У багатьох виробників виникає задача вибору найкоротших маршрутів на мережі поставок виробленого продукту, що суттєво впливає на вартість транспортування. Найкоротший маршрут визначається вартістю транспортування продукту на окремих відрізках, які його утворюють.

Припустимо, що компанія-постачальник у місті N розробляє план поставок продукту зі свого складу у міські магазини «Фора» з мінімальними транспортними витратами. Визначимо мережу магазинів «Фора» і можливих транспортних маршрутів у вигляді неорієнтовного зв’язаного графу G, де кількість вузлів дорівнює кількості магазинів плюс ще один вузол, де розміщено склад. Також визначимо вартість транспортування продукту по всім можливим маршрутам, які з’єднують вузли графа.

1

2

3

4

5

6

7

8

9

0

122

152

7

8

4

18

8

11

7

8

17

10

5

8

11

21

10

3

Тоді, оптимальний план перевозок, у якого загальна сума транспортних витрат на поставку продуктів у всі магазини буде мінімальна, являє собою остовне дерево цього графа. Враховуючі властивості остовного дерева графа, …

Визначимо матрицю суміжності графа G наступним чином:

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
| 0 | 0 |  | 12 | 21 | 7 |  |  |  |  | 4 |
| 1 |  | 0 | 3 | 8 | 15 |  |  |  | 17 |  |
| 2 | 12 | 3 | 0 |  |  | 8 | 10 |  |  |  |
| 3 | 21 | 8 |  | 0 |  |  | 8 | 11 |  |  |
| 4 | 7 | 15 |  |  | 0 |  |  |  |  | 8 |
| 5 |  |  | 8 |  |  | 0 | 5 | 7 |  |  |
| 6 |  |  | 10 | 8 |  | 5 | 0 |  |  | 18 |
| 7 |  |  |  | 11 |  | 7 |  | 0 | 10 |  |
| 8 |  | 17 |  |  |  |  |  | 10 | 0 | 11 |
| 9 | 4 |  |  |  | 8 |  | 18 |  | 11 | 0 |

Сформуємо вхідні дані для програми new\_prima.c у файлі f19.txt, заповнюючи вартість відсутніх ребер нулями. Далі виконаємо програму з цими вхідними даними. В результаті отримаємо список ребер остовного дерева і оптимальну загальну суму транпортних витрат на доставку продуктів в магазини «Фора». На мал. жирними лініями показано остовне дерево на графі G, тобто маршрути доставки продуктів в магазини «Фора».

1

2

3

4

5

6

7

8

9

0

122

152

7

8

4

18

8

11

7

8

17

10

5

8

11

21

10

3

**Мін. остовне дерево**

ребро вартість

0 - 9 4

0 - 4 7

9 - 8 11

8 - 7 10

7 - 5 7

5 - 6 5

5 - 2 8

2 - 1 3

6 - 3 8

--------------------

Мін. сума = 63

Вартість змінюється у часі в залежності від різноманітних факторів, наприклад ремонту дорожнього покриття або транспортних розв’язок. Деякі міста, наприклад Сінгапур, автоматично змінюють вартість проїзду по міським магістралям в залежності від навантаження. Це визначає жорсткі вимоги до алгоритму стосовно часу, тобто забезпечення оперативної побудови нових транспортних планів.

Оцінка складності алгоритму побудови мінімального остовного дерева. Складність алгоритму – це кількісна оцінка ресурсів, таких як час і пам’ять, необхідні для його виконання

Оцінка часу. Алгоритм передбачає 3 вкладені цикли:

1. for (n = 0; n < m-1; n) – вибір ребра остовного дерева – (m-1) ітерацій
2. q = 0; while ( q < m ) – кількість ітерацій у цьому циклі становить n, оскільки він виконується для вузлів із множини vis, яка включає перелік вузлів, що включені в остовне дерево на етапі n. Враховуючі що кількість вузлів множини vis збільшується на 1 на кожному кроці n, кількість ітерацій в у перших двох вкладених циклах становить n(n+1)/2. <https://ru.wikipedia.org/wiki/Треугольное_число>
3. Кількість ітерацій k у третьому циклі s = 0; while ( s < m ) – вибір існуючих ребер, які з’єднують вузли поточного стану остовного дерева з сусідніми вузлами і вибір з них мінімального залежить від зв’язності графу. Тоді загальна кількість ітерацій трьох вкладених циклів становить n(n+1)k/2.

Таким чином складність алгоритму ПРИМА в даної реалізації становить:

**m(m+1)k/2 ~ m2k/2,**

де, m – кількість вузлів на графі G