

Алгоритм работы программы по теме курсовой работы:

Программа читает файл «citymap.txt», заполненный по определенному шаблону, и забирает оттуда список всех городов и расстояния между ними, составленные в виде матрицы, где ноль – это отсутствие маршрута, после чего дает возможность оператору ввести стартовый город и города, по которым должен пройти маршрут.

После этого, при помощи функции библиотеки algorithm «next\_permutation» в массив строк записываются все возможные комбинации городов из заданных оператором на этапе ввода стартового города и городов–адресатов.

Затем цикл проходит по каждой комбинации, подсчитывая для нее путь минимальной длины при помощи применения реализации алгоритма Дейкстры (примечание 1) к каждой паре точек, использование которого становится необходимым, поскольку в каждый город может вести более одного пути, что создает вероятность попасть из одного города в другой быстрее, используя промежуточные пункты. По окончании каждой итерации цикла, кроме первой, полученная длина пути сравнивается с минимальной найденной длиной, и заменяет ее в случае, если оказывается меньше.

По итогу выполнения программы выводится схема маршрута с отметкой начального пункта, посещенных пунктов маршрута и посещенных точек, являющихся транзитными.

Примечание 1:

Присутствующая реализация алгоритма Дейкстры работает по следующему принципу:

- 1) Для вычисления мин. расстояния берутся идентификаторы двух точек.
- 2) Создаются два одномерных массива `v_sum` и `v_visited`, объем которых равняется общему количеству городов на карте. Первый – для хранения минимальной стоимости достижения каждого города из точки начала поиска, второй – для отметки о посещении города.
- 3) Запускается цикл, чья цель – пройти по всем городам карты, с целью определения стоимости доступности для каждого города относительно выбранного города. В процессе выполнения массив `v_sum` заполняется и при нахождении меньшей стоимости для города корректируется, а в массиве `v_visited` отмечается статус посещенности города (0 или 1).
- 4) По окончании цикла подсчета доступности запускается цикл восстановления пути, который проверяет, посредством посещения каких именно точек получился минимальный путь, проходя из конца в начало. В процессе в строковую переменную ведется запись маршрута.
- 5) После выполнения этих двух циклов в переменную `sum` записывается найденный минимальный путь между двух точек. Цикл, в который вложен этот алгоритм, переходит на следующую итерацию, беря другую пару точек и помещая их в те же самые условия.