|  |  |
| --- | --- |
| **Gerb-BMSTU_01** | **Министерство науки и высшего образования Российской Федерации**  Калужский филиал  федерального государственного бюджетного  образовательного учреждения высшего образования  ***«Московский государственный технический университет имени Н.Э. Баумана (национальный исследовательский университет)»***  ***(КФ МГТУ им. Н.Э. Баумана)*** |

**ФАКУЛЬТЕТ** ***М-КФ «Машиностроительный»***

**КАФЕДРА** \_***М10-КФ «Высшая математика и физика»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_***

**ЛАБОРАТОРНАЯ РАБОТА №7**

**«Минимальные каркасы»**

**ДИСЦИПЛИНА: «Дискретная математика»**

|  |  |  |
| --- | --- | --- |
| Выполнил: студент гр. ИУК4-32Б | | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ( Карельский М.К. )  (Подпись) |
| Проверил: доцент кафедры М10-КФ | | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ ( Булычев В.А. )  (Подпись) |
| Дата сдачи (защиты):  Результаты сдачи (защиты): | | |
|  | - Балльная оценка:  - Оценка: | |

Калуга, 2021

**Цель:** реализация алгоритмов для построения минимального каркаса (остовного дерева) взвешенного графа.

**Задание:**

1. Прочитайте из файла «data.txt» значения переменной N и весовую матрицу A размера для своего варианта.
2. С помощью алгоритма Краскала найдите минимальное остовное дерево заданного графа и его вес.
3. С помощью алгоритма Прима найдите минимальное остовное дерево заданного графа и его вес.
4. Нарисуйте исходный граф и найденные каркасы (если они разные, сделайте два рисунка). Этот пункт задания можно сделать как с помощью компьютера, так и вручную.

**Вариант 10**

**Листинг:**

import math

n = int

A = []

with open('data.txt') as f:

n = f.readline()

n = int(n)

print(' ', ''.join('{:>3}'.format(i + 1) for i in range(n)))

print()

for line in f:

numbers = line.split()

A.append([int(numbers[i]) for i in range(n)])

print('{:>2}'.format(len(A)), ''.join('{:>3}'.format(i) for i in numbers))

print()

weight = 0

frame = []

marks = [i for i in range(n)]

edges = []

for i in range(n):

for j in range(i + 1, n):

edges.append([A[i][j], i, j])

edges.sort()

edgesInFrame = 0

for edge in edges:

if marks[edge[1]] != marks[edge[2]]:

weight += edge[0]

frame.append([edge[1] + 1, edge[2] + 1])

edgesInFrame += 1

markToChange = marks[edge[2]]

for i in range(n):

if marks[i] == markToChange:

marks[i] = marks[edge[1]]

if edgesInFrame == n - 1:

break

print('Алгоритм Краскала:', ' '.join(f'{i[0]}-{i[1]}' for i in frame))

print('Вес =', weight)

weight = 0

frame = []

marks = [i for i in range(n)]

edgesInFrame = 0

for i in range(n - 1):

minEdge = [math.inf]

side = int

for edge in edges:

if edge[0] < minEdge[0]:

if marks[edge[1]] == 0 and marks[edge[2]] != 0:

minEdge = edge

side = 2

elif marks[edge[2]] == 0 and marks[edge[1]] != 0:

minEdge = edge

side = 1

weight += minEdge[0]

frame.append([minEdge[1] + 1, minEdge[2] + 1])

markToChange = marks[minEdge[side]]

for j in range(n):

if marks[j] == markToChange:

marks[j] = 0

print('Алгоритм Прима:', ' '.join(f'{i[0]}-{i[1]}' for i in frame))

print('Вес =', weight)

**Результат:**

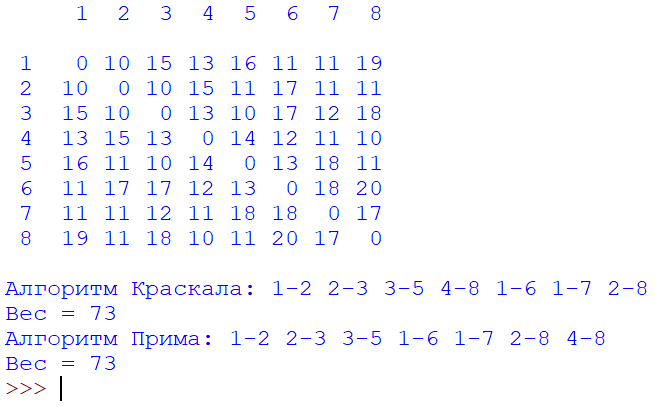


Рис. 1. Результат работы

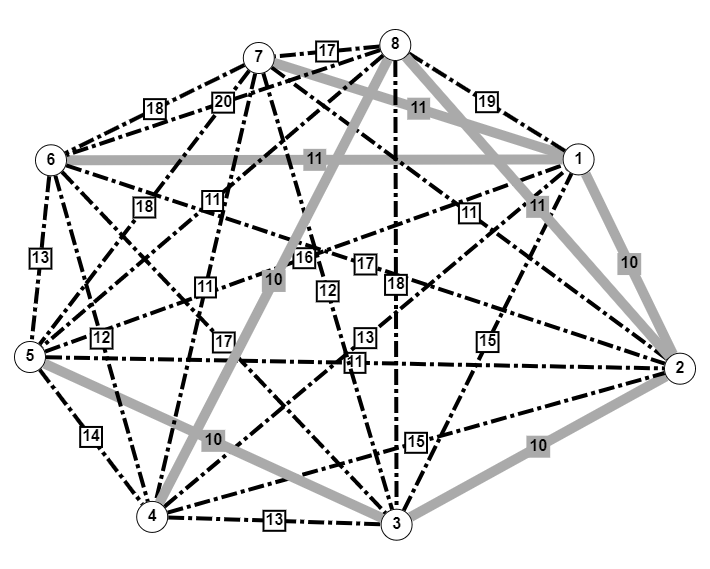


Рис. 2. Каркас

**Вывод:** в ходе выполнения лабораторной работы составлены программы, реализующие алгоритмы Краскала и Прима для построения минимального остовного дерева. Изучены возможности пакета GraphTheory.

**Контрольные вопросы:**

1. Что называется каркасом (остовным деревом) неориентированного связного графа?

Каркасом (остовом) называется подграф, который содержит вершины графа и является деревом.

1. Сформулируйте задачу нахождения минимального каркаса.

Найти каркас взвешенного дерева с минимальной суммой весов.

1. В чем, на ваш взгляд, главное отличие алгоритмов Краскала и Прима?

Алгоритм Краскала начинает с ребра, а Прима – с узла.

1. Какова сложность алгоритма Краскала? алгоритма Прима?

Алгоритм Краскала – О(n log n)

Алгоритм Прима – О(n2)