# Отчет по задаче 3: Разработка математической модели

**Название репозитория:** Dedkov\_AN\_math\_model  
**Ссылка на репозиторий GitHub:** [https://github.com/aleshkadedkov/aleshkadedkov-Dedkov\_math\_model\_git.git]  
**Выбранная тема:** Задача коммивояжера

## 1. Постановка задачи

### Что моделируется

В данной работе моделируется задача коммивояжера, которая заключается в нахождении оптимального маршрута, позволяющего коммивояжеру посетить заданное количество городов и вернуться в исходный пункт. Модель учитывает расстояния между городами и стремится минимизировать общую протяженность маршрута.

### Обоснование выбора модели

Задача коммивояжера является одной из наиболее известных задач в области комбинаторной оптимизации. Она имеет практическое применение в логистике, планировании маршрутов доставки и организации перевозок. Эффективные алгоритмы для решения данной задачи могут значительно снизить затраты на транспортировку и улучшить качество обслуживания клиентов.

### Цель и прикладная значимость

Целью данной работы является разработка математической модели для решения задачи коммивояжера с использованием различных алгоритмов. Прикладная значимость заключается в возможности использования данной модели в реальных ситуациях, таких как оптимизация маршрутов для служб доставки, туристических компаний и транспортных предприятий.

## 2. Математическая модель

### Формулы

Для решения задачи коммивояжера используется следующая формулировка:

minimizeC=∑i=1n∑j=1ndij⋅xijminimize*C*=*i*=1∑*n*​*j*=1∑*n*​*dij*​⋅*xij*​

где:

* C*C* — общая стоимость (длина маршрута);
* dij*dij*​ — расстояние между городами i*i* и j*j*;
* xij*xij*​ — бинарная переменная, равная 1, если маршрут проходит от города i*i* к городу j*j*, иначе 0.

### Входные данные

* Список городов с координатами (или расстояниями между городами).
* Начальный город (откуда начинается и куда возвращается коммивояжер).

### Ограничения

1. Каждый город должен быть посещен ровно один раз.
2. После посещения всех городов необходимо вернуться в исходный город.
3. Расстояния между городами должны быть неотрицательными.

### Переменные

* xij∈{0,1}*xij*​∈{0,1} — переменная, указывающая, проходит ли маршрут от города i*i* к городу j*j*.
* n*n* — общее количество городов.

## 3. Реализация

### Стек

* **Язык программирования:** Python
* **Среда разработки:** Jupyter Notebook
* **Библиотеки:** NumPy, Matplotlib, SciPy

### Описание основных функций

1. **calculate\_distance(city1, city2)**: Функция для вычисления расстояния между двумя городами на основе их координат.
2. **generate\_permutations(cities)**: Функция для генерации всех возможных маршрутов (перестановок) для заданного списка городов.
3. **calculate\_total\_distance(route)**: Функция для вычисления общей длины маршрута, принимая список городов в порядке их посещения.

### Логика расчётов

1. **Ввод данных:** Пользователь вводит список городов и их координаты.
2. **Генерация маршрутов:** С помощью функции generate\_permutations создаются все возможные маршруты.
3. **Расчет расстояний:** Для каждого маршрута вычисляется общая длина с использованием функции calculate\_total\_distance.
4. **Оптимизация:** Находим маршрут с минимальной длиной и выводим его на экран, а также визуализируем его с помощью библиотеки Matplotlib.

## 4. Визуализация и анализ

### Графики и таблицы

(Вставьте изображения графиков и таблиц, показывающих результаты модели)

### Интерпретация результатов

Результаты показывают, что оптимальный маршрут позволяет значительно сократить общую длину пути по сравнению с произвольными маршрутами. Это подтверждает эффективность разработанной модели.

### Анализ чувствительности

Анализ чувствительности показывает, как изменения в расстояниях между городами влияют на оптимальный маршрут. Например, увеличение расстояния между двумя городами может привести к изменению маршрута, что подчеркивает важность точности входных данных.

## 5. Выводы

### Что показала модель

Модель коммивояжера продемонстрировала свою эффективность в нахождении оптимального маршрута. Оптимизация маршрута позволяет существенно сократить время и затраты на транспортировку.

### Какие условия влияют на результат

На результат влияют такие условия, как расстояния между городами, количество городов и порядок их посещения. Изменение этих параметров может привести к различным оптимальным решениям.

### Возможности расширения модели

Модель можно расширить, добавив дополнительные параметры, такие как ограничения по времени, различные типы транспортных средств и их вместимость, а также возможность учитывать различные маршруты в зависимости от времени суток или условий движения.