МОСКОВСКИЙ АВИАЦИОННЫЙ ИНСТИТУТ
(НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)
ФАКУЛЬТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И ПРИКЛАДНОЙ МАТЕМАТИКИ
КАФЕДРА МАТЕМАТИЧЕСКОЙ КИБЕРНЕТИКИ

КУРСОВАЯ РАБОТА

ПЕРЕЧИСЛЕНИЕ ПУТЕЙ ОРИЕНТИРОВАННОГО ГРАФА МЕТОДОМ ЛАТИНСКОЙ КОМПОЗИЦИИ

Студент: Березнев Н.В.

Группа 8О-103Б

Преподаватель: Смерчинская С.О.

Оценка:

Дата:

Задание

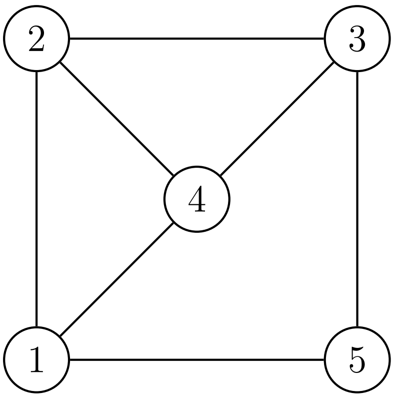
Вариант 4

1. Определить для орграфа, заданного матрицей смежности:

$$A = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix}$$

а) матрицу односторонней связности;
б) матрицу сильной связности;
в) компоненты сильной связности;
г) матрицу контуров;
д) изображение графа и компонент сильной связности;

2. Используя алгоритм Терри, определить замкнутый маршрут, проходящий ровно по два раза (по одному в каждом направлении) через каждое ребро графа.



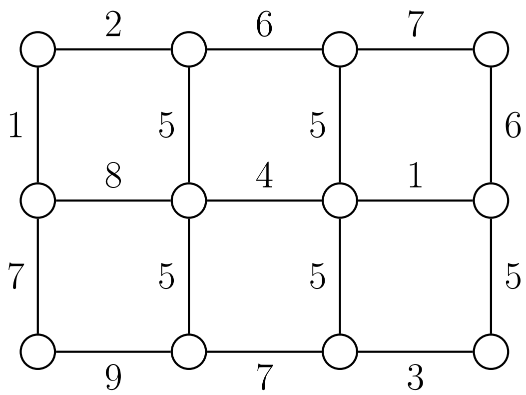
3. Используя алгоритм "фронта волны", найти все минимальные пути из первой вершины в последнюю орграфа, заданного матрицей смежности.

$$A = \begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 1 & 0 \end{pmatrix}$$

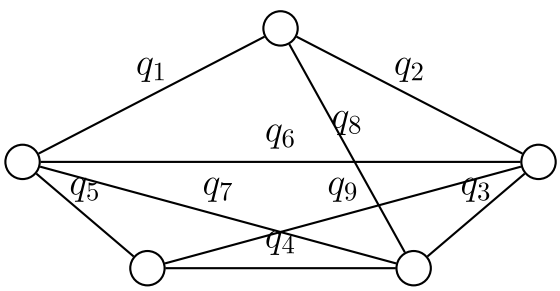
4. Используя алгоритм Форда, найти минимальные пути из первой вершины во все достижимые вершины в нагруженном графе, заданном матрицей длин дуг.

$$C = \begin{pmatrix} \infty & 3 & 5 & \infty & 6 & \infty & \infty & \infty \\ 2 & \infty & 1 & 4 & \infty & \infty & \infty & \infty \\ 3 & \infty & \infty & 4 & 2 & \infty & \infty & \infty \\ \infty & \infty & \infty & \infty & \infty & 3 & 5 & \infty \\ 4 & \infty & \infty & \infty & \infty & 6 & \infty & 7 \\ \infty & \infty & \infty & \infty & \infty & \infty & 3 & 2 \\ 6 & \infty & \infty & \infty & \infty & \infty & \infty & 1 \\ 8 & \infty & \infty & \infty & 11 & \infty & \infty & \infty \end{pmatrix}$$

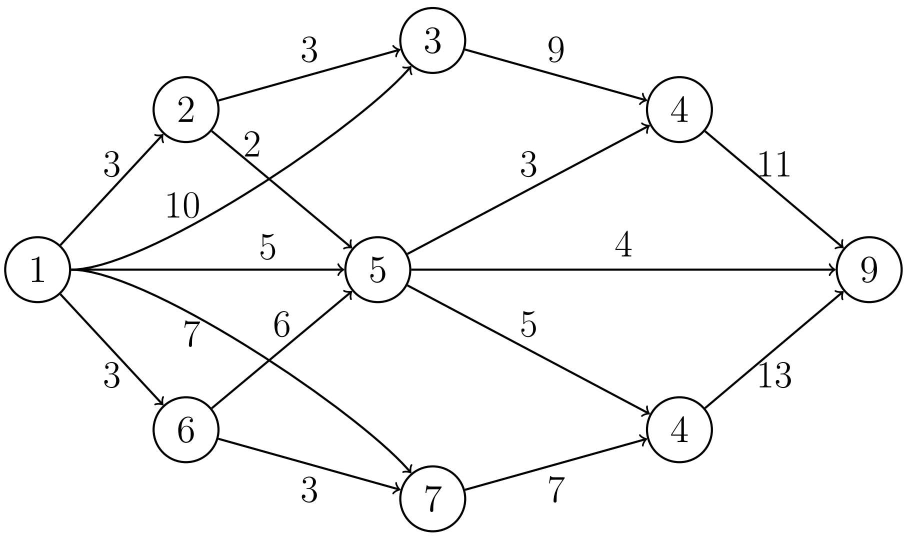
5. Найти остовное дерево с минимальной суммой длин входящих в него ребер.



6. Пусть каждому ребру неориентированного графа соответствует некоторый элемент электрической цепи. Составить линейно независимые системы уравнений Кирхгофа для токов и напряжений. Пусть первому и пятому ребру соответствуют источники тока с ЭДС $E1$ и $E2$, а остальные элементы являются сопротивлениями. Используя закон Ома, и, предполагая внутренние сопротивления источников тока равными нулю, получить систему уравнений для токов.



7. Построить максимальный поток по транспортной сети.



8. Перечисление путей ориентированного графа методом латинской композиции.

1. Изучить алгоритм.
2. Составить программу алгоритма.
3. Отладить тестовые примеры.
4. Провести оценку сложности алгоритма.
5. Составить прикладную задачу, для решения которой используется данный алгоритм.

Задание №1

а) **Способ №1**

$$A = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix}$$

$$A^2 = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix} * \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix} = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 \end{pmatrix}$$

$$A^3 = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 \end{pmatrix} * \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 \end{pmatrix}$$

$$T = E \vee A \vee A^2 \vee A^3 = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{pmatrix}$$

Способ №2

k = 0

$$T^{(0)} = E \vee A = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \vee \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \end{pmatrix}$$

k = 1,   k - 1 = 0

$$T^{(1)} = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \end{pmatrix} \vee \begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{pmatrix}$$

.....

Очевидно, что $T^{(4)} = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{pmatrix}$, значит $T = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{pmatrix}$

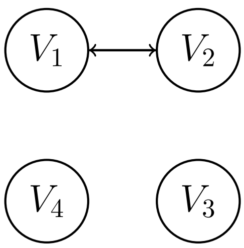
б) $\overline{S} = T \& T^T = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{pmatrix} \& \begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \end{pmatrix} = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$

$\overline{S} = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$ - матрица сильной связности

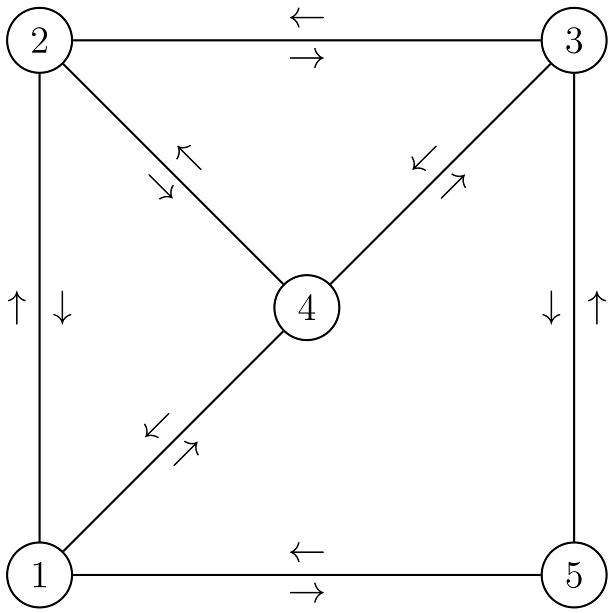
в) $\overline{S} = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$ => $\overline{S_1} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$ => $\overline{S_2} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$

г) $K = \overline{S} \& A = \begin{pmatrix} 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \& \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$

д)



Задание №2



$$1 \to 2 \to 3 \to 5 \to 1 \to 4 \to 3 \to 2 \to 4 \to 1 \to 5 \to 3 \to 4 \to 2 \to 1$$

Задание №3

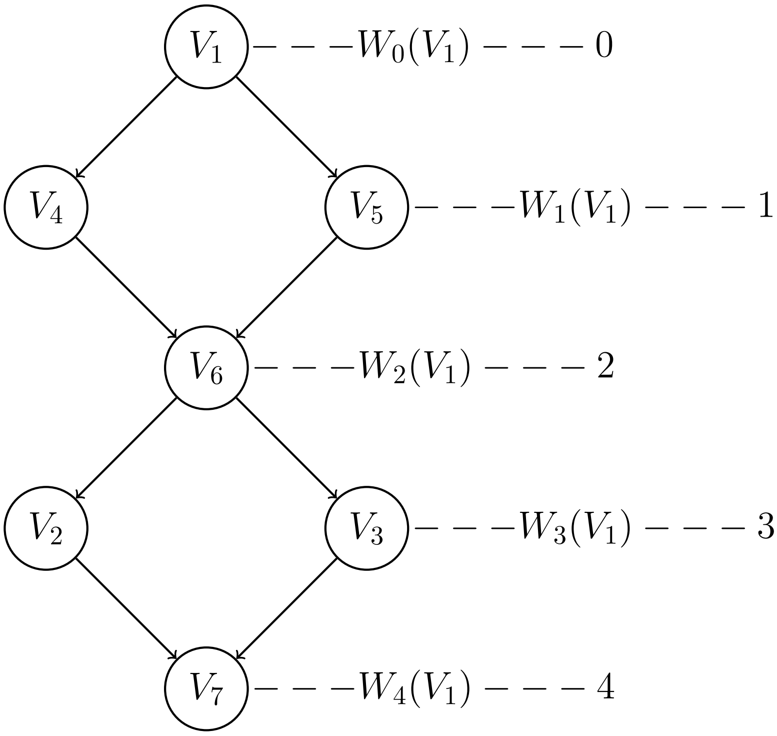
$W_0(v_1) = \{v_1\}$

$\Gamma W_0(v_1) = \{v_4, v_5\}$

$\Gamma W_1(v_1) = \{v_6\}$

$\Gamma W_2(v_1) = \{v_2, v_3\}$

$\Gamma W_3(v_1) = \{v_7\}$



Найдем промежуточные вершины кратчайших путей:

1) v_7

2) $w_3(v_1) \cap \Gamma^{-1} v_8 = \{v_2, v_3\} \cap \{v_2, v_3\} = \{v_2, v_3\}$

3.1) $w_2(v_1) \cap \Gamma^{-1} v_2 = \{v_6\} \cap \{v_3, v_6, v_7\} = \{v_6\}$

3.2) $w_2(v_1) \cap \Gamma^{-1} v_3 = \{v_6\} \cap \{v_2, v_6\} = \{v_6\}$

4.1) $w_1(v_1) \cap \Gamma^{-1} v_6 = \{v_4, v_5\} \cap \{v_2, v_4, v_5, v_7\} = \{v_4, v_5\}$

4.2) $w_1(v_1) \cap \Gamma^{-1} v_6 = \{v_4, v_5\} \cap \{v_2, v_4, v_5, v_7\} = \{v_4, v_5\}$

5.1.1) $w_0(v_1) \cap \Gamma^{-1} v_4 = \{v_1\} \cap \{v_1, v_2, v_3, v_5, v_6\} = \{v_1\}$

5.1.2) $w_0(v_1) \cap \Gamma^{-1} v_5 = \{v_1\} \cap \{v_1, v_3, v_4, v_7\} = \{v_1\}$

5.2.1) $w_0(v_1) \cap \Gamma^{-1} v_4 = \{v_1\} \cap \{v_1, v_2, v_3, v_5, v_6\} = \{v_1\}$

5.2.2) $w_0(v_1) \cap \Gamma^{-1} v_5 = \{v_1\} \cap \{v_1, v_3, v_4, v_7\} = \{v_1\}$

Кратчайших путей 4:

1) $v_1 - v_4 - v_6 - v_2 - v_7$

2) $v_1 - v_4 - v_6 - v_3 - v_7$

3) $v_1 - v_5 - v_6 - v_2 - v_7$

4) $v_1 - v_5 - v_6 - v_3 - v_7$

Задание №4

$$C = \begin{pmatrix} \infty & 3 & 5 & \infty & 6 & \infty & \infty & \infty \\ 2 & \infty & 1 & 4 & \infty & \infty & \infty & \infty \\ 3 & \infty & \infty & 4 & 2 & \infty & \infty & \infty \\ \infty & \infty & \infty & \infty & \infty & 3 & 5 & \infty \\ 4 & \infty & \infty & \infty & \infty & 6 & \infty & 7 \\ \infty & \infty & \infty & \infty & \infty & \infty & 3 & 2 \\ 6 & \infty & \infty & \infty & \infty & \infty & \infty & 1 \\ 8 & \infty & \infty & \infty & 11 & \infty & \infty & \infty \end{pmatrix}$$

Составим таблицу итераций:

	V_1	V_2	V_3	V_4	V_5	V_6	V_7	V_8	$\lambda_i^{(0)}$	$\lambda_i^{(1)}$	$\lambda_i^{(2)}$	$\lambda_i^{(3)}$	$\lambda_i^{(4)}$	$\lambda_i^{(5)}$	$\lambda_i^{(6)}$	$\lambda_i^{(7)}$
V_1	∞	3	5	∞	6	∞	∞	∞	0	0	0	0	0	0	0	0
V_2	2	∞	1	4	∞	∞	∞	∞	∞	3	3	3	3	3	3	3
V_3	3	∞	∞	4	2	∞	∞	∞	∞	5	4	4	4	4	4	4
V_4	∞	∞	∞	∞	∞	3	5	∞	∞	∞	7	7	7	7	7	7
V_5	4	∞	∞	∞	∞	6	∞	7	∞	6	6	6	6	6	6	6
V_6	∞	∞	∞	∞	∞	∞	3	2	∞	∞	12	10	10	10	10	10
V_7	6	∞	∞	∞	∞	∞	∞	1	∞	∞	∞	12	12	12	12	12
V_8	8	∞	∞	∞	11	∞	∞	∞	∞	∞	13	13	12	12	12	12

Найдем вершины, входящие в минимальные пути из v_1 во все остальные вершины графа.

1. Минимальный путь из v_1 в v_2: v_1 - v_2, его длина равна 3.
   - $\lambda_1^{(0)} + C_{12} = 0 + 3 = 3 = \lambda_2^{(1)}$
2. Минимальный путь из v_1 в v_3: v_1 - v_2 - v_3, его длина равна 4.
   - $\lambda_1^{(0)} + C_{12} = 0 + 3 = 3 = \lambda_2^{(1)}$
   - $\lambda_2^{(0)} + C_{23} = 3 + 1 = 4 = \lambda_3^{(2)}$
3. Минимальный путь из v_1 в v_4: v_1 - v_2 - v_4, его длина равна 7.
   - $\lambda_1^{(0)} + C_{12} = 0 + 3 = 3 = \lambda_2^{(1)}$
   - $\lambda_2^{(0)} + C_{24} = 0 + 7 = 7 = \lambda_4^{(2)}$
4. Минимальный путь из v_1 в v_5: v_1 - v_5, его длина равна 6.
   - $\lambda_1^{(0)} + C_{15} = 0 + 6 = 6 = \lambda_5^{(1)}$
5. Минимальный путь из v_1 в v_6: v_1 - v_2 - v_4 - v_6, его длина равна 10.
   - $\lambda_1^{(0)} + C_{12} = 0 + 3 = 3 = \lambda_2^{(1)}$
   - $\lambda_2^{(1)} + C_{24} = 3 + 4 = 7 = \lambda_4^{(2)}$
   - $\lambda_4^{(2)} + C_{46} = 7 + 3 = 10 = \lambda_6^{(3)}$

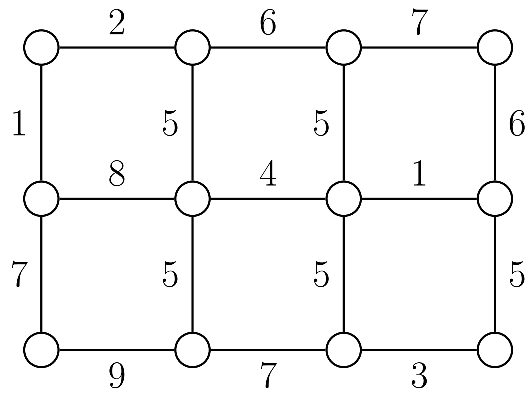
6. Минимальный путь из v_1 в v_7: v_1 - v_2 - v_4 - v_7, его длина равна 12.

- $\lambda_1^{(0)} + C_{12} = 0 + 3 = 3 = \lambda_2^{(1)}$
- $\lambda_2^{(1)} + C_{24} = 3 + 4 = 7 = \lambda_6^{(2)}$
- $\lambda_4^{(2)} + C_{47} = 7 + 5 = 12 = \lambda_7^{(3)}$

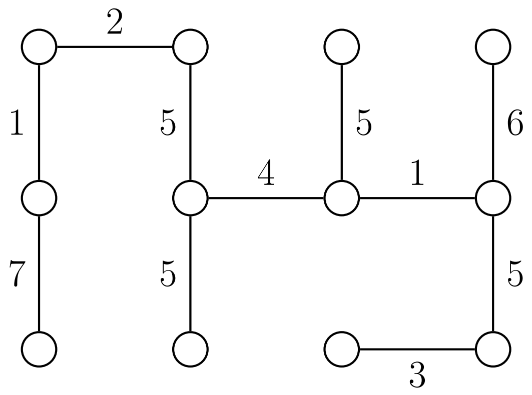
7. Минимальный путь из v_1 в v_8: v_1 - v_2 - v_4 - v_6 - v_8, его длина равна 12.

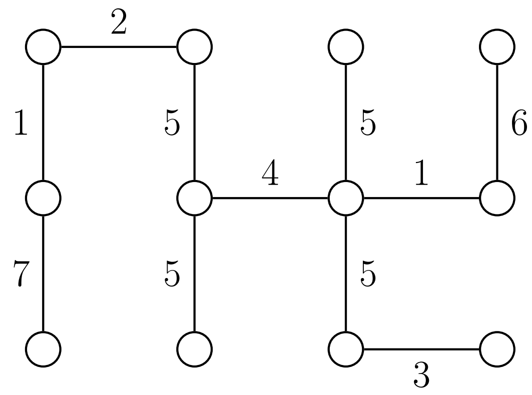
- $\lambda_1^{(0)} + C_{12} = 0 + 3 = 3 = \lambda_2^{(1)}$
- $\lambda_2^{(1)} + C_{24} = 3 + 4 = 7 = \lambda_6^{(2)}$
- $\lambda_4^{(2)} + C_{46} = 7 + 3 = \lambda_6^{(3)}$
- $\lambda_4^{(3)} + C_{68} = 10 + 2 = 12 = \lambda_8^{(4)}$

Задание №5

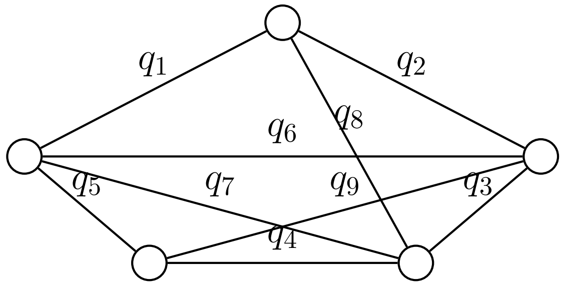


Возможные остовные деревья с минимальной суммой длин ребер, равной 44:

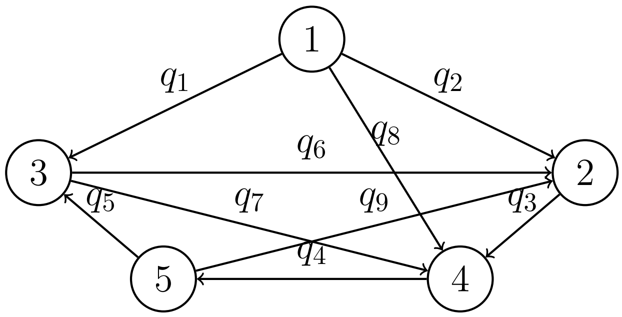




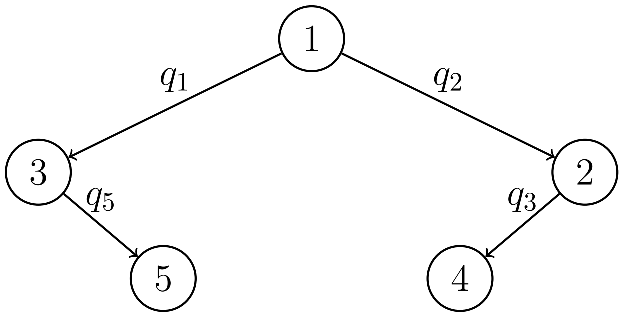
Задание №6



1. Зададим произвольную ориентацию



2. Построим произвольное остовное дерево D



3. Найдем базис циклов и соответствующие вектор-циклы

$$(D+q_8) : \mu_1 : v_1 - v_3 - v_2 - v_1 \Rightarrow C(\mu_1) = (0, -1, -1, 0, 0, 0, 0, 1, 0)$$

$$(D+q_4) : \mu_2 : v_3 - v_4 - v_5 - v_1 - v_2 - v_3 \Rightarrow C(\mu_2) = (1, 1, 1, 1, 1, 0, 0, 0, 0)$$

$$(D+q_6) : \mu_3 : v_5 - v_2 - v_1 - v_5 \Rightarrow C(\mu_3) = (-1, -1, 0, 0, 0, 1, 0, 0, 0)$$

$$(D+q_7) : \mu_4 : v_3 - v_5 - v_1 - v_2 - v_3(\mu_4) = (1, 1, 1, 0, 0, 0, 1, 0, 0)$$

$$(D+q_9) : \mu_5 : v_2 - v_4 - v_5 - v_1 - v_2(\mu_5) = (1, 1, 0, 0, 1, 0, 0, 0, 1)$$

4. Составим цикломатическую матрицу

$$C = \begin{pmatrix} 0 & -1 & -1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ -1 & -1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \end{pmatrix}$$

5. Запишем закон Кирхгова для напряжений

$$\begin{pmatrix} 0 & -1 & -1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ -1 & -1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \\ u_7 \\ u_8 \\ u_9 \end{pmatrix} = 0$$

$$\begin{cases} u_8 - u_2 - u_3 = 0 \\ u_1 + u_2 + u_3 + u_4 + u_5 = 0 \\ u_6 - u_1 - u_2 = 0 \\ u_1 + u_2 + u_3 + u_7 = 0 \\ u_1 + u_2 + u_5 + u_9 = 0 \end{cases}$$

$$\begin{cases} u_8 = u_2 + u_3 \\ u_1 = -u_2 - u_3 - u_4 - u_5 \\ u_6 = u_1 + u_2 \\ u_1 = -u_2 - u_3 - u_7 \\ u_1 = -u_2 - u_5 - u_9 \end{cases}$$

6,7. Выпишем закон и уравнения Кирхгова для токов

Найдем матрицу инцидентности

	q_1	q_2	q_3	q_4	q_5	q_6	q_7	q_8	q_9
v_1	1	-1	0	0	0	0	0	-1	0
v_2	0	1	-1	0	0	1	0	0	-1
v_3	0	0	1	-1	0	0	-1	1	0
v_4	0	0	0	1	-1	0	0	0	1
v_5	-1	0	0	0	1	-1	1	0	0

$$\begin{pmatrix} 1 & -1 & 0 & 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 1 & -1 & 0 & 0 & 1 & 0 & 0 & -1 \\ 0 & 0 & 1 & -1 & 0 & 0 & -1 & 1 & 0 \\ 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 1 \\ -1 & 0 & 0 & 0 & 1 & -1 & 1 & 0 & 0 \end{pmatrix} \begin{pmatrix} I_1 \\ I_2 \\ I_3 \\ I_4 \\ I_5 \\ I_6 \\ I_7 \\ I_8 \\ I_9 \end{pmatrix} = 0$$

$$\begin{cases} I_1 - I_2 - I_8 = 0 \\ I_2 - I_3 + I_6 - I_9 = 0 \\ I_3 - I_4 + I_8 - I_7 = 0 \\ I_4 - I_5 + I_9 = 0 \\ I_5 - I_1 - I_6 + I_7 = 0 \end{cases} \qquad \begin{cases} I_1 = I_2 + I_8 \\ I_3 = I_4 + I_7 - I_8 \\ I_4 = I_5 - I_9 \\ I_5 = I_1 + I_6 - I_7 \end{cases}$$

8. Подставим закон Ома

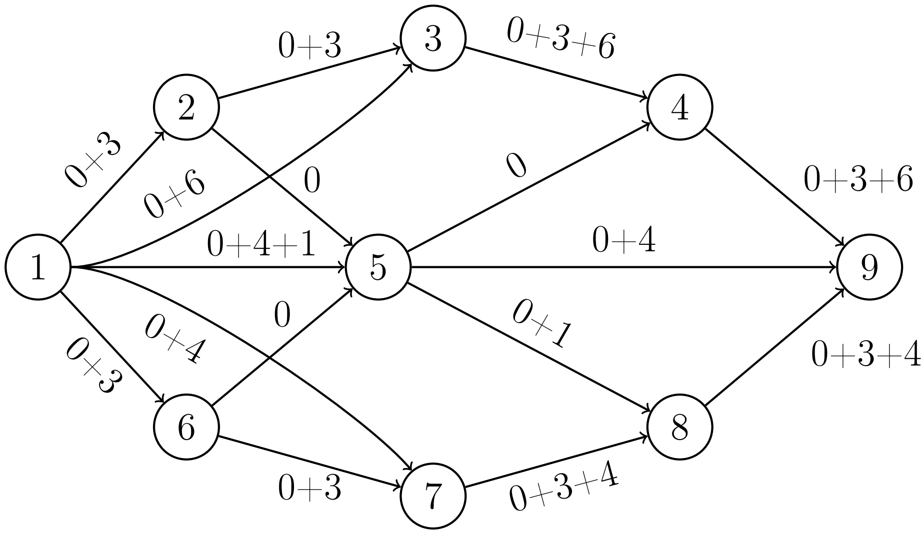
$$\begin{cases} 0 = -I_8R_8 + I_2R_2 + I_3R_3 \\ E_1 + E_2 = -I_2R_2 - I_3R_3 - I_4R_4 \\ E_1 = I_6R_6 - I_2R_2 \\ E_1 = -I_2R_2 - I_3R_3 - I_7R_7 \\ E_1 + E_2 = -I_2R_2 - I_9R_9 \end{cases}$$

9. Совместная система имеет вид

$$\begin{cases} I_1 = I_2 + I_8 \\ I_3 = I_4 + I_7 - I_8 \\ I_4 = I_5 - I_9 \\ I_5 = I_1 + I_6 - I_7 \\ 0 = -I_8R_8 + I_2R_2 + I_3R_3 \\ E_1 + E_2 = -I_2R_2 - I_3R_3 - I_4R_4 \\ E_1 = I_6R_6 - I_2R_2 \\ E_1 = -I_2R_2 - I_3R_3 - I_7R_7 \\ E_1 + E_2 = -I_2R_2 - I_9R_9 \end{cases}$$

9 уравнений и 9 неизвестных: $(I_1, I_2, I_3, I_4, I_5, I_6, I_7, I_8, I_9)$ ЭДС $E_1 E_2$ и сопротивления $R_2, R_3, R_4, R_5, R_6, R_7, R_8, R_9$ известны

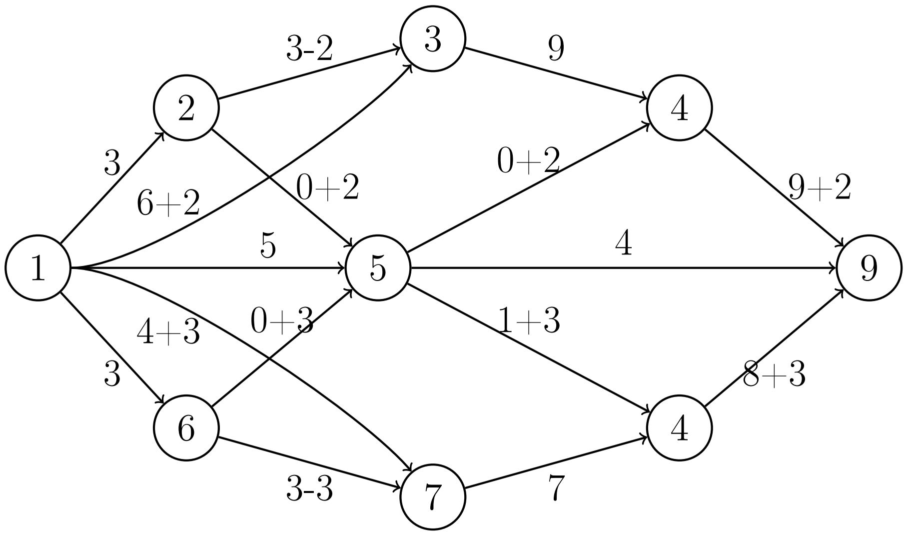
Задание №7



Полный поток:

1. $v_1 - v_2 - v_3 - v_4 - v_9$
   - $\min\{3, 3, 9, 11\} = 3$
2. $v_1 - v_6 - v_7 - v_8 - v_9$
   - $\min\{3, 3, 7, 13\} = 3$
3. $v_1 - v_5 - v_9$
   - $\min\{5, 4\} = 4$
4. $v_1 - v_3 - v_4 - v_9$
   - $\min\{10, 9 - 3, 11 - 3\} = 6$
5. $v_1 - v_7 - v_8 - v_9$
   - $\min\{7, 7 - 3, 13 - 3\} = 4$
6. $v_1 - v_5 - v_8 - v_9$
   - $\min\{5 - 4, 5, 13 - 7\} = 1$

Величина полного потока $\Phi_{\text{пол.}} = 3 + 3 + 4 + 6 + 4 + 1 = 21$



Максимальный поток:

1. $v_1 - v_3 - v_2 - v_5 - v_4 - v_9$
   - $\Delta_1 = \min\{10-6, 3, 2, 3, 11-9\} = 2$
2. $v_1 - v_7 - v_6 - v_5 - v_8 - v_9$
   - $\Delta_2 = \min\{7-4, 3, 6, 5, 13-8\} = 3$

Величина максимального потока $\Phi_{\text{макс.}} = 11 + 4 + 11 = 26$

Задание №8

Перечисление путей ориентированного графа методом латинской композиции

1. Основные понятия и определения

Определение 1. Тензор - матрица, элементами которой могут быть переменные, векторы, матрицы, символы, текстовые строки или другие тензоры.

Определение 2. Метод латинской композиции - матричный(тензорный) способ перечисления путей в графе и орграфе. Способ основывается на построении матрицы с обозначением путей, которые идут от i к j вершине графа (i,j – строка и столбец матрицы соответственно), а затем возведение этой матрицы в степень до тех пор, пока матрица не станет нулевой. Каждая степень исходной матрицы будет содержать пути определённой длин - таким образом будут перечислены все пути в орграфе.

2. Описание алгоритма

В данной работе используется адаптированный для задачи раскраски вершин графа алгоритм поиска в глубину:

1. В интерактивном окне указывается число вершин графа.
2. После этого в том же окне отмечаются вершины, между которыми есть связь.
3. После указания желаемого количества связей между вершинами в графе программа автоматически заполняет матрицу, после чего возводит ее в (n-2) степень.
4. В результате возведения матрицу в степень m, мы будем получать простые пути длины m. Возведение матрицы происходит до тех пор, пока она не станет нулевой или число возведений в степень не достигнет (n-2) раз.
5. После того, как в результате возведения матрицы в степень получается нулевая матрица, программа выводит полученные графы.
6. После вывода всех графов с окрашенными путями, можно продолжить пользоваться программой, указав новое число вершин в графе или изменив связи между вершинами.

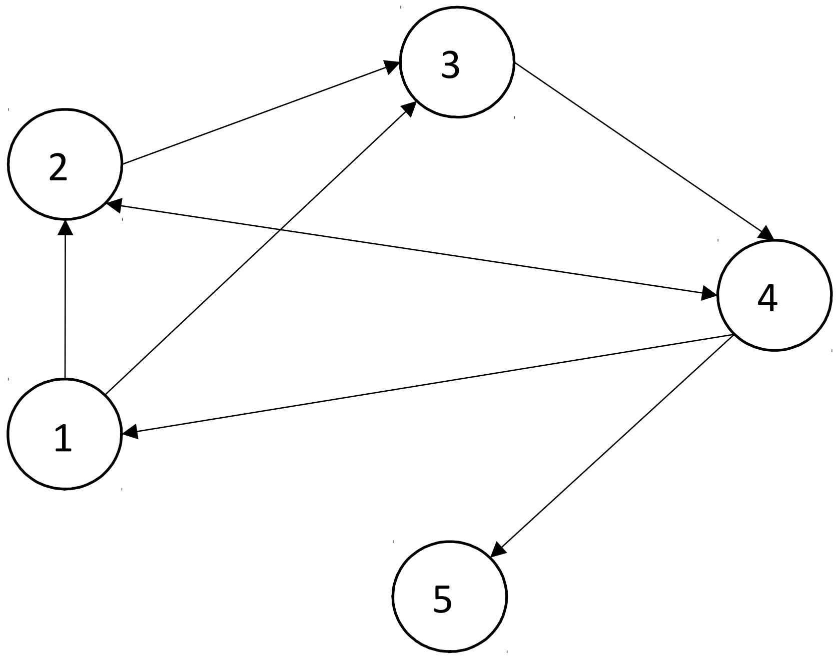
3. Блок-схема

5. Вычисление сложности алгоритма

Поскольку в программе осуществляется только умножение матриц (n-2) раз, то сложность данного алгоритма равна (n - 2) * $O(n^3)$.

6. Тестовый пример

Для примера был взят орграф с 5 вершинами, содержащий 8 рёбер:



1. При вводе данных создаётся матрица и вводится информация о рёбрах графа:

	V1	V2	V3	V4	V5
V1	[0]	[1,2]	[1,3]	[0]	[0]
	[0]	[0]	[2,3]	[2,4]	[0]
V3	[0]	[0]	[0]	[3,4]	[0]
V4	[4,1]	[4,2]	[0]	[0]	[4,5]
V5	[0]	[0]	[0]	[0]	[0]

2. Исходная матрица умножается на саму себя, т.е. возводится в квадрат:

[0]	[1,2]	[1,3]	[0]	[0]
[0]	[0]	[2,3]	[2,4]	[0]
[0]	[0]	[0]	[3,4]	[0]
[4,1]	[4,2]	[0]	[0]	[4,5]
[0]	[0]	[0]	[0]	[0]

*

[0]	[1,2]	[1,3]	[0]	[0]
[0]	[0]	[2,3]	[2,4]	[0]
[0]	[0]	[0]	[3,4]	[0]
[4,1]	[4,2]	[0]	[0]	[4,5]
[0]	[0]	[0]	[0]	[0]

=

=

[0]	[0]	[1,2,3]	[1,2,4] [1,3,4]	[0]
[2,4,1]	[0]	[0]	[2,3,4]	[2,4,5]
[3,4,1]	[3,4,2]	[0]	[0]	[3,4,5]
[0]	[4,1,2]	[4,1,3] [4,2,3]	[0]	[0]
[0]	[0]	[0]	[0]	[0]

3. Полученные пути выводим на экран. Далее полученную матрицу домножаем на начальную.

[0]	[0]	[1,2,3]	[1,2,4] [1,3,4]	[0]
[2,4,1]	[0]	[0]	[2,3,4]	[2,4,5]
[3,4,1]	[3,4,2]	[0]	[0]	[3,4,5]
[0]	[4,1,2]	[4,1,3] [4,2,3]	[0]	[0]
[0]	[0]	[0]	[0]	[0]

*

[0]	[1,2]	[1,3]	[0]	[0]
[0]	[0]	[2,3]	[2,4]	[0]
[0]	[0]	[0]	[3,4]	[0]
[4,1]	[4,2]	[0]	[0]	[4,5]
[0]	[0]	[0]	[0]	[0]

=

[0]	[1,3,4,2]	[0]	[1,2,3,4]	[1,2,4,5] [1,3,4,5]
[2,3,4,1]	[0]	[2,4,1,3]	[0]	[2,3,4,5]
[0]	[3,4,1,2]	[0]	[0]	[0]
[0]	[0]	[4,1,2,3]	[0]	[0]
[0]	[0]	[0]	[0]	[0]

4. Аналогично 3 пункту выводим пути путём считывания матрицы и ещё раз домножаем полученную матрицу на начальную

[0]	[1,3,4,2]	[0]	[1,2,3,4]	[1,2,4,5] [1,3,4,5]
[2,3,4,1]	[0]	[2,4,1,3]	[0]	[2,3,4,5]
[0]	[3,4,1,2]	[0]	[0]	[0]
[0]	[0]	[4,1,2,3]	[0]	[0]
[0]	[0]	[0]	[0]	[0]

*

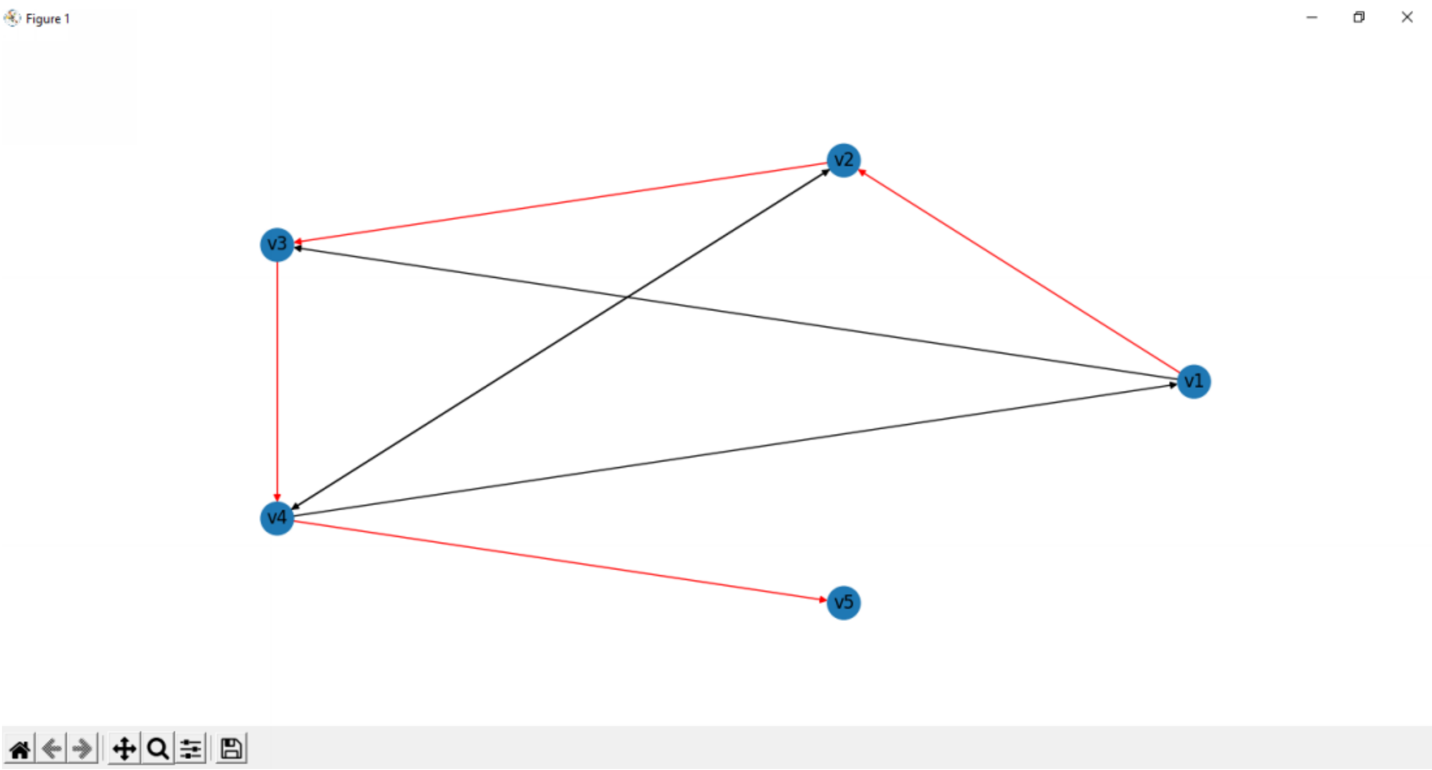
[0]	[1,2]	[1,3]	[0]	[0]
[0]	[0]	[2,3]	[2,4]	[0]
[0]	[0]	[0]	[3,4]	[0]
[4,1]	[4,2]	[0]	[0]	[4,5]
[0]	[0]	[0]	[0]	[0]

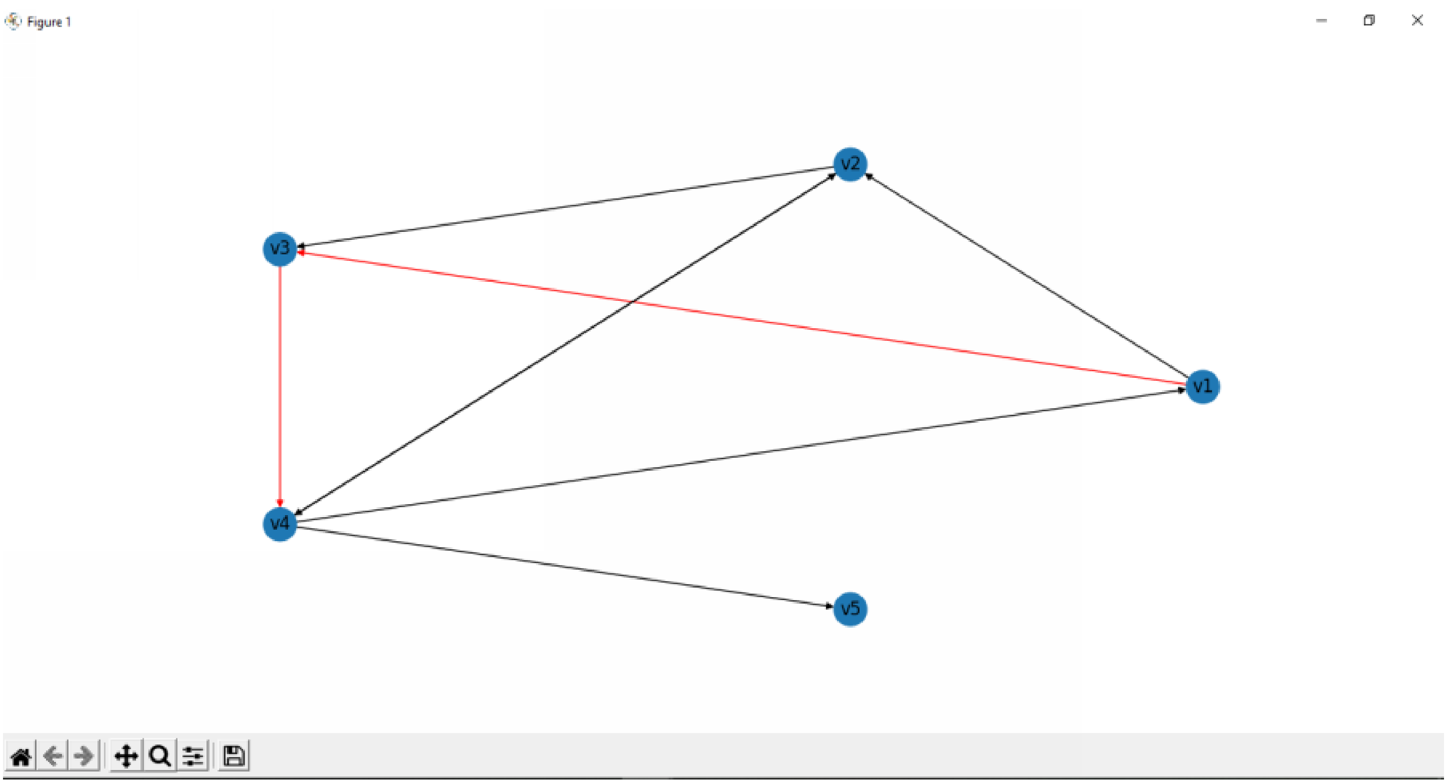
=

[0]	[0]	[0]	[0]	[1,2,3,4,5]
[0]	[0]	[0]	[0]	[0]
[0]	[0]	[0]	[0]	[0]
[0]	[0]	[0]	[0]	[0]
[0]	[0]	[0]	[0]	[0]

5. Был получен единственный простой путь длины 5, поэтому последующие матрицы будут нулевые, выводим путь путём считывания матрицы и завершаем алгоритм.

7. Скриншот программы для данного примера





8. Прикладная задача

Если дополнить орграф информацией о длине путей, то данный алгоритм можно будет применять для построения всех возможных обходов пунктов, необходимых для посещения. В частности, речь может идти о курьерах, которым будет крайне удобно оптимизировать свой маршрут путем расчёта минимального пути, по которому можно обойти как все пункты, так и их часть.