Приложение 1

**Министерство образования и науки Республики Казахстан**

**Astana IT University**

**НАУЧНЫЙ ПРОЕКТ**

на тему:

**«Применение алгоритмов построения минимального остовного дерева (МОД)»**

Выполнили: Серикбай Арсен, Оралбай Максат

Группа: MCS-2301

Проверила: ассоц. проф. Молдахметова З. Н.

Астана

2025

Приложение 2

**СОДЕРЖАНИЕ**

**ВВЕДЕНИЕ** …………………………………………………………………….

**ГЛАВА1. Теоретические основы минимального остовного дерева**……………………………

[1.1. Сущность и содержание теории графов](#_Toc483955736)……………………………

[1.2. Остовное дерево. Задача построения остовного дерева минимального](#_Toc483955737) веса………………………………………………………………………………

[1.3. А](#_Toc483955736)лгоритм Прима…………………………………………………………

[1.4. А](#_Toc483955737)лгоритм Крускала…………………………………………………………

**ГЛАВА2. Реализация и анализ алгоритмов**……………………………

2.1. **Инструменты и технологии**…………………………

2.2. **Реализация алгоритмов**…………………………

2.3. **Применение к реальной задаче** …………………………

2.4. **Сравнительный анализ** …………………………

**ЗАКЛЮЧЕНИЕ** ………………………………………………………………..

**СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ** ………………………

**ПРИЛОЖЕНИЕ**………………………………………………………………..

**Введение**

## **Актуальность темы исследования:**

Актуальность и практическая значимость предлагаемой работы заключается в том, что в большинстве областей возникает необходимость поисков эффективных способов планирования сложных процессов и проектов, которые в итоге можно реализовать с помощью МОД и минимизации затраты в ходе производства, а далее – принимать оптимальные управленческие решения. Проблема построения минимального остовного дерева (МОД) имеет широкий спектр практических применений в таких областях, как:

- **Инфраструктура:** Оптимизация прокладки дорог, трубопроводов, кабелей и других сетей.

- **Компьютерные сети:** Минимизация затрат на подключение серверов и маршрутизаторов.

- **Энергетика:** Построение минимальных сетей энергоснабжения.

- **Генетика и биоинформатика:** Анализ филогенетических деревьев.

**Цель:**

Изучить алгоритмы построения минимального остовного дерева (алгоритмы Крускала и Прима), исследовать их эффективность.

**Задачи работы:**

1. Описать основные теоретические аспекты задачи минимального остовного дерева.

2. Реализовать алгоритмы Крускала и Прима на языке Python.

3. Сравнить эффективность алгоритмов на различных графах.

5. Визуализировать полученные результаты.

## **Объект исследования:**

Алгоритмы для построения минимального остовного дерева.

## **Предмет исследования:**

Применение алгоритмов прима и крускала для построения минимального остовного дерева.

**Гипотеза:**

В ходе работы были изучены алгоритмы построения минимального остовного дерева, их эффективность и возможности применения в различных областях. Реализация алгоритмов Прима и Крускала в среде Python позволила провести сравнительный анализ их производительности. Полученные результаты подтверждают эффективность алгоритмов при решении задач оптимизации сетей. Дальнейшие исследования могут быть направлены на изучение гибридных алгоритмов и их применение в динамических системах. Анализ экспериментальных данных показал, что гипотеза исследования полностью подтвердилась: алгоритм Прима действительно показал лучшую производительность на плотных графах, тогда как алгоритм Крускала оказался эффективнее на разреженных графах. Эти выводы могут быть полезны для выбора наиболее подходящего метода построения минимального остовного дерева в зависимости от структуры графа и задачи, которую необходимо решить.

## **Методы исследования:**

* Теоретический анализ литературы по теме минимального остовного дерева.
* Экспериментальное исследование работы алгоритмов на различных графах.
* Сравнительный анализ временной сложности алгоритмов.
* Визуализация результатов с помощью программных инструментов.

## **Теоретические основы работы:**

Работа базируется на теории графов, алгоритмах прима и крускала

## **Практическая значимость:**

Результаты работы могут быть использованы в транспортном планировании, проектировании компьютерных сетей и других сферах, где необходимо минимизировать затраты на соединение объектов.

## **Структура работы:**

Работа состоит из 3 глав:

1. Теоретические основы минимального остовного дерева.
2. Реализация и анализ алгоритмов.
3. Заключение.

**ГЛАВА1. Теоретические основы минимального остовного дерева**

**1.1. Основные понятия теории графов**  
Теория графов — это раздел дискретной математики, изучающий структуры, состоящие из вершин и рёбер, соединяющих их. Граф определяется как G = (V, E), где V — множество вершин, а E — множество рёбер, соединяющих вершины. Графы бывают ориентированными и неориентированными, взвешенными и невзвешенными, связными и несвязными.

**1.2. Задача построения минимального остовного дерева**  
Остовное дерево (spanning tree) — это подграф, содержащий все вершины исходного графа, но не содержащий циклов и имеющий минимальное возможное количество рёбер. Минимальное остовное дерево (МОД) — это остовное дерево с минимальной суммой весов рёбер. Построение МОД — это важная задача в теории графов, находящая применение в сетевом планировании, маршрутизации и инфраструктурных проектах.

**1.3. Алгоритм Прима**  
Алгоритм Прима является жадным методом поиска минимального остовного дерева. Он строит остовное дерево, начиная с произвольной вершины, и на каждом шаге добавляет к дереву ближайшую вершину, соединенную минимальным по весу ребром. Основные шаги алгоритма:

1. Выбираем произвольную начальную вершину.
2. Добавляем минимальное ребро, соединяющее вершину с остальной частью графа.
3. Повторяем процесс, пока все вершины не будут включены в дерево.

**1.4. Алгоритм Крускала**  
Алгоритм Крускала — еще один жадный алгоритм, который сортирует все рёбра по весу и последовательно добавляет их в остовное дерево, если они не создают цикла. Основные шаги алгоритма:

1. Сортируем рёбра графа по возрастанию веса.
2. Инициализируем множество остовного дерева (изначально пустое).
3. Последовательно добавляем рёбра, если они не образуют цикла.
4. Повторяем процесс, пока не будет добавлено (V - 1) рёбер.

Алгоритмы Прима и Крускала широко при

**ГЛАВА2. Реализация и анализ алгоритмов**……………………………

**3.1. Инструменты и технологииДля реализации используются язык Python, библиотеки NetworkX и Matplotlib, а также среды разработки Jupyter Notebook и PyCharm.**

**3.2. Реализация алгоритмов**  
Алгоритмы Прима и Крускала были реализованы с использованием Python. Код включает функции для обработки графов, сортировки рёбер и построения минимального остовного дерева.

**3.3. Применение к реальной задаче**  
Алгоритмы были протестированы на задаче оптимального построения сети дорог между городами, позволяя минимизировать затраты на строительство.

**3.4. Сравнительный анализ**  
Были проведены тестирования на различных графах. Алгоритм Крускала оказался быстрее на разреженных графах, тогда как алгоритм Прима показал лучшую производительность на плотных графах.

**ЗАКЛЮЧЕНИЕ**

В ходе работы были изучены алгоритмы построения минимального остовного дерева, их эффективность и возможности применения в различных областях. Реализация алгоритмов Прима и Крускала в среде Python позволила провести сравнительный анализ их производительности. Полученные результаты подтверждают эффективность алгоритмов при решении задач оптимизации сетей. Дальнейшие исследования могут быть направлены на изучение гибридных алгоритмов и их применение в динамических системах.

**СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

Кормен Т. Х. и др. Часть VI. Алгоритмы для работы с графами // Алгоритмы: построение и анализ = Introduction to Algorithms. — 2-е изд. — М.: Вильямс, 2006. – 1296 c.

Лутц М. Программирование на Python, том I, 4-е издание. – Пер. с англ. – СПб.: Символ-Плюс, 2011. – 992 с.

Лутц М. Программирование на Python, том II, 4-е издание. – Пер. с англ. – СПб.: Символ-Плюс, 2011. – 992 с.

<https://cyberleninka.ru/article/n/postroenie-minimalnogo-ostovnogo-dereva-algoritmom-boruvki-programmnaya-realizatsiya/viewer>

<https://cyberleninka.ru/article/n/nahozhdenie-ostovnogo-dereva-minimalnogo-vesa-s-primeneniem-algoritma-kraskala-i-algoritma-prima/viewer>

**ПРИЛОЖЕНИЯ** …………………………………………………………………