«Белорусский государственный технологический университет»

Кафедра Информационных Систем и Технологий

**Курс “Математическое программирование”**

**Отчет по лабораторной работе №3**

**Решение задачи о коммивояжере методом ветвей и границ**

Выполнил: Грудинский П.В.

ФИТ 2 курс, 1 группа

Минск 2020

**Метод ветвей и границ** ([англ.](https://ru.wikipedia.org/wiki/%D0%90%D0%BD%D0%B3%D0%BB%D0%B8%D0%B9%D1%81%D0%BA%D0%B8%D0%B9_%D1%8F%D0%B7%D1%8B%D0%BA) *branch and bound*) — общий алгоритмический [метод](https://ru.wikipedia.org/wiki/%D0%9C%D0%B5%D1%82%D0%BE%D0%B4) для нахождения оптимальных решений различных задач оптимизации, особенно дискретной и [комбинаторной оптимизации](https://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BC%D0%B1%D0%B8%D0%BD%D0%B0%D1%82%D0%BE%D1%80%D0%BD%D0%B0%D1%8F_%D0%BE%D0%BF%D1%82%D0%B8%D0%BC%D0%B8%D0%B7%D0%B0%D1%86%D0%B8%D1%8F). Метод является развитием метода [полного перебора](https://ru.wikipedia.org/wiki/%D0%9F%D0%BE%D0%BB%D0%BD%D1%8B%D0%B9_%D0%BF%D0%B5%D1%80%D0%B5%D0%B1%D0%BE%D1%80), в отличие от последнего — с отсевом подмножеств допустимых решений, заведомо не содержащих оптимальных решений.

**1.Исходная матрица расстояний:**

Формируем матрицу расстояний по исходным данным:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Город | 1 | 2 | 3 | 4 | 5 |
| 1 | ∞ | 2 | 22 | ∞ | 1 |
| 2 | 1 | ∞ | 16 | 67 | 83 |
| 3 | 3 | 3 | ∞ | 86 | 50 |
| 4 | 18 | 57 | 4 | ∞ | 3 |
| 5 | 92 | 67 | 52 | 14 | ∞ |

Итерация 1.Найдем в каждой строке минимальное значение:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Город | 1 | 2 | 3 | 4 | 5 |  |
| 1 | ∞ | 2 | 22 | ∞ | 1 | **1** |
| 2 | 1 | ∞ | 16 | 67 | 80 | **1** |
| 3 | 3 | 3 | ∞ | 86 | 59 | **3** |
| 4 | 18 | 57 | 4 | ∞ | 3 | **3** |
| 5 | 92 | 67 | 52 | 14 | ∞ | **14** |

Произведем редукцию строк – вычтем из каждого элемента строки соответствующий ей минимальный элемент:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Город | 1 | 2 | 3 | 4 | 5 |  |
| 1 | ∞ | 1 | 21 | ∞ | 0 | 1 |
| 2 | 0 | ∞ | 15 | 66 | 79 | 1 |
| 3 | 0 | 0 | ∞ | 83 | 56 | 3 |
| 4 | 15 | 54 | 1 | ∞ | 0 | 3 |
| 5 | 78 | 53 | 38 | 0 | ∞ | 14 |

Теперь найдем минимальное значение в каждом столбце:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Город | 1 | 2 | 3 | 4 | 5 |
| 1 | ∞ | 1 | 21 | ∞ | 0 |
| 2 | 0 | ∞ | 15 | 66 | 79 |
| 3 | 0 | 0 | ∞ | 83 | 56 |
| 4 | 15 | 54 | 1 | ∞ | 0 |
| 5 | 78 | 53 | 38 | 0 | ∞ |
|  | **0** | **0** | **1** | **0** | **0** |

Произведем редукцию столбцов:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Город | 1 | 2 | 3 | 4 | 5 |
| 1 | ∞ | 1 | 20 | ∞ | 0 |
| 2 | 0 | ∞ | 14 | 66 | 79 |
| 3 | 0 | 0 | ∞ | 83 | 56 |
| 4 | 15 | 54 | 0 | ∞ | 0 |
| 5 | 78 | 53 | 37 | 0 | ∞ |
|  | 0 | 0 | 1 | 0 | 0 |

Корневая вершина: **φ=22+1=23.** Эту величину можно принять в качестве нижней границы длины кратчайшего кольцевого маршрута, проходящего через все города.

Граф имеет вид:

**φ=22+1=23**

В получившейся матрице найдем нулевые клетки и вычислим для нулевых клеток оценки:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Город | 1 | 2 | 3 | 4 | 5 |
| 1 | ∞ | 1 | 20 | ∞ | 0 (1) |
| 2 | 0 (14) | ∞ | 14 | 66 | 79 |
| 3 | 0 (0) | 0 (1) | ∞ | 83 | 56 |
| 4 | 15 | 54 | 0 (14) | ∞ | 0 (0) |
| 5 | 78 | 53 | 37 | 0 (103) | ∞ |

Найдем среди вычисленных оценок нулевых клеток максимальную оценку:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Город | 1 | 2 | 3 | 4 | 5 |
| 1 | ∞ | 1 | 20 | ∞ | 0 (1) |
| 2 | 0 (14) | ∞ | 14 | 66 | 79 |
| 3 | 0 (0) | 0 (1) | ∞ | 83 | 56 |
| 4 | 15 | 54 | 0 (14) | ∞ | 0 (0) |
| 5 | 78 | 53 | 37 | 0 (103) | ∞ |

Заменяем нулевой элемент с максимальной оценкой на бесконечность:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Город | 1 | 2 | 3 | 4 | 5 |  |
| 1 | ∞ | 1 | 20 | ∞ | 0 | 0 | |
| 2 | 0 | ∞ | 14 | 66 | 79 | 0 |
| 3 | 0 | 0 | ∞ | 83 | 56 | 0 |
| 4 | 15 | 54 | 0 | ∞ | 0 | 0 |
| 5 | 78 | 53 | 37 | ∞ | ∞ | 37 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Город | 1 | 2 | 3 | 4 | 5 |
| 1 | ∞ | 2 | 20 | ∞ | 0 |
| 2 | 0 | ∞ | 14 | 66 | 79 |
| 3 | 0 | 0 | ∞ | 83 | 56 |
| 4 | 15 | 54 | 0 | ∞ | 0 |
| 5 | 41 | 16 | 0 | ∞ | ∞ |
|  | **0** | **0** | **0** | **66** | **0** |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Город | 1 | 2 | 3 | 4 | 5 |
| 1 | ∞ | 1 | 20 | ∞ | 0 |
| 2 | 0 | ∞ | 14 | 0 | 79 |
| 3 | 0 | 0 | ∞ | 17 | 56 |
| 4 | 15 | 54 | 0 | ∞ | 0 |
| 5 | 41 | 16 | 0 | ∞ | ∞ |

Граф имеет вид:

**φ=22+1=23**

**φ=23+103=126**

Проведем редукцию матрицы – полностью исключим строку и столбец, соответствующие нулевой клетке с максимальной оценкой.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Город | 1 | 2 | 3 | 5 |
| 1 | ∞ | 1 | 20 | 0 |
| 2 | 0 | ∞ | 14 | 79 |
| 3 | 0 | 0 | ∞ | 56 |
| 4 | 15 | 54 | 0 | 0 |

На этом этапе, чтобы исключить возврат в уже посещенный город, мы ставим символ '∞' на обратный путь.

Необходимо поставить символ '∞' во все клетки, которые соответствуют путям, замыкающим маршрут раньше времени. Т. о. отсекаются возможные 'петли' (подциклы). У нас их нет.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Город | 1 | 2 | 3 | 5 |
| 1 | ∞ | 1 | 20 | 0 |
| 2 | 0 | ∞ | 14 | 79 |
| 3 | 0 | 0 | ∞ | 56 |
| 4 | 15 | 54 | 0 | ∞ |

Итерация 2.Найдем в каждой строке минимальное значение. И произведем редукцию строк.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Город | 1 | 2 | 3 | 5 |  |
| 1 | ∞ | 1 | 20 | 0 | 0 |
| 2 | 0 | ∞ | 14 | 79 | 0 |
| 3 | 0 | 0 | ∞ | 56 | 0 |
| 4 | 15 | 54 | 0 | ∞ | 0 |

Теперь найдем минимальное значение в каждом столбце и произведем редукцию столбцов:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Город | 1 | 2 | 3 | 5 |
| 1 | ∞ | 1 | 20 | 0 |
| 2 | 0 | ∞ | 14 | 79 |
| 3 | 0 | 0 | ∞ | 56 |
| 4 | 15 | 54 | 0 | ∞ |
|  | 0 | 0 | 0 | 0 |

Граф имеет вид:

**φ=22+1=23 φ=23+0=23**

**φ=23+103=126**

В получившейся матрице найдем нулевые клетки и вычислим для нулевых клеток оценки:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Город | 1 | 2 | 3 | 5 |
| 1 | ∞ | 1 | 20 | **0 (57)** |
| 2 | **0 (14)** | ∞ | 14 | 79 |
| 3 | **0 (0)** | **0 (1)** | ∞ | 56 |
| 4 | 15 | 54 | **0 (29)** | ∞ |

Найдем среди вычисленных оценок нулевых клеток максимальную оценку: **0 (47).**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Город | 1 | 2 | 3 | 5 |
| 1 | ∞ | 1 | 20 | ∞ |
| 2 | 0 | ∞ | 14 | 79 |
| 3 | 0 | 0 | ∞ | 56 |
| 4 | 15 | 54 | 0 | ∞ |

Заменяем нулевой элемент с максимальной оценкой на бесконечность:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Город | 1 | 2 | 3 | 5 |  |
| 1 | ∞ | 1 | 20 | ∞ | **1** |
| 2 | 0 | ∞ | 14 | 79 | **0** |
| 3 | 0 | 0 | ∞ | 56 | **0** |
| 4 | 15 | 54 | 0 | ∞ | **0** |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Город | 1 | 2 | 3 | 5 |
| 1 | ∞ | 0 | 19 | ∞ |
| 2 | 0 | ∞ | 14 | 79 |
| 3 | 0 | 0 | ∞ | 56 |
| 4 | 15 | 54 | 0 | ∞ |
|  | **0** | **0** | **0** | **56** |

**φ=22+1=23 φ=23+0=23**

**φ=23+103=126 φ=23+57=80**

Проведем редукцию матрицы:

|  |  |  |  |
| --- | --- | --- | --- |
| Город | 1 | 2 | 3 |
| 2 | 0 | ∞ | 14 |
| 3 | 0 | 0 | ∞ |
| 4 | 15 | 54 | 0 |

Т.к. обратного пути и возможных подциклов (петель) нет, то пропускаем шаг.

|  |  |  |  |
| --- | --- | --- | --- |
| Город | 1 | 2 | 3 |
| 2 | 0 | ∞ | 14 |
| 3 | 0 | 0 | ∞ |
| 4 | 15 | 54 | 0 |

Итерация 3.Найдем в каждой строке минимальное значение и произведем редукцию строк:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Город | 1 | 2 | 3 |  |
| 2 | 0 | ∞ | 14 | 0 |
| 3 | 0 | 0 | ∞ | 0 |
| 4 | 15 | 54 | 0 | 0 |

Теперь найдем минимальное значение в каждом столбце и произведем редукцию столбцов:

|  |  |  |  |
| --- | --- | --- | --- |
| Город | 1 | 2 | 3 |
| 2 | 0 | ∞ | 14 |
| 3 | 0 | 0 | ∞ |
| 4 | 15 | 54 | 0 |
|  | 0 | 0 | 0 |

**φ=22+1=23 φ=23+0=23 φ=23+0=23**

**φ=23+103=126 φ=23+57=80**

В получившейся матрице найдем нулевые клетки и вычислим для нулевых клеток оценки:

|  |  |  |  |
| --- | --- | --- | --- |
| Город | 1 | 2 | 3 |
| 2 | **0 (14)** | ∞ | 14 |
| 3 | **0 (0)** | **0 (54)** | ∞ |
| 4 | 15 | 54 | **0 (29)** |

Найдем среди вычисленных оценок нулевых клеток максимальную оценку: **0 (54)**.

Ставим знак '∞' вместо нулевого элемента с максимальной оценкой:

|  |  |  |  |
| --- | --- | --- | --- |
| Город | 1 | 2 | 3 |
| 2 | 0 | ∞ | 14 |
| 3 | 0 | ∞ | ∞ |
| 4 | 15 | 54 | 0 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Город | 1 | 2 | 3 |  |
| 2 | 0 | ∞ | 14 | 0 |
| 3 | 0 | ∞ | ∞ | 0 |
| 4 | 15 | 54 | 0 | 0 |

|  |  |  |  |
| --- | --- | --- | --- |
| Город | 1 | 2 | 3 |
| 2 | 0 | ∞ | 14 |
| 3 | 0 | ∞ | ∞ |
| 4 | 15 | 54 | 0 |
|  | 0 | 54 | 0 |

**φ=22+1=23 φ=23+0=23 φ=23+0=23**

**φ=23+103=126 φ=23+57=80 φ=23+54=87**

Проведем редукцию матрицы:

|  |  |  |
| --- | --- | --- |
| Город | 1 | 3 |
| 2 | 0 | 14 |
| 4 | 15 | 0 |

Чтобы исключить возврат в уже посещенный город, мы ставим символ '∞' на обратный путь:

|  |  |  |
| --- | --- | --- |
| Город | 1 | 3 |
| 2 | 0 | ∞ |
| 4 | 15 | 0 |

Путей, замыкающих маршрут раньше времени, нету, поэтому пропускаем шаг.

Итерация 4.Найдем в каждой строке минимальное значение и произведем редукцию строк:

|  |  |  |  |
| --- | --- | --- | --- |
| Город | 1 | 3 |  |
| 2 | **0** | ∞ | 0 | |
| 4 | **15** | **0** | 0 | |

Найдём минимальное значение в каждом столбце и произведем редукцию столбцов:

|  |  |  |
| --- | --- | --- |
| Город | 1 | 3 |
| 2 | **0** | ∞ |
| 4 | **15** | **0** |
|  | 0 | 0 |

**φ=22+1=23 φ=23+0=23 φ=23+0=23 φ=23+0=23**

**φ=23+103=126 φ=23+57=80 φ=23+54=87**

В получившейся матрице найдем нулевые клетки и вычислим для нулевых клеток оценки:

|  |  |  |
| --- | --- | --- |
| Город | 1 | 3 |
| 2 | **0 (15)** | ∞ |
| 4 | 15 | **0 (15)** |

Найдем среди вычисленных оценок нулевых клеток максимальную оценку: **0 (15)**. У нас их две равных.

**φ=22+1=23 φ=23+0=23 φ=23+0=23 φ=23+0=23**

**φ=23+103=126 φ=23+57=80 φ=23+54=87 φ=23+0=23**

**Первый вариант решения.**

Проведем редукцию матрицы:

|  |  |
| --- | --- |
| Город | 1 |
| 2 | 0 |

Ни обратного пути, ни возможных петель нет, поэтому пропускаем шаг.

Итерация 5 (1 вариант).Последний отрезок оптимального пути можно с легкостью определить без расчетов, просто посмотрев какая пара городов в цепи осталась незамкнутой. Это и будет искомый отрезок маршрута.

**Второй вариант решения.**

Проведем редукцию матрицы:

|  |  |
| --- | --- |
| Город | 3 |
| 4 | **0** |

Ни обратного пути, ни возможных петель нет, поэтому пропускаем шаг.

Итерация 5 (второй вариант).Последний отрезок оптимального пути можно с легкостью определить без расчетов, просто посмотрев какая пара городов в цепи осталась незамкнутой. Это и будет искомый отрезок маршрута.

Т.к. нам необходимо, чтоб начало и конец маршрута совпадали, и каждый город был задействован, то второй вариант решения не является правильным.

**Ответ**

Т. о., задача коммивояжера успешно решена, оптимальный (кратчайший) путь найден. Ему соответствует следующий маршрут: **1→5→4→3→2→1**. Общая длина найденного оптимального маршрута составляет **23** ед. дл.

**Задание 3.** Проверить полученное решение при помощи генератора перестановок и включить копию экрана с решением в отчет. (В программе отсчёт начинается с 0 и заканчивается 4).

