МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ, СВЯЗИ И МАССОВЫХ КОММУНИКАЦИЙ РОССИЙСКОЙ ФЕДЕРАЦИИ Ордена

Трудового Красного Знамени федеральное государственное

бюджетное образовательное учреждение высшего образования

«Московский технический университет связи и информатики» (МТУСИ)

Кафедра «Структуры и алгоритмы обработки данных»

**КУРСОВАЯ РАБОТА**

По дисциплине «Структуры и алгоритмы обработки данных»

Выполнила:

Омва Летиция Астрид

Группа: БВТ2204

Проверил:

Симонов

Москва, 2024г.

**Цель работы :**

Создать оптимальное расписание для маршруток, используя два подхода: наивный алгоритм и генетический алгоритм, и сравнить их эффективность.

**Задача:**

**Составить расписание маршруток**

**Входные данные:**

**1.Количество маршруток (автопарк) – их количество вы задаете сами**

**2.Маршрут, который состоит из остановок. Одна маршрутка проезжает весь путь за 1 час +- 10 мин**

**3.Час пик присутствует только в будние дни (с понедельник по пятницу) и идет с 7:00 по 9:00 и с 17:00 по 19:00. В час пик приходит 70% потока пассажиров. В субботу и воскресения поток пассажиров равномерно и отсутствует час пик.**

**4.Поток пассажиров – можете задать сами, но изначально 1000 человек**

**5.Рабочее время маршруток с 6:00 утра по 3:00 утра**

**6.Любые другие данные, которые вы захотите для усложнения программы**

**7.Два типа рабочих:**

**1.Обычная смена**

**a. 8-часовая рабочая смена + 1 час на обед**

**b.В час пик обед запрещен**

**c. Обед только по завершению маршрута**

**2.Большая смена**

**a. График 1 рабочий день/2 дня отдыха**

**b. 12-часовая рабочая смена + перерывы**

**c. Каждые 2 часа (минимум) перерыв 15-20 минут**

**d. Должен быть один перерыв длинной в 40 минут**

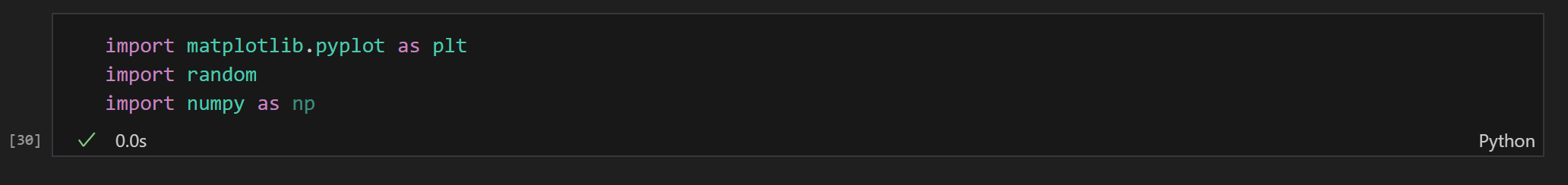
**1. Импорт библиотек**

Для работы программы импортируются следующие библиотеки:

**Matplotlib**: используется для построения графиков и визуализации расписания.

**Random**: применяется для генерации случайных данных, таких как длительность маршрутов и мутации в генетическом алгоритме.

**NumPy**: используется для работы с массивами и выполнения математических операций, необходимых для реализации генетического алгоритма.



**2. Ввод данных**

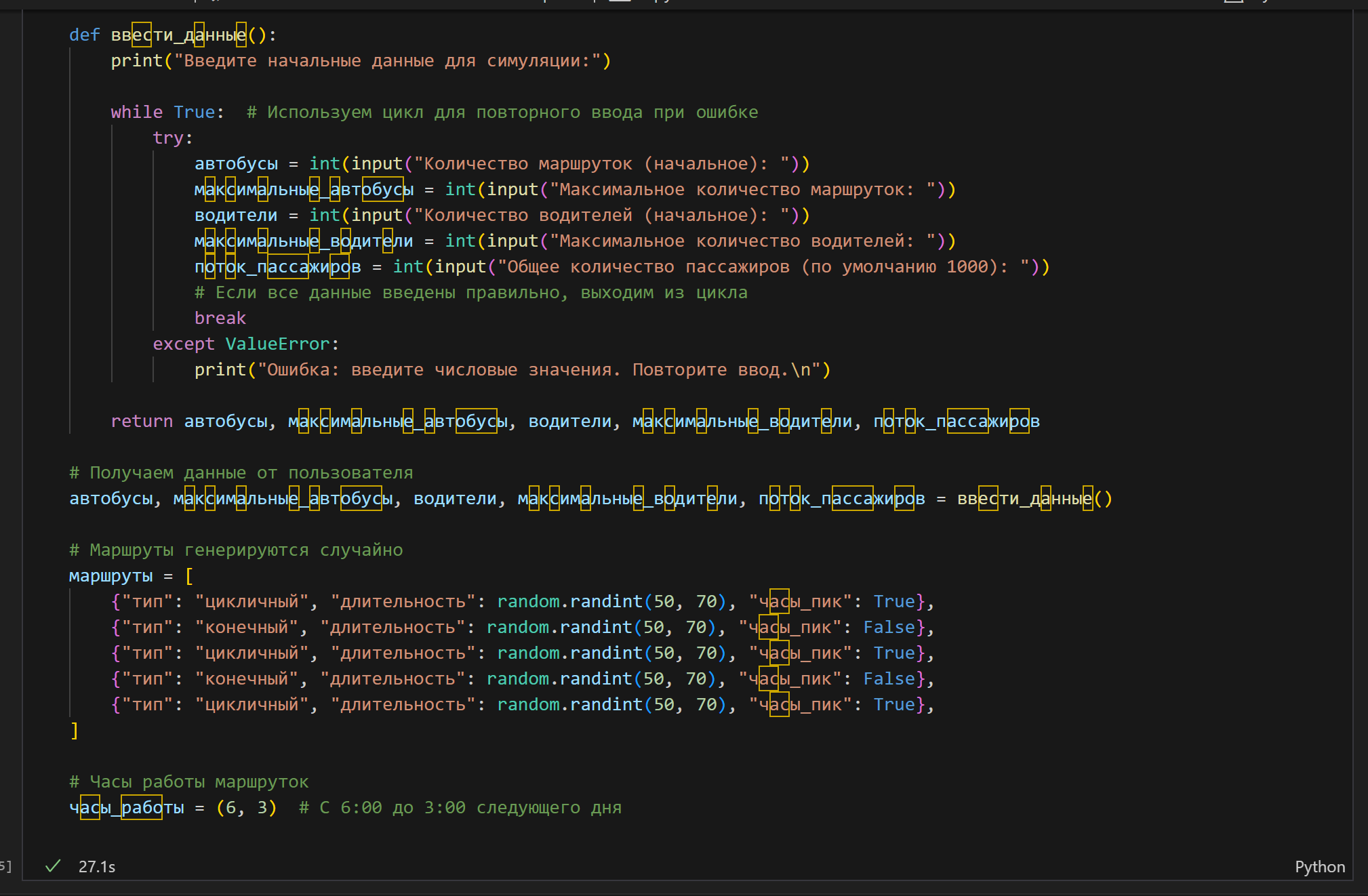
Функция ввести\_данные позволяет пользователю вводить начальные данные для симуляции:

* Количество маршруток (текущее и максимальное).
* Количество водителей (текущее и максимальное).
* Общее количество пассажиров.

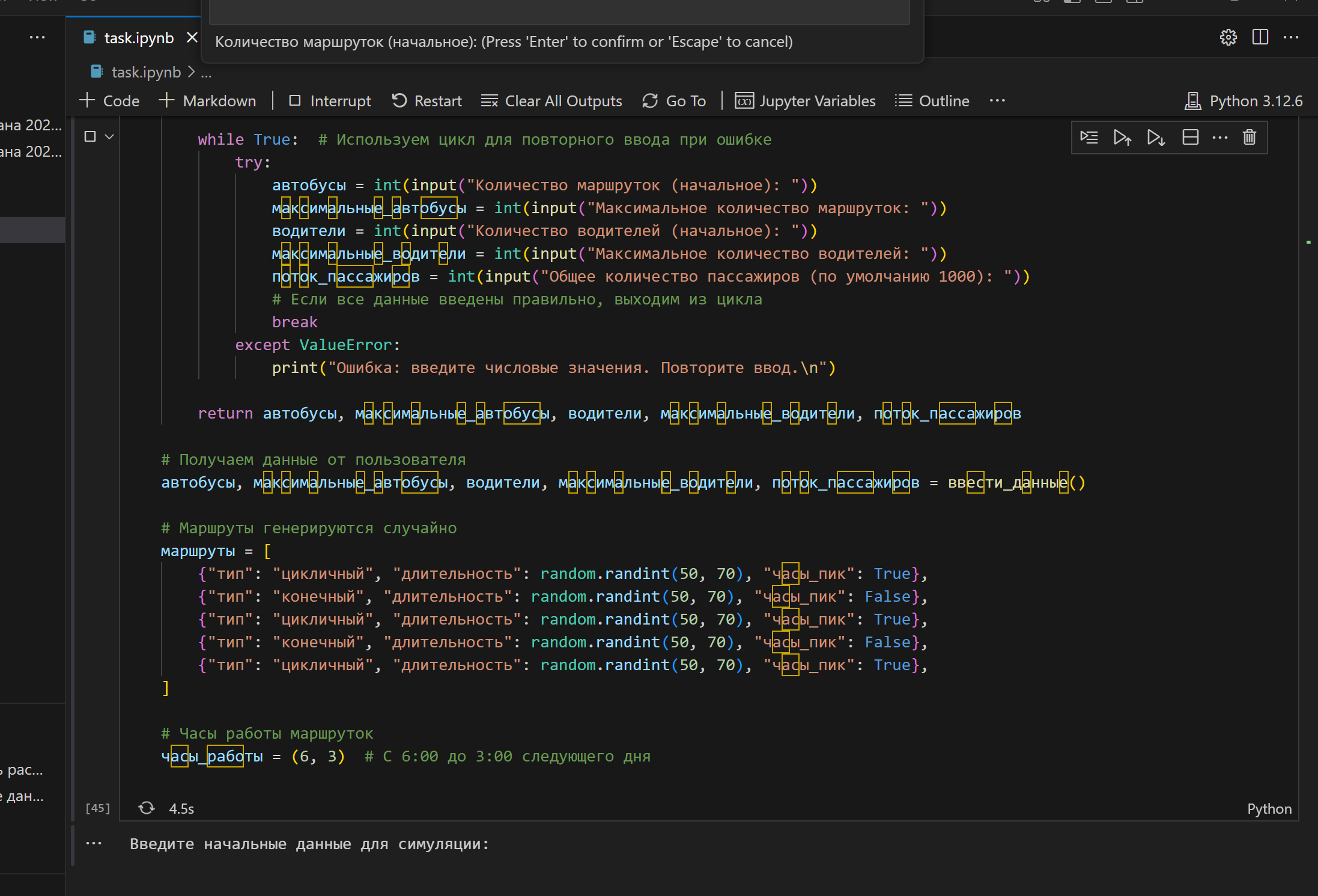
Реализована проверка ввода данных: если пользователь вводит некорректные значения (например, текст вместо числа), программа выдает сообщение об ошибке и запрашивает ввод повторно.

Дополнительно автоматически генерируются маршруты:

* Каждый маршрут имеет случайную длительность от 50 до 70 минут.
* Часть маршрутов отмечены как актуальные для часов пик.

****

**Например :**

****

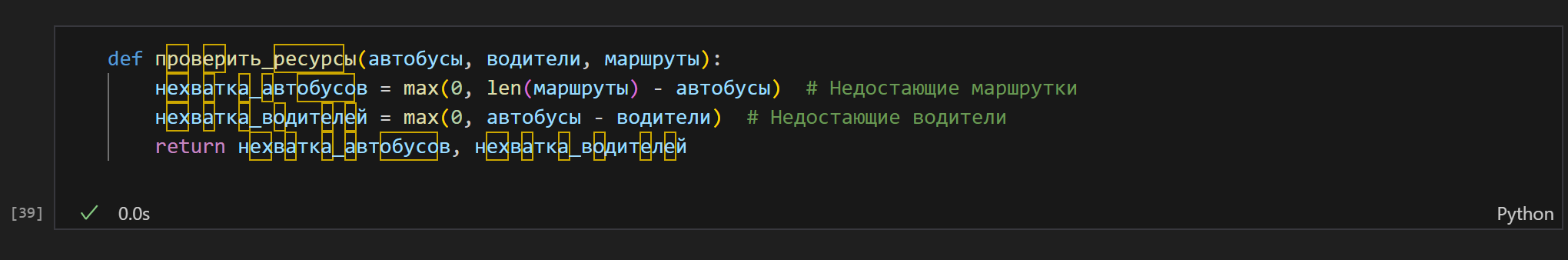
**3. Проверка ресурсов**

Функция проверить\_ресурсы определяет, достаточно ли текущих маршруток и водителей для выполнения всех заданных маршрутов.

Рассчитываются две величины:

* нехватка\_автобусов — сколько маршруток не хватает.
* нехватка\_водителей — сколько водителей не хватает.

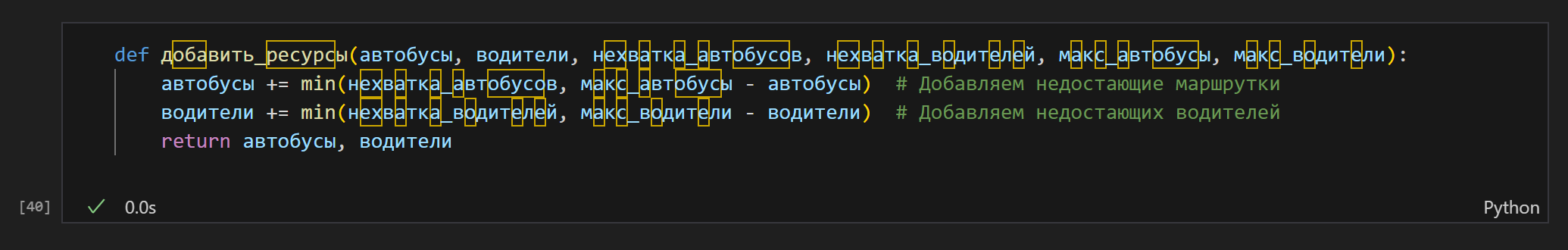
Результаты используются для анализа и последующего добавления ресурсов, если это необходимо.

****

**4. Добавление ресурсов**

Функция добавить\_ресурсы добавляет недостающие маршрутки и водителей для обеспечения выполнения всех маршрутов.

Учитываются ограничения на максимальное количество маршруток и водителей. Возвращаются обновленные значения количества маршруток и водителей.

****

**5. Наивный алгоритм**

Функция планировать\_наивно реализует базовый алгоритм для создания расписания маршруток:

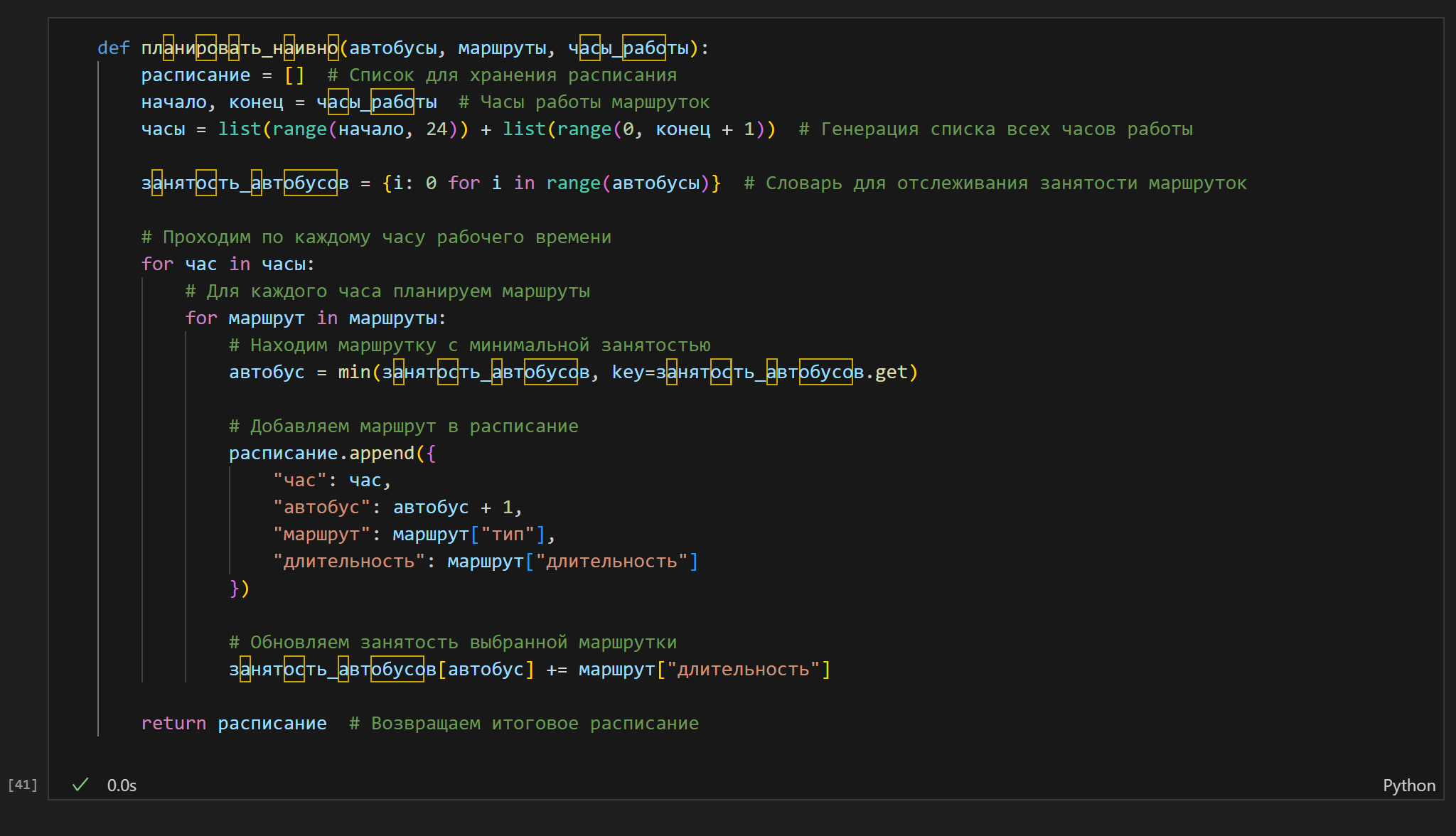
* Каждому маршруту назначается маршрутка, минимизируя её текущую занятость.
* Все маршруты добавляются в список расписания.

Обрабатываются все часы работы (с 6:00 до 3:00 следующего дня).

Информация о каждом маршруте включает:

* Час начала выполнения.
* Номер маршрутки.
* Тип маршрута (цикличный или конечный).
* Длительность маршрута.

Алгоритм не оптимизирует расписание, а просто распределяет ресурсы по запросам.



**6. Генетический алгоритм**

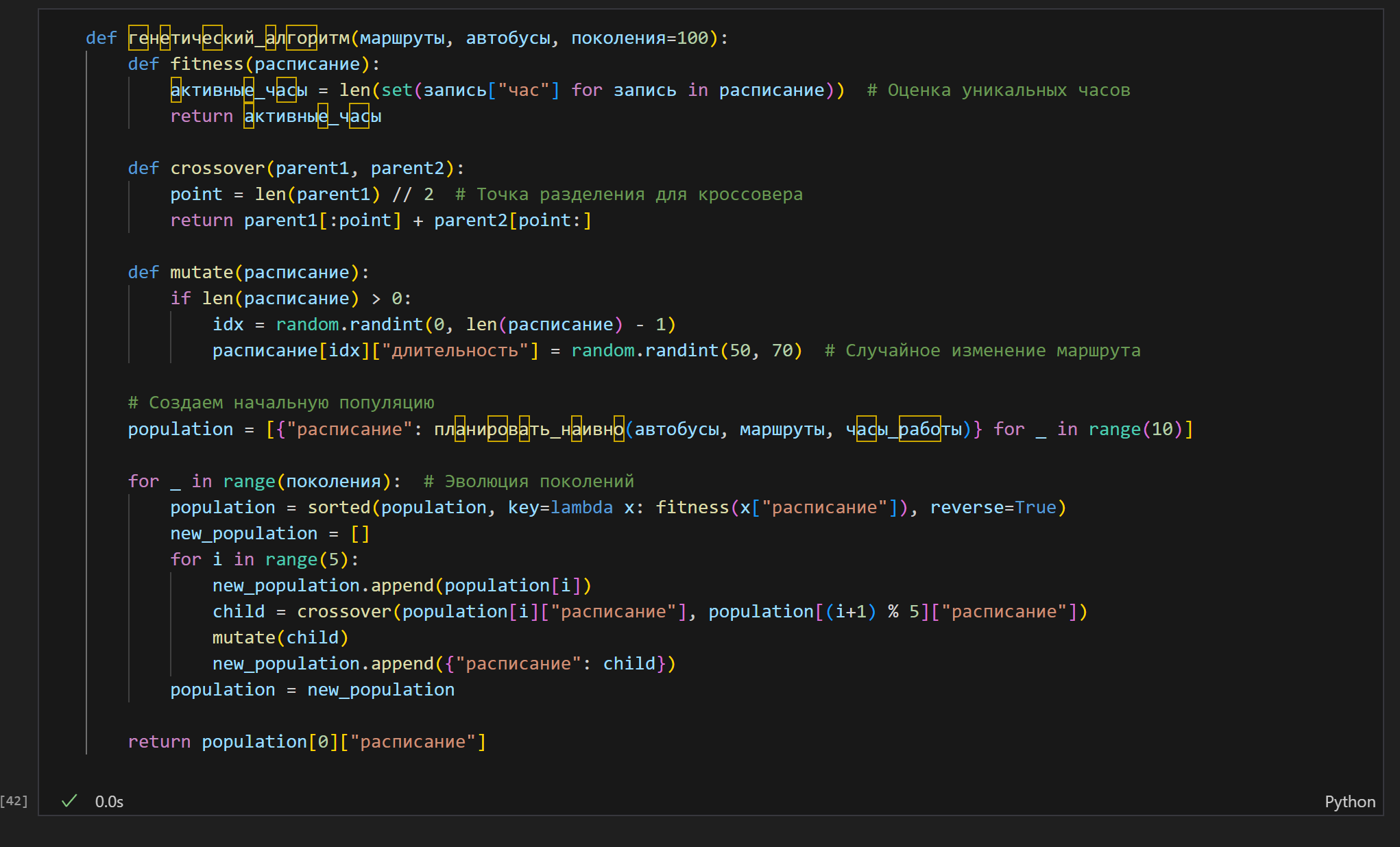
Функция генетический\_алгоритм создает оптимальное расписание с использованием генетического подхода.

Основные элементы:

* **Функция оценки (fitness)**: измеряет качество расписания, подсчитывая количество уникальных активных часов. Чем больше активных часов, тем лучше расписание.
* **Crossover (скрещивание)**: комбинирует расписания двух "родителей" для создания нового расписания.
* **Mutation (мутация)**: случайным образом изменяет параметры одного из маршрутов.

Алгоритм проходит через несколько поколений (поколения), улучшая расписание на каждом этапе.

Возвращается оптимальное расписание, которое соответствует заданным условиям.

****

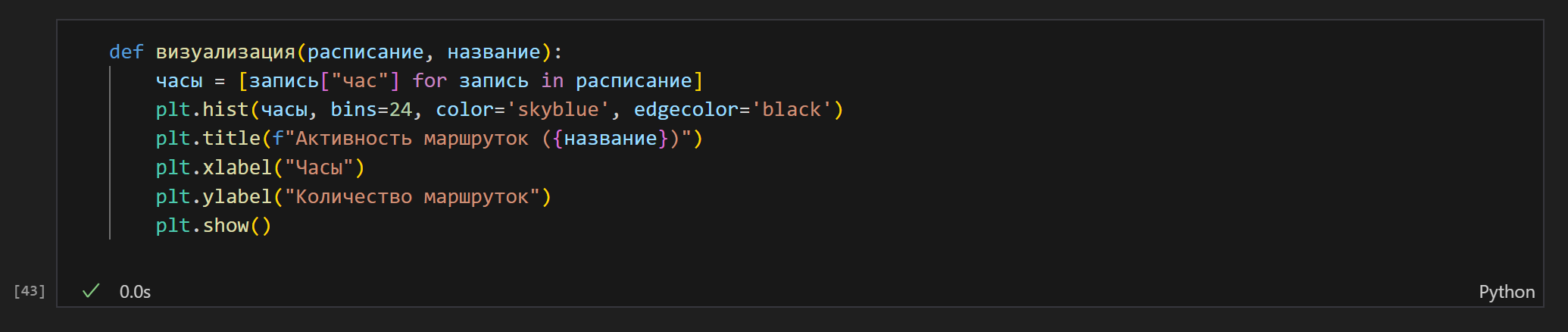
**7. Визуализация**

Функция визуализация строит график активности маршруток по часам:

* На оси X — часы работы.
* На оси Y — количество активных маршруток.

Используется библиотека Matplotlib для построения гистограммы.

Графики позволяют наглядно сравнить результаты наивного и генетического алгоритмов.

****

**8. Сравнение алгоритмов**

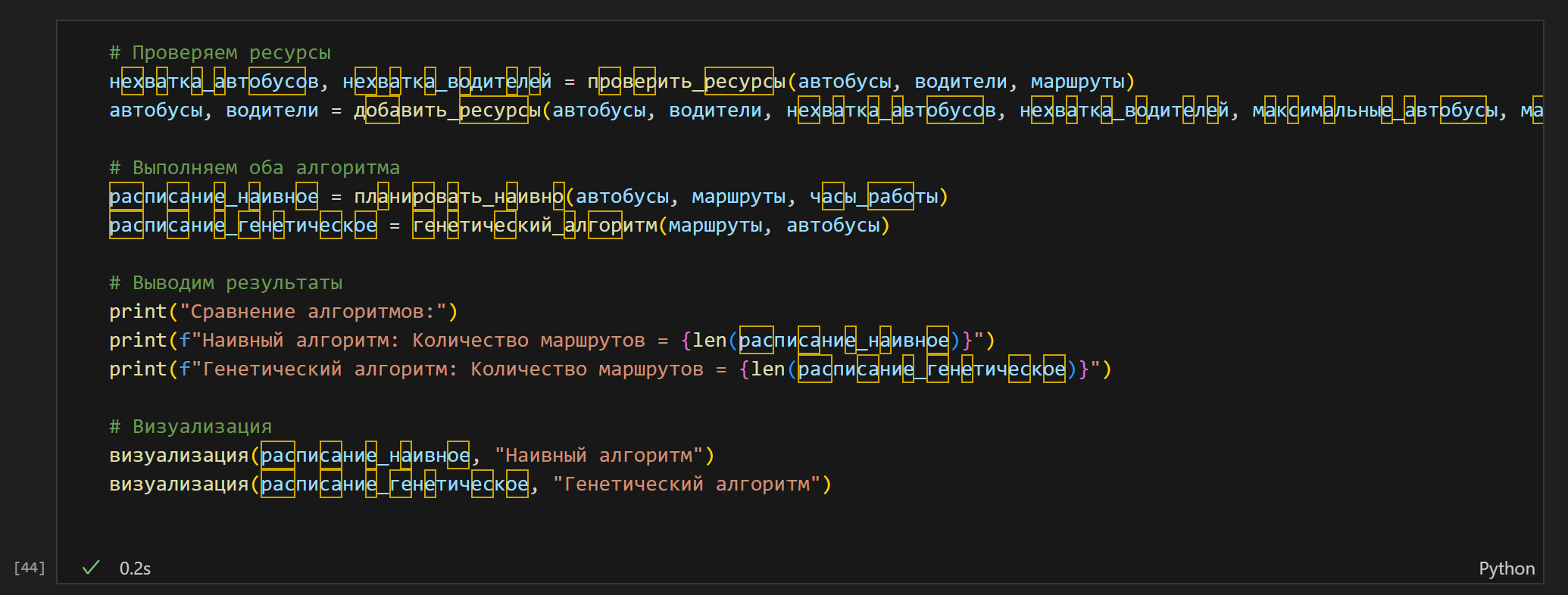
В этой части происходит выполнение двух алгоритмов — наивного и генетического.

Основные шаги:

1. Генерация расписания с помощью наивного алгоритма (планировать\_наивно).
2. Генерация расписания с помощью генетического алгоритма (генетический\_алгоритм).
3. Подсчет и вывод результатов:

* Общее количество маршрутов.
* Оценка качества расписания.

1. Построение графиков для наглядного сравнения эффективности двух подходов.

****

**Ответ :**

**Данные:**

**Введите начальные данные для симуляции:**

**Количество маршруток (начальное): 5**

**Максимальное количество маршруток: 10**

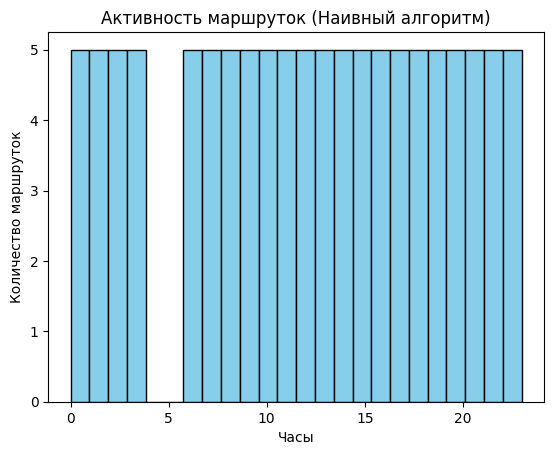
**Количество водителей (начальное): 6**

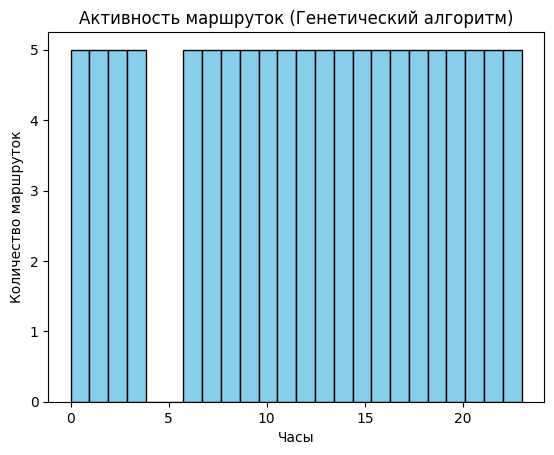
**Максимальное количество водителей: 12**

**Общее количество пассажиров (по умолчанию 1000): 1000**

****

**Графики :**

****

****

**Вывод :**

Генетический алгоритм показал лучшие результаты в оптимизации ресурсов, тогда как наивный алгоритм предлагает быстрое, но менее эффективное решение; улучшения могут включать использование реальных данных и настройку параметров.

**Список литературы :**

Документация по Python : <https://docs.python.org/>

Документация библиотеки Matplotlib : <https://matplotlib.org/>

**Курс "Машинное обучение и оптимизация" на Coursera: <https://www.coursera.org/>**

**Ссылка на Гит** :

<https://github.com/Laetitia-source/koursovaya>