Белорусский государственный технологический университет

Кафедра Информационных Систем и Технологий

**Лабораторная работа №3**

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

Выполнила студентка 2 курса 6 группы

Пунько Алина Андреевна

Минск 2019

**Условие**

Найти оптимальный маршрут для коммивояжера, если известно, что кол-во городов равно 5, а расстояние между городами задается следующей матрицей d:

где *n* – номер варианта;

n=10;

Задачу следует решить с использованием метода ветвей и границ.

Полученное условие:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Город | 0 | 1 | 2 | 3 | 4 |
| 0 | INF | 20 | 31 | INF | 10 |
| 1 | 10 | INF | 25 | 58 | 74 |
| 2 | 12 | 30 | INF | 86 | 59 |
| 3 | 27 | 48 | 40 | INF | 30 |
| 5 | 83 | 79 | 52 | 23 | INF |

**Ход решения**

Имеем 5 городов, построим матрицу расстояний между городами:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Город | 0 | 1 | 2 | 3 | 4 |
| 0 | INF | 20 | 31 | INF | 10 |
| 1 | 10 | INF | 25 | 58 | 74 |
| 2 | 12 | 30 | INF | 86 | 59 |
| 3 | 27 | 48 | 40 | INF | 30 |
| 4 | 83 | 79 | 52 | 23 | INF |

Находим минимальное значение в каждой строке (di) и выписываем его в отдельный столбец:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Город | 0 | 1 | 2 | 3 | 4 |
| 0 | INF | 20 | 31 | INF | 10 | 10 |
| 1 | 10 | INF | 25 | 58 | 74 | 10 |
| 2 | 12 | 30 | INF | 86 | 59 | 12 |
| 3 | 27 | 48 | 40 | INF | 30 | 27 |
| 4 | 83 | 79 | 52 | 23 | INF | 23 |
| **82** |

Производим редукцию строк – из каждого элемента в строке вычитаем соответствующее значение найденного минимума (di).

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 0 | 1 | 2 | 3 | 4 |
| 0 | INF | 10 | 31 | INF | 0 | 10 |
| 1 | 0 | INF | 15 | 48 | 64 | 10 |
| 2 | 0 | 18 | INF | 74 | 47 | 12 |
| 3 | 0 | 21 | 13 | INF | 3 | 27 |
| 4 | 60 | 56 | 29 | 0 | INF | 23 |
| **82** |

Находим минимальные значения в каждом столбце (dj). Эти минимумы выписываем в отдельную строку.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 0 | 1 | | | 2 | 3 | | 4 |
| 0 | INF | 10 | | | 21 | INF | | 0 |
| 1 | 0 | INF | | | 15 | 48 | | 64 |
| 2 | 0 | 18 | | | INF | 74 | | 47 |
| 3 | 0 | 21 | | | 13 | INF | | 3 |
| 4 | 60 | 56 | | | 29 | 0 | | INF |
| 0 | | 10 | 13 | | | 0 | 0 | | 23 |

Вычитаем из каждого элемента матрицы соответствующее ему минимальные значения в каждом столбце dj.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 0 | 1 | | | 2 | 3 | | 4 |
| 0 | INF | 0 | | | 8 | INF | | 0 |
| 1 | 0 | INF | | | 2 | 48 | | 64 |
| 2 | 0 | 8 | | | INF | 74 | | 47 |
| 3 | 0 | 11 | | | 0 | INF | | 3 |
| 4 | 60 | 46 | | | 16 | 0 | | INF |
| 0 | | 10 | 13 | | | 0 | 0 | | 23 |

Тогда корневой вершиной будет

**f=23+82=105.**

Для каждой нулевой клетки получившейся преобразованной матрицы находим «оценку». Полученную оценку записываем рядом с нулем, в скобках.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 0 | 1 | 2 | 3 | 4 |
| 0 | INF | 0(8) | 8 | INF | 0(3) |
| 1 | 0(2) | INF | 2 | 48 | 64 |
| 2 | 0(8) | 8 | INF | 74 | 47 |
| 3 | 0(0) | 11 | 0(2) | INF | 3 |
| 4 | 60 | 46 | 16 | 0(64) | INF |

Выбираем нулевую клетку с наибольшей оценкой. Будем рассматривать дугу (4,3). Так как удаление дуги (4,3) позволяет получить саму большую константу приведения, т.е. увеличение нижней границы. Ту строку и тот столбец полностью вычеркиваем. В клетку, соответствующую обратному пути ставим еще один знак «INF» (т.к. мы уже не будем возвращаться обратно).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 0 | 1 | 2 | 4 |
| 0 | INF | 0(8) | 8 | 0(3) |
| 1 | 0(2) | INF | 2 | 64 |
| 2 | 0(8) | 8 | INF | 47 |
| 3 | 0(0) | 11 | 0(2) | INF |

Видим, что матрица уже является приведенной.

Значит снова для каждой нулевой клетки получившейся преобразованной матрицы находим «оценку». Полученную оценку записываем рядом с нулем, в скобках.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | 0 | 1 | 2 | 4 |
| 0 | INF | 0(8) | 8 | 0(47) |
| 1 | 0(2) | INF | 2 | 64 |
| 2 | 0(8) | 8 | INF | 47 |
| 3 | 0(0) | 11 | 0(2) | INF |

Далее будем рассматривать дугу (0.4). Так как удаление дуги (0.4) позволяет получить самую большую константу приведения, т.е. увеличение нижней границы. Для этого заменим вес дуги (0.4) на знак “INF.

Ту строку и тот столбец полностью вычеркиваем. В клетку, соответствующую обратному пути ставим еще один знак «INF» (т.к. мы уже не будем возвращаться обратно).

|  |  |  |  |
| --- | --- | --- | --- |
|  | 0 | 1 | 2 |
| 1 | 0 | INF | 2 |
| 2 | 0 | 8 | INF |
| 3 | 0 | 11 | 0 |

Приведём матрицу по столбцам, так как по строкам она уже приведена:

|  |  |  |  |
| --- | --- | --- | --- |
|  | 0 | 1 | 2 |
| 1 | 0 | INF | 2 |
| 2 | 0 | 8 | INF |
| 3 | 0 | 11 | 0 |
| 0 | 8 | 0 | 8 |

Проведем редукцию столбцов:

|  |  |  |  |
| --- | --- | --- | --- |
|  | 0 | 1 | 2 |
| 1 | 0 | INF | 2 |
| 2 | 0 | 0 | INF |
| 3 | 0 | 11 | 0 |

Cнова для каждой нулевой клетки получившейся преобразованной матрицы находим «оценку». Полученную оценку записываем рядом с нулем, в скобках.

|  |  |  |  |
| --- | --- | --- | --- |
|  | 0 | 1 | 2 |
| 1 | 0(2) | INF | 2 |
| 2 | 0(0) | 0(11) | INF |
| 3 | 0(0) | 11 | 0(2) |

Далее будем рассматривать дугу (2,1). Так как удаление дуги (2,1) позволяет получить самую большую константу приведения, ту строку и тот столбец, полностью вычеркиваем. В клетку, соответствующую обратному пути ставим еще один знак «INF» (т.к. мы уже не будем возвращаться обратно).

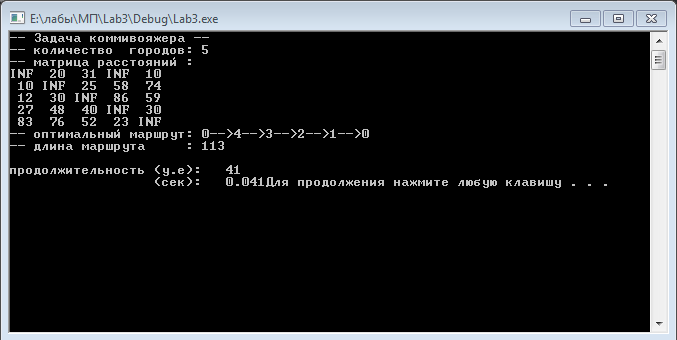
В клетку(2,1) соответствующую обратному пути ставим еще один знак «INF»

|  |  |  |
| --- | --- | --- |
|  | 0 | 2 |
| 1 | 0 | INF |
| 3 | 0 | 0 |

У нас остаются два маршрута (1,0) и (3,2)

**Решение: (0,4), (4,3), (3,2), (2,1), (1,0)**

**Длина оптимального маршрута: φ=113**



Риc. 1 Проверка решения про помощи генератора перестановок

Φ=105+0=105

Φ=105

φ=113+0=113

φ=105+8=113

R

R (5, 4) (1, 5)

R (5, 4) (1, 5) (3, 2)

R (5, 4)

φ=105+64=169

φ=113+0=113

φ=105+47=152

**R (5, 4) (1, 5)**

**R**

R (5, 4 ) (4,3) (3, 2) (2,1) (1, 5)

Вывод: выполнив данную работу, мы рассмотрели решение задачи коммивояжера методом ветвей и границ, убедились в его верности, проверив решение программным способом.