

Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования «ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
УПРАВЛЕНИЯ» Институт заочного образования

Выполнила: Белова Ксения Юрьевна

1. **Описание кода:**
2. Данный код на Python представляет собой программу планирования туристической поездки, использующую жадный алгоритм для выбора наиболее ценных достопримечательностей, которые можно посетить за ограниченное количество дней.
3. **Функциональные компоненты:**
4. **greedy_travel_planner(attractions, max_days):** Эта функция является основной частью алгоритма планирования. Она принимает на вход список словарей **attractions**, где каждый словарь содержит информацию о достопримечательности (название, стоимость, продолжительность посещения), а также максимальное количество дней для поездки **max_days**. Функция использует жадный алгоритм для выбора наиболее ценных достопримечательностей, которые можно посетить за отведенное время. Достопримечательности сортируются по убыванию "ценности за день" (отношение стоимости к продолжительности). Затем функция перебирает отсортированные достопримечательности и добавляет их в план поездки, если это не превышает общее количество доступных дней. Функция возвращает словарь, содержащий план поездки (список выбранных достопримечательностей), общую ценность поездки и общую продолжительность поездки в днях.
5. **get_attraction_from_user():** Эта функция запрашивает у пользователя информацию о достопримечательности через консоль. Она запрашивает название, стоимость (целое число) и продолжительность посещения (целое число дней). Функция проверяет корректность введенных данных (стоимость и продолжительность должны быть положительными числами) и повторяет запрос, если пользователь ввел некорректные данные. Функция возвращает словарь с информацией о введенной достопримечательности.
6. **Алгоритм работы программы:**
7. **Получение данных о достопримечательностях:** Программа запрашивает у пользователя данные о достопримечательностях, используя функцию **get_attraction_from_user()**, и добавляет их в список **attractions**. Пользователь может добавлять достопримечательности до тех пор, пока не введет "нет" в ответ на вопрос о добавлении еще одной достопримечательности.

8. **Получение максимального количества дней:** Программа запрашивает у пользователя максимальное количество дней для поездки и проверяет, что введенное значение является положительным числом.
9. **Планирование поездки:** Программа вызывает функцию `greedy_travel_planner(attractions, max_days)` для планирования поездки на основе введенных данных о достопримечательностях и максимальном количестве дней.
10. **Вывод результатов:** Программа выводит на консоль результаты планирования: список выбранных достопримечательностей, общую ценность поездки и общую продолжительность поездки в днях.
11. **Принципы, заложенные в программе:**
12. **Жадный алгоритм:** Программа использует жадный алгоритм для выбора наиболее ценных достопримечательностей на каждом шагу, не учитывая, как этот выбор повлияет на дальнейшие возможности. Это простой и быстрый способ планирования поездки, но он не гарантирует оптимальное решение (максимальную общую ценность).
13. **Взаимодействие с пользователем:** Программа позволяет пользователю ввести данные о достопримечательностях и максимальном количестве дней, что делает ее интерактивной и гибкой.
14. **Проверка ввода данных:** Программа проверяет корректность введенных данных (стоимость и продолжительность должны быть положительными числами), что предотвращает ошибки и обеспечивает правильную работу алгоритма.