритм Краскала – сортировка рёбер и объединение компонент связности; - сравнение алгоритмов Прима и Краскала по сложности и структурам данных; - применение остовных деревьев для минимизации стоимости инфраструктуры.
8	Асимптотический анализ вычислительной сложности Рассматриваемые вопросы: - нотации O, ? и ? как инструменты сравнительной оценки; - анализ временной сложности для последовательных и вложенных циклов; - анализ пространственной сложности при различных представлениях графа; - практическое значение асимптотических оценок для транспортных сетей большого масштаба; - ограничения асимптотического анализа и необходимость экспериментальной проверки.
9	Комбинаторные модели дискретных структур Рассматриваемые вопросы: - перестановки, сочетания и размещения как базовые комбинаторные объекты; - принцип включения и исключения для подсчёта маршрутов с ограничениями; - связь комбинаторных формул с полным перебором на графах; - оценка числа возможных маршрутов и её влияние на выбор алгоритма; - применение комбинаторного анализа для оценки сложности задач маршрутизации.

№ п/п	Тематика лекционных занятий / краткое содержание
10	Классификация графовых задач по режимам запросов Рассматриваемые вопросы: - статический поиск пути – однократный запрос к неизменному графу; - динамическое обновление – изменение весов и структуры рёбер во времени; - потоковые задачи – многократные запросы с учётом пропускной способности; - соответствие классов алгоритмов режимам запросов; - инженерные компромиссы при выборе решения под заданный режим эксплуатации.
11	Адаптация алгоритмов к предметным ограничениям и ООП-проектирование Рассматриваемые вопросы: - модификация структур данных для учёта типа состава и пропускной способности; - сохранение корректности при добавлении предметных ограничений; - проектирование иерархии классов графа, вершины и ребра на языке Java; - применение шаблонов проектирования для обеспечения единого внешнего контракта; - разделение ответственности между моделью, алгоритмом и проверкой.
12	Генерация тестовых наборов и методология воспроизводимого эксперимента Рассматриваемые вопросы: - классификация тестовых сценариев – нормальные, граничные и дегенеративные случаи; - методы генерации связных графов заданного размера и плотности; - постановка гипотезы о производительности и выбор метрик измерения; - измерение времени выполнения и потребления памяти в JVM; - структура и содержание протокола воспроизводимого эксперимента.
13	Факторы производительности JVM и расхождение теории с измерениями Рассматриваемые вопросы: - влияние JIT-компиляции и кэширования инструкций на время выполнения; - роль сборщика мусора и выделения памяти в измерениях производительности; - зависимость фактического времени от структуры входных данных и их размещения в памяти; - методы профилирования Java-приложений для локализации узких мест; - интерпретация расхождения между асимптотической оценкой и экспериментальными данными.
14	Инкрементальное обновление графовых моделей Рассматриваемые вопросы: - версионированные снимки графа как способ фиксации состояния сети; - локальный пересчёт затронутых маршрутов при закрытии участка; - оценка стоимости события изменения в терминах времени пересчёта; - сравнение полного пересчёта и инкрементального обновления; - обеспечение корректности нетронутых маршрутов при структурных изменениях.
15	Специальные классы графов и теория потоков в сетях Рассматриваемые вопросы: - свойства деревьев, двудольных графов и их применение в моделировании; - эйлеровы и гамильтоновы циклы в задачах обхода транспортной инфраструктуры; - постановка задачи о максимальном потоке и теорема Форда–Фалкерсона; - алгоритм Эдмонса–Карпа и его сложность; - применение теории потоков для анализа пропускной способности магистралей.
16	Графовые и комбинаторные методы в инженерной практике транспортной отрасли Рассматриваемые вопросы: - типовые классы инженерных задач транспорта и соответствующие им графовые модели – маршрутизация, расписания, размещение ресурсов, анализ надёжности сети; - жизненный цикл алгоритмического решения от формальной модели до эксплуатации в составе программной системы; - воспроизводимость и документирование как условия проверяемости инженерного результата; - организация коллективной разработки алгоритмических модулей и роль управления версиями в

№ п/п	Тематика лекционных занятий / краткое содержание
	совместной работе; - перспективные направления – динамические и вероятностные графовые модели, приближённые и эвристические алгоритмы для сетей большого масштаба.

4.2. Занятия семинарского типа.

Практические занятия

№ п/п	Тематика практических занятий/краткое содержание
1	Формализация предметных сущностей транспортной сети Студент получает обезличенное описание транспортной сети индустриального партнёра и выделяет из него станции, участки пути и ограничения движения. Он фиксирует каждую выделенную сущность в таблице соответствия предметного термина и математического объекта графа. Результатом работы является согласованный с преподавателем словарь сущностей, который послужит основой для построения формальной модели.
2	Построение ориентированного взвешенного графа На основе словаря сущностей студент строит ориентированный взвешенный граф учебной транспортной сети, определяя множество вершин и множество рёбер с направлением. Он выбирает способ представления графа в памяти и обосновывает этот выбор плотностью сети и профилем предполагаемых операций. Результатом является формальная графовая модель с явно заданными параметрами и схема её представления.
3	Определение весов и ограничений графовой модели Студент дополняет графовую модель весами и ограничениями, интерпретируя время в пути, тариф и пропускную способность как веса и атрибуты рёбер. Он проверяет непротиворечивость ограничений и корректность направлений на наличие тупиковых вершин без выхода для заданного типа состава. Результатом является полная спецификация графа с весовой функцией и множеством ограничений.
4	Верификация модели и подготовка к передаче Студент верифицирует построенную модель на учебном наборе маршрутов, сопоставляя предметный результат с формальным путём в графе. Он устраняет выявленные несоответствия между описанием сети и моделью и оформляет итоговый словарь данных. Результатом является зафиксированная формальная модель, передаваемая в качестве входных данных для следующей фазы.
5	Сравнительный анализ и выбор алгоритмов поиска пути Студент анализирует режим запросов к модели и формирует матрицу выбора алгоритма поиска пути по критериям временной сложности, пространственной сложности и характера входных данных. Он сравнивает не менее двух алгоритмов и аргументированно выбирает базовый и оптимизированный варианты для реализации. Результатом является обоснованная таблица решений с указанием границ применимости каждого алгоритма.
6	Проектирование структур данных графа на Java Студент проектирует иерархию классов для представления графа, вершины и ребра на языке Java, обеспечивая разделение ответственности между моделью и алгоритмом. Он выбирает структуры данных из стандартной библиотеки Java и фиксирует внешний контракт программного модуля. Результатом является проектная спецификация классов и интерфейсов, готовая к реализации.
7	Реализация базового варианта алгоритма поиска пути Студент реализует базовый вариант выбранного алгоритма поиска пути на языке Java в соответствии с зафиксированным контрактом. Он обеспечивает корректную обработку нормальных сценариев и явное возвращение результата при отсутствии пути между заданными вершинами. Результатом является работающий программный модуль с воспроизводимой сборкой.

№ п/п	Тематика практических занятий/краткое содержание
8	Реализация оптимизированного варианта алгоритма Студент реализует оптимизированный вариант алгоритма с изменённой структурой данных или эвристикой, сохраняя единый внешний контракт с базовым вариантом. Он проверяет эквивалентность результатов обеих реализаций на общем наборе тестовых данных. Результатом является вторая реализация и протокол проверки эквивалентности поведения.
9	Генерация воспроизводимых тестовых наборов данных Студент генерирует воспроизводимые наборы данных для графа не менее трёх масштабов, включая нормальные, граничные и дегенеративные случаи. Он документирует параметры генерации и фиксирует начальное состояние генератора псевдослучайных чисел. Результатом является версионированный архив тестовых данных с описанием каждого сценария.
10	Разработка автоматизированных проверок корректности Студент разрабатывает автоматизированные проверки корректности обеих реализаций алгоритма с использованием фреймворка JUnit 5. Он покрывает нормальные, граничные и ошибочные сценарии, включая отсутствие пути и несуществующую вершину. Результатом является набор независимых проверок, успешно проходящихся для обоих вариантов реализации.
11	Проведение сравнительного эксперимента Студент проводит сравнительное испытание базового и оптимизированного вариантов на единых наборах данных в изолированном контролируемом окружении. Он измеряет время выполнения и потребление памяти JVM для каждого масштаба и фиксирует результаты в протоколе. Результатом является воспроизводимый протокол измерений с таблицей сравнительных показателей.
12	Анализ результатов и оформление протокола эксперимента Студент анализирует полученные измерения, объясняет расхождение между асимптотической оценкой и фактическими результатами и формулирует инженерную рекомендацию по выбору алгоритма. Он оформляет протокол эксперимента и таблицу решений в соответствии с установленными требованиями. Результатом является завершённый раздел портфолио с обоснованным выбором для эксплуатации.
13	Моделирование закрытия участка и версионирование графа Студент моделирует сценарий закрытия участка транспортной сети путём удаления или блокировки ребра в графовой модели. Он проектирует версионированный снимок графа до изменения и определяет множество маршрутов, затронутых закрытием участка. Результатом является спецификация изменения и перечень маршрутов, подлежащих пересчёту.
14	Проектирование инкрементального пересчёта маршрутов Студент проектирует алгоритм локального пересчёта затронутых маршрутов без полного пересчёта всей сети. Он сравнивает стратегию инкрементального обновления с полным пересчётом и обосновывает выбор под ресурсные ограничения оперативного режима. Результатом является псевдокод и проектная спецификация инкрементального алгоритма.
15	Реализация и проверка инкрементального перестроения Студент реализует на языке Java базовый и оптимизированный варианты инкрементального пересчёта затронутых маршрутов при закрытии участка. Он проверяет корректность результата для нетронутых маршрутов и измеряет время перестроения в сравнении с полным пересчётом. Результатом является работающий модуль с протоколом измерений стоимости события.
16	Сборка портфолио и подготовка инженерной записки Студент собирает итоговое портфолио из всех артефактов семестра, проверяет их преемственность и полноту. Он готовит инженерную записку с обоснованием выбора алгоритмов, границами применимости и рекомендациями для двух режимов эксплуатации. Результатом является завершённое связное портфолио, представляемое на промежуточную аттестацию.

4.3. Самостоятельная работа обучающихся.

№ п/п	Вид самостоятельной работы
1	Изучение рекомендованной литературы.
2	Подготовка к практическим работам.
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	Гисин, В. Б. Дискретная математика : учебник и практикум для вузов / В. Б. Гисин. — 2-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2023. — 468 с. — (Высшее образование). — ISBN 978-5-534-16763-4.	Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/531659 (дата обращения: 28.10.2026).
2	Иванов, Б. Н. Дискретная математика и теория графов : учебное пособие для вузов / Б. Н. Иванов. — Москва : Издательство Юрайт, 2023. — 177 с. — (Высшее образование). — ISBN 978-5-534-14470-3.	Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/520078 (дата обращения: 28.10.2026).
3	Константинова, Е. В. Теория графов: алгебраическая теория : учебник для вузов / Е. В. Константинова. — Москва : Издательство Юрайт, 2025. — 123 с. — (Высшее образование). — ISBN 978-5-534-20172-7.	Образовательная платформа Юрайт [сайт]. — URL: https://urait.ru/bcode/569211 (дата обращения: 28.10.2026).
4	Ландо, С. К. Лекции о производящих функциях : учебное пособие / С. К. Ландо. — 3-е изд., испр. — Москва : МЦНМО, 2007. — 144 с. — ISBN 978-5-94057-042-4.	Лань : электронно-библиотечная система. — URL: https://e.lanbook.com/book/9364 (дата обращения: 28.10.2026).

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

ЭБС «Лань» [Электронный ресурс]. – Режим доступа: https://e.lanbook.com/ (дата обращения: 17.10.2026).

Образовательная платформа «Юрайт» [Электронный ресурс]. – Режим доступа: https://urait.ru/ (дата обращения: 17.10.2026).

Oracle Java Documentation [Электронный ресурс]. – Режим доступа: https://docs.oracle.com/en/java/ (дата обращения: 17.10.2026).

JUnit 5 User Guide [Электронный ресурс]. – Режим доступа: https://junit.org/junit5/docs/current/user-guide/ (дата обращения: 17.10.2026).

JGraphT Documentation [Электронный ресурс]. – Режим доступа: https://jgrapht.org/ (дата обращения: 17.10.2026).

Git Documentation и книга Pro Git (русский перевод) [Электронный ресурс]. – Режим доступа: https://git-scm.com/book/ru/v2 (дата обращения: 17.10.2026).

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

Операционные системы: Astra Linux / ALT Linux / РЕД ОС.

Офисные пакеты: Р7-Офис / МойОфис (для подготовки инженерной записки и протоколов по ГОСТ).

Платформа Java: Liberica JDK (отечественная сборка OpenJDK от BellSoft, включена в реестр российского ПО).

Среда разработки: IntelliJ IDEA Community Edition (open-source) / Apache NetBeans.

Система сборки: Apache Maven / Gradle (open-source).

Тестирование: JUnit 5 (open-source).

Графовая библиотека и визуализация: JGraphT (open-source Java-библиотека графовых алгоритмов) / Graphviz (open-source).

Измерение производительности: JMH (Java Microbenchmark Harness, OpenJDK) / VisualVM.

Управление версиями: Git с хостингом GitVerse / GitFlic (отечественные платформы).

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 4 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»
Института железнодорожного
транспорта И.С. Разживайкин

Согласовано:

Заведующий кафедрой ЦТУТП В.Е. Нутович

Директор ЦС IT - программ ВИШ О.Б. Проневич

Председатель учебно-методической
комиссии Н.А. Андриянова