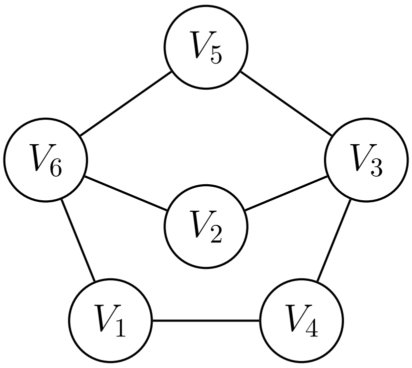
Вариант 22

1. Определить для орграфа, заданного матрицей смежности:

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

а) матрицу односторонней связности (2 способа, включая итерационный алгоритм);
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.



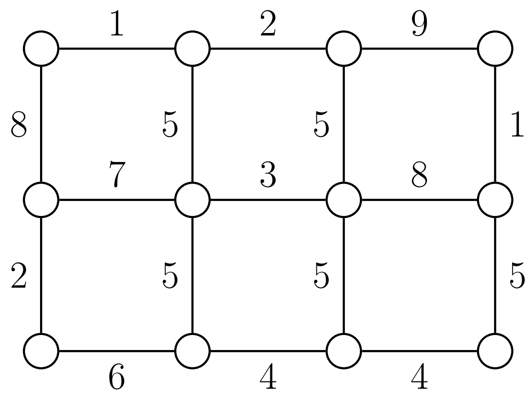
3. Используя алгоритм “фронта волны”, найти все минимальные пути из первой вершины в последнюю орграфа, заданного матрицей смежности.

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

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

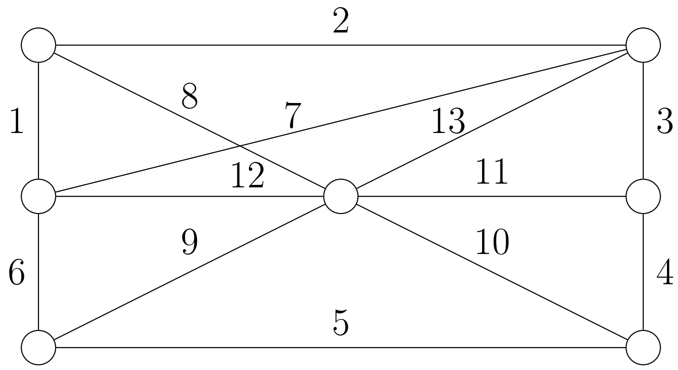
$$C = \begin{pmatrix} \infty & 2 & 7 & 8 & \infty & \infty & \infty \\ 12 & \infty & 4 & \infty & 6 & \infty & \infty \\ \infty & 4 & \infty & 1 & 3 & 5 & 7 \\ \infty & \infty & 1 & \infty & \infty & 3 & \infty \\ \infty & \infty & 3 & \infty & \infty & \infty & 5 \\ \infty & \infty & 5 & \infty & \infty & \infty & 2 \\ 2 & \infty & \infty & 3 & 4 & 6 & 7 \end{pmatrix}$$

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

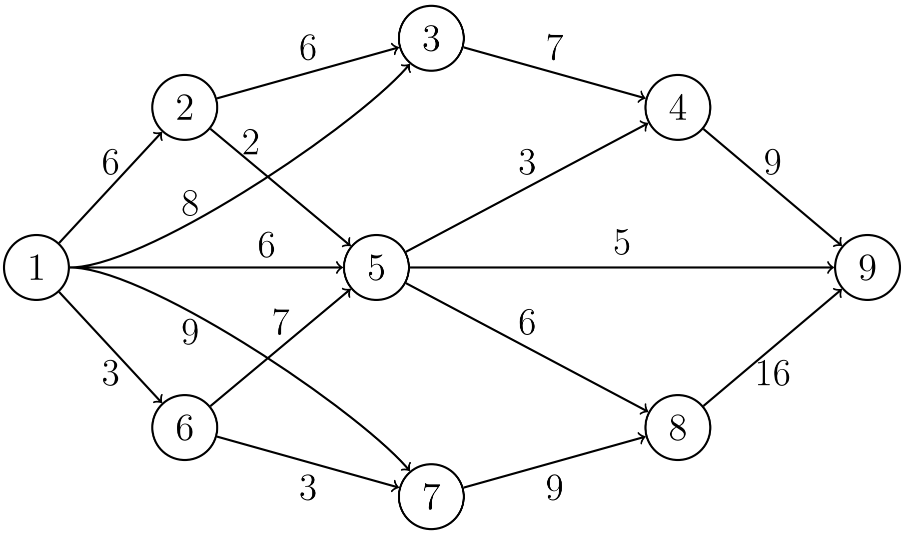


Значения X1 - X13 приведены в задании, значения X14 - X17 равны 5.

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.



Задание №1

а)

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

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

$$A^3 = A^2 * A = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 \end{pmatrix} * \begin{pmatrix} 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 1 & 1 & 0 \end{pmatrix}$$

$$T = E \vee A \vee A^2 \vee A^3$$

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

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

По итерационному алгоритму Уоршалла.

$$k = 0$$

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

$$k = 1, k - 1 = 0$$

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

$$k = 2, k - 1 = 1$$

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

$$k = 3, k - 1 = 2$$

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

$$k = 4, k - 1 = 3$$

$$T^4 = \begin{pmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\0&0&0&1\end{pmatrix} \vee \begin{pmatrix}0&1&1&1\\0&0&1&0\\0&1&0&0\\1&1&1&0\end{pmatrix} = \begin{pmatrix}1&1&1&1\\0&1&1&0\\0&1&1&0\\1&1&1&1\end{pmatrix} = T$$

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

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

в) Компоненты сильной связности

Выбираем первую строку, как ненулевую в матрице сильной связности

$$\overline{S} = \begin{pmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 \end{pmatrix}$$

Номера вершин первой компоненты сильной связности соответствуют номерам столбцов матрицы S, в которых в первой строке стоят единицы:

$\{V_1, V_4\}$

1. Обнуляем первый и четвертый столбец матрицы S. Получаем матрицу

$$\overline{S_1} = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}$$

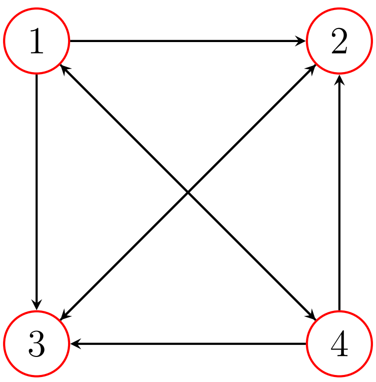
2. Ищем ненулевую строку матрицы S1: это вторая строка. Единицы две – во втором и третьем столбце. Следовательно, вторая компонента сильной связности: $\{V_2, V_3\}$.

3. Обнуляем третий столбец матрицы S1, получаем нулевую матрицу. Следовательно, других компонент сильной связности нет.

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

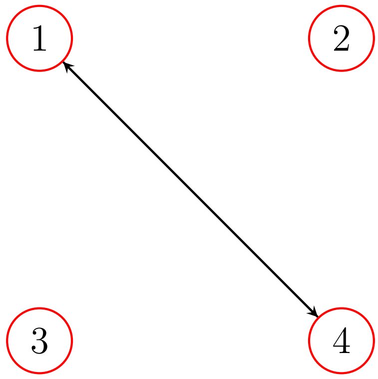
Дуги $<V_1, V_4>$ $<V_2, V_3>$, $<V_3, V_2>$, $<V_4, V_1>$ принадлежат какому-либо контуру исходного графа.

д) Граф

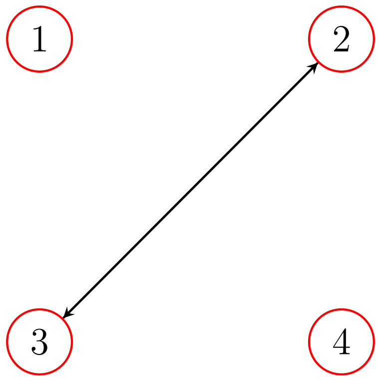


Компоненты сильной связности: $V_1 = \{v_1, v_4\}; V_2 = \{v_2, v_3\}$

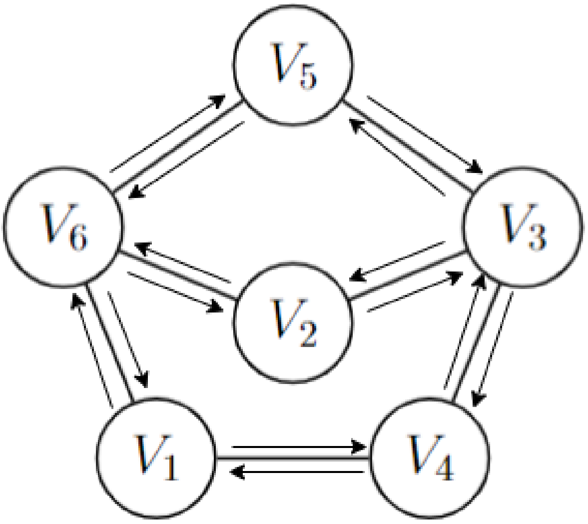
D1:



D2:



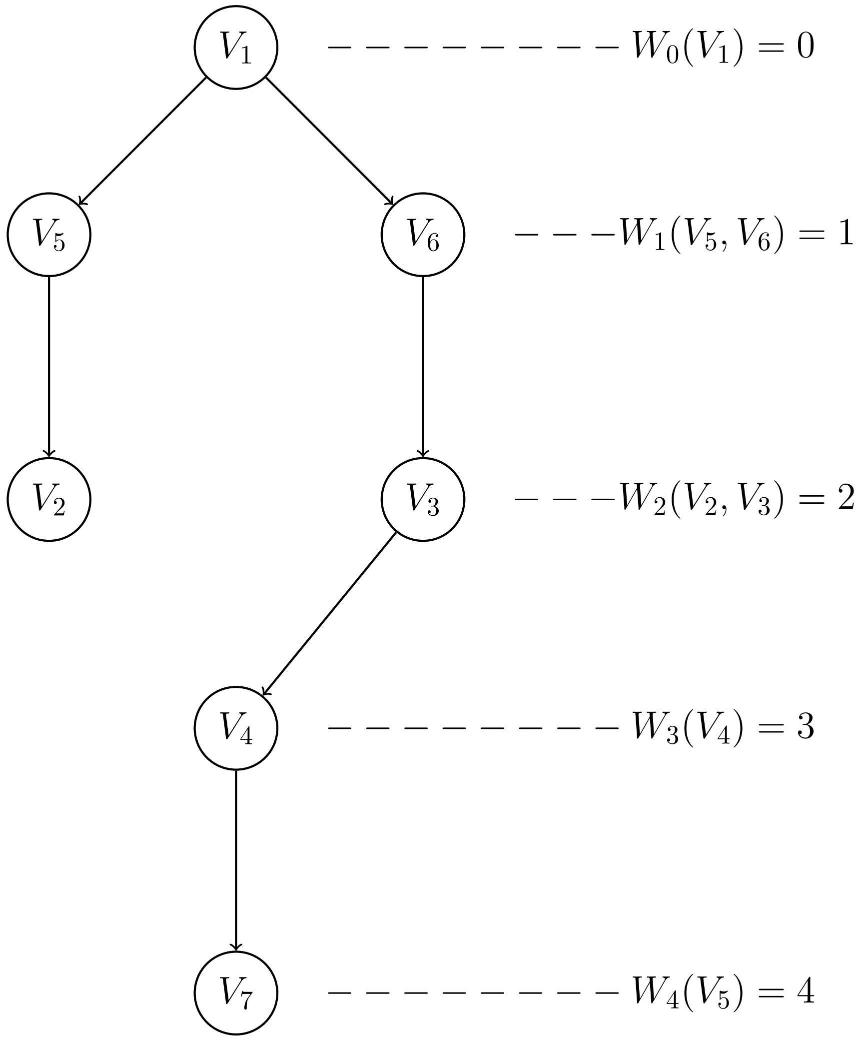
Задание №2



Маршрут: $1 \to 6 \to 5 \to 3 \to 4 \to 3 \to 2 \to 6 \to 2 \to 3 \to 5 \to 6 \to 1 \to 4 \to 1$

Задание №3

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



k	FW_k	Комментарий
0	$FW_0 = \{V_1\}$	В начальный момент времени волна сосредоточена в V_1
1	$FW_1 = \{V_5, V_6\}$	Видно из первой строки матрицы смежности
2	$FW_2 = \{V_5, V_6\}$	V_5, V_6 - источники распространения волны
3	$FW_3 = \{V_4\}$	Из вершины V_3 есть дуга в V_4
4	$FW_4 = \{V_7\}$	Достигнуто условие окончания прямого хода. Минимальная длина 4

	промежуточные вершины кратчайших путей
1	V_7
2	$W_3(V_1) \cap \Gamma^{-1} V_7 = \{V_4\} \cap \{V_4\} = \{V_4\}$
3	$W_2(V_1) \cap \Gamma^{-1} V_4 = \{V_2, V_3\} \cap \{V_3\} = \{V_3\}$
4	$W_1(V_1) \cap \Gamma^{-1} V_3 = \{V_5, V_6\} \cap \{V_6\} = \{V_6\}$
5	$W_0(V_1) \cap \Gamma^{-1} V_6 = \{V_1\} \cap \{V_1\} = \{V_1\}$

=> Кратчайший путь: $\mathrm{V}_1 \rightarrow V_6 \rightarrow V_3 \rightarrow V_4 \rightarrow V_7$

Задание №4

$$C = \begin{pmatrix} \infty & 2 & 7 & 8 & \infty & \infty & \infty \\ 12 & \infty & 4 & \infty & 6 & \infty & \infty \\ \infty & 4 & \infty & 1 & 3 & 5 & 7 \\ \infty & \infty & 1 & \infty & \infty & 3 & \infty \\ \infty & \infty & 3 & \infty & \infty & \infty & 5 \\ \infty & \infty & 5 & \infty & \infty & \infty & 2 \\ 2 & \infty & \infty & 3 & 4 & 6 & 7 \end{pmatrix}$$

Составим таблицу итераций:

	V_1	V_2	V_3	V_4	V_5	V_6	V_7	$\lambda_i^{(0)}$	$\lambda_i^{(1)}$	$\lambda_i^{(2)}$	$\lambda_i^{(3)}$	$\lambda_i^{(4)}$	$\lambda_i^{(5)}$	$\lambda_i^{(6)}$
V_1	∞	2	7	8	∞	∞	∞	0	0	0	0	0	0	0
V_2	12	∞	4	∞	6	∞	∞	∞	2	2	2	2	2	2
V_3	∞	4	∞	1	3	5	7	∞	7	6	6	6	6	6
V_4	∞	∞	1	∞	∞	3	∞	∞	8	8	7	7	7	7
V_5	∞	∞	3	∞	∞	∞	5	∞	∞	8	8	8	8	8
V_6	∞	∞	5	∞	∞	∞	2	∞	∞	11	11	10	10	10
V_7	2	∞	∞	3	4	6	7	∞	∞	14	13	13	12	12

1. Минимальный путь из $v_1 \to v_2$: $v_1 \to v_2$, его длина 2

   $\lambda_2^{(1)} = \lambda_1^{(0)} + C_{12} \quad (0 + 2 = 2)$

2. Минимальный путь из $v_1 \to v_3$: $v_1 \to v_2 \to v_3$, его длина 6

   $\lambda_3^{(2)} = \lambda_1^{(0)} + C_{23} \quad (2 + 4 = 6)$

   $\lambda_2^{(1)} = \lambda_3^{(1)} + C_{12} \quad (0 + 2 = 2)$

3. Минимальный путь из $v_1 \to v_4$: $v_1 \to v_2 \to v_3 \to v_4$, его длина 7

   $\lambda_4^{(3)} = \lambda_3^{(2)} + C_{34} \quad (6 + 1 = 7)$

   $\lambda_3^{(2)} = \lambda_2^{(1)} + C_{23} \quad (2 + 4 = 6)$

   $\lambda_2^{(1)} = \lambda_1^{(0)} + C_{12} \quad (0 + 2 = 2)$

④ Минимальный путь из $v_1 \to v_5$: $v_1 \to v_2 \to v_5$, его длина 8

$\lambda_5^{(2)} = \lambda_2^{(1)} + C_{25} \quad (2+6=8)$

$\lambda_2^{(1)} = \lambda_1^{(0)} + C_{12} \quad (0+2=2)$

⑤ Минимальный путь из $v_1 \to v_6$: $v_1 \to v_2 \to v_3 \to v_4 \to v_6$, его длина 10

$\lambda_6^{(4)} = \lambda_4^{(3)} + C_{46} \quad (7+3=10)$

$\lambda_4^{(3)} = \lambda_3^{(2)} + C_{34} \quad (6+1=7)$

$\lambda_3^{(2)} = \lambda_2^{(1)} + C_{23} \quad (2+4=6)$

$\lambda_2^{(1)} = \lambda_1^{(0)} + C_{12} \quad (0+2=2)$

⑥ Минимальный путь из $v_1 \to v_7$: $v_1 \to v_2 \to v_3 \to v_4 \to v_6 \to v_7$, его длина 12

$\lambda_7^{(5)} = \lambda_6^{(4)} + C_{67} \quad (10+2=12)$

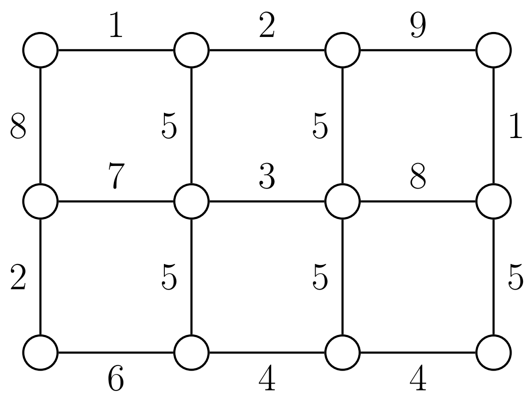
$\lambda_6^{(4)} = \lambda_4^{(3)} + C_{46} \quad (7+3=10)$

$\lambda_4^{(3)} = \lambda_3^{(2)} + C_{34} \quad (6+1=7)$

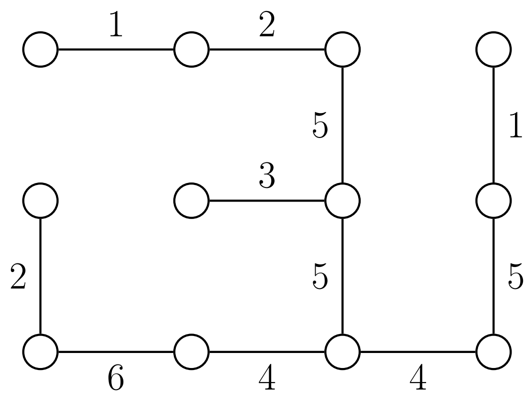
$\lambda_3^{(2)} = \lambda_2^{(1)} + C_{23} \quad (2+4=6)$

$\lambda_2^{(1)} = \lambda_1^{(0)} + C_{12} \quad (0+2=2)$

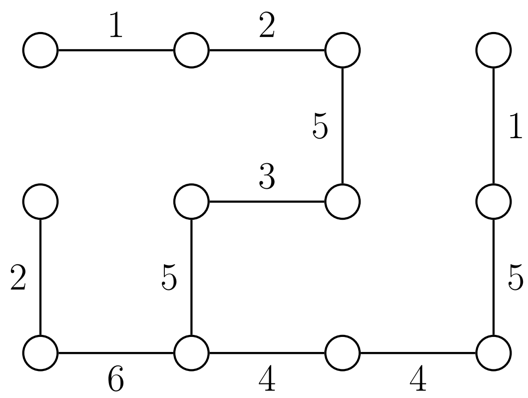
Задание №5

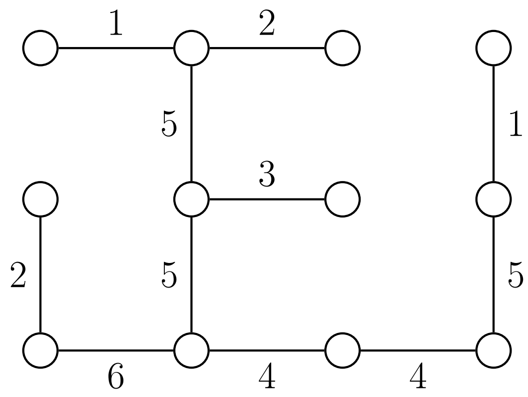


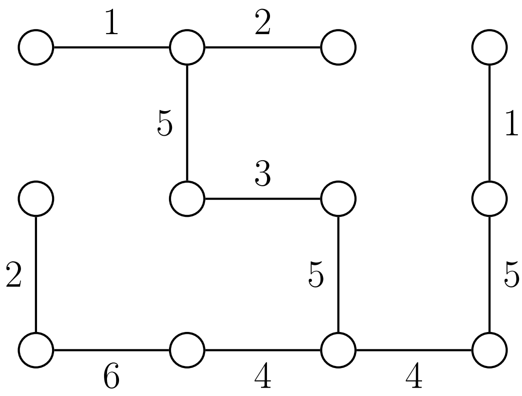
Возможное остовное дерево с минимальной суммой длин ребер 38:



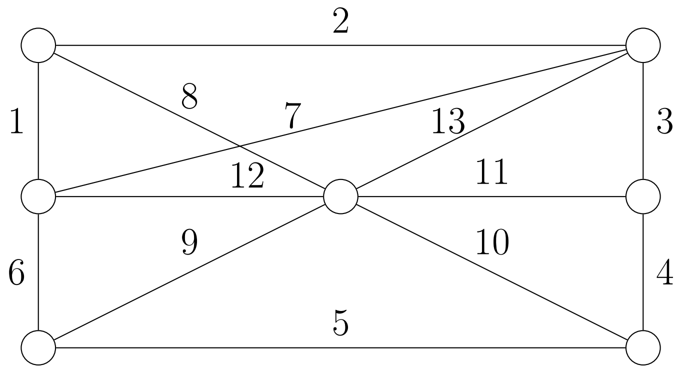
Минимальный вес остовного дерева $L(D) = 38$ Ещё есть три варианта остовного дерева с минимальной суммой длин рёбер - 38 :



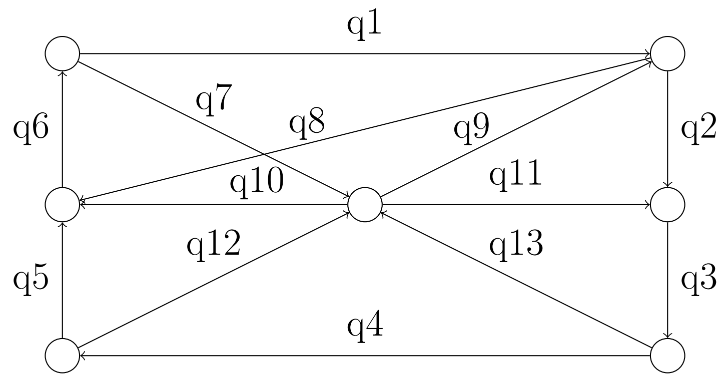



1
2
5
3
1
2
5
5
6
4
4

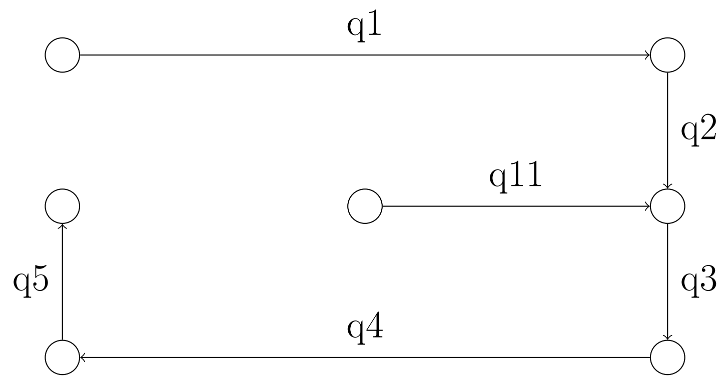
Задание №6



1. Зададим на графе произвольную ориентацию:



2. Построим произвольное остовное дерево D заданного графа:



$(D + q6) : \mu_1 : V_1 - V_2 - V_3 - V_7 - V_6 - V_5 - V_1$
$C(\mu_1) : (1, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0)$
$(D + q7) : \mu_2 : V_1 - V_2 - V_3 - V_4 - V_1$
$C(\mu_2) : (1, 1, 0, 0, 0, 0, -1, 0, 0, 0, -1, 0, 0)$
$(D + q8) : \mu_3 : V_2 - V_3 - V_7 - V_6 - V_5 - V_2$
$C(\mu_3) : (0, 1, 1, 1, 1, 0, 0, -1, 0, 0, 0, 0, 0)$

$(D+q9): \mu_4 : V_2 - V_3 - V_4 - V_2$
$C(\mu_4): (0, 1, 0, 0, 0, 0, 0, 0, 1, 0, -1, 0, 0)$
$(D+q10): \mu_5 : V_4 - V_3 - V_7 - V_6 - V_5 - V_4$
$C(\mu_5): (0, 0, 1, 1, 1, 0, 0, 0, 0, -1, 1, 0, 0)$
$(D+q12): \mu_6 : V_4 - V_3 - V_7 - V_6 - V_4$
$C(\mu_6): (0, 0, 1, 1, 0, 0, 0, 0, 0, 0, 1, 1, 0)$
$(D+q13): \mu_7 : V_4 - V_3 - V_7 - V_4$
$C(\mu_7): (0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 1, 0, 1)$

3. Составим цикломатическую матрицу:

	q1	q2	q3	q4	q5	q6	q7	q8	q9	q10	q11	q12	q13
μ_1	1	1	1	1	1	1	0	0	0	0	0	0	0
μ_2	1	1	0	0	0	0	-1	0	0	0	-1	0	0
μ_3	0	1	1	1	1	0	0	-1	0	0	0	0	0
μ_4	0	1	0	0	0	0	0	0	1	0	-1	0	0
μ_5	0	0	1	1	1	0	0	0	0	-1	1	0	0
μ_6	0	0	1	1	0	0	0	0	0	0	1	1	0
μ_7	0	0	1	0	0	0	0	0	0	0	1	0	1

4. Запишем закон Кирхгова для напряжений:

$$\begin{cases} u_1 + u_2 + u_4 + u_5 + u_6 = 0 \\ u_1 + u_2 - u_7 - u_11 = 0 \\ u_2 + u_3 + u_4 + u_5 - u_8 = 0 \\ u_2 + u_9 - u_11 = 0 \\ u3 + u_4 + u_5 - u_10 + u_11 = 0 \\ u_3 + u_4 + u_11 + u_12 = 0 \\ u_3 + u_11 + u_13 = 0 \end{cases}$$
$u_1, u_2, u_3, u_4, u_5, u_6, u_9 -$ базисные переменные

5. Найдём матрицу инцидентности:

	q1	q2	q3	q4	q5	q6	q7	q8	q9	q10	q11	q12	q13
u_1	-1	0	0	0	0	1	-1	0	0	0	0	0	0
u_2	1	-1	0	0	0	0	0	-1	1	0	0	0	0
u_3	0	1	-1	0	0	0	0	0	0	0	0	1	0
u_4	0	0	0	0	0	0	1	0	-1	-1	-1	1	1
u_5	0	0	0	0	1	-1	0	1	0	1	0	0	0
u_6	0	0	0	1	-1	0	0	0	0	0	0	-1	0
u_7	0	0	1	-1	0	0	0	0	0	0	0	0	-1

$$\begin{cases} -I_1 + I_6 - I_7 = 0 \\ I_1 - I_2 + I_8 + I_9 = 0 \\ I_2 - I_3 + I_{11} = 0 \\ I_7 - I_9 - I_{10} - I_{11} + I_{12} + I_{13} = 0 \\ I_5 - I_6 + I_8 + I_{10} = 0 \\ I_4 - I_5 - I_{12} = 0 \\ I_3 - I_4 - I_{13} = 0 \end{cases}$$

6. Подставим закон Ома:

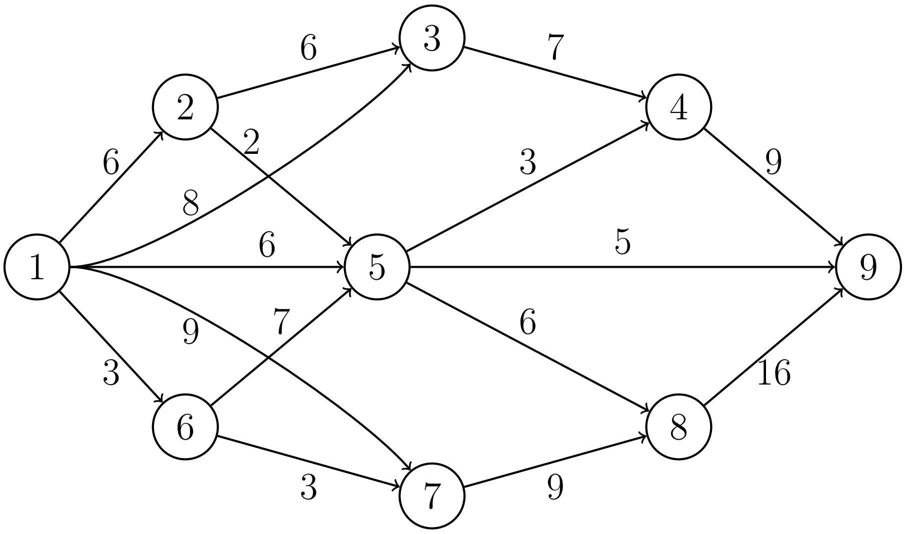
$$\begin{cases} \varepsilon_1 = I_7R_7 + I_9R_9 \\ 0 = -I_2R_2 - I_9R_9 + I_{11}R_{11} \\ 0 = -I_3R_3 - I_{11}R_{11} - I_{13}R_{13} \\ 0 = -I_4R_4 - I_{12}R_{12} + I_{13}R_{13} \\ \varepsilon_2 = I_{10}R_{10} + I_{12}R_{12} \\ 0 = -I_6R_6 - I_7R_7 - I_{10}R_{10} \\ 0 = -I_8R_8 - I_9R_9 + I_{10}R_{10} \end{cases}$$

7. Совместная система имеет вид:

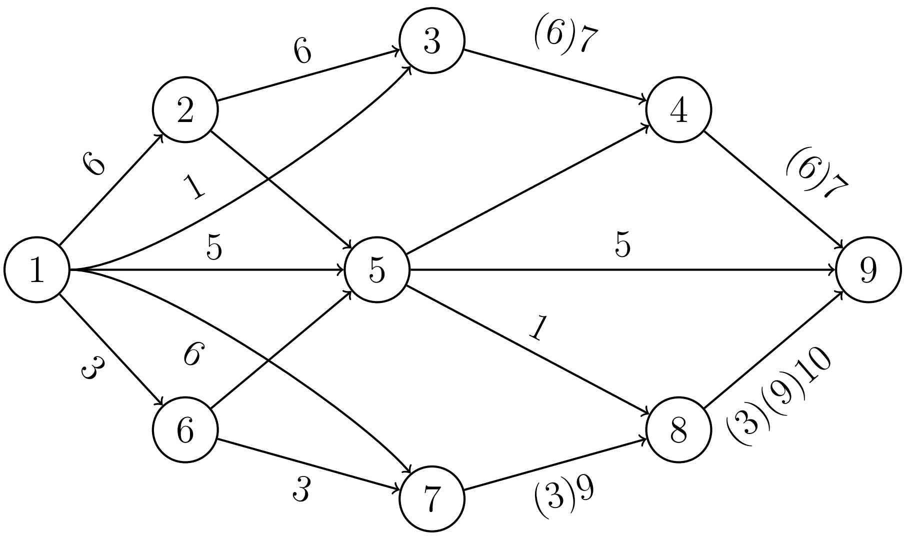
$$\begin{cases}
-I_1 + I_6 - I_7 = 0 \\
I_1 - I_2 + I_8 + I_9 = 0 \\
I_2 - I_3 + I_{11} = 0 \\
I_5 - I_6 + I_8 + I_{10} = 0 \\
I_4 - I_5 - I_{12} = 0 \\
I_3 - I_4 - I_{13} = 0 \\
\varepsilon_1 = I_7 R_7 + I_9 R_9 \\
0 = -I_2 R_2 - I_9 R_9 + I_{11} R_{11} \\
0 = -I_3 R_3 - I_{11} R_{11} - I_{13} R_{13} \\
0 = -I_4 R_4 - I_{12} R_{12} + I_{13} R_{13} \\
\varepsilon_2 = I_{10} R_{10} + I_{12} R_{12} \\
0 = -I_6 R_6 - I_7 R_7 - I_{10} R_{10} \\
0 = -I_8 R_8 - I_9 R_9 + I_{10} R_{10}
\end{cases}$$

13 уравнений и 13 неизвестных - $\mathrm{I}_1, I_2, .., \mathrm{I}_{13}$;
ЭДС ε_1, ε_2 и сопротивления $\mathrm{R}_1, R_3, R_4, R_6, R_7, .., \mathrm{R}_{13}$ известны.

Задание №7



Построение полного потока:



1. $v_1 - v_2 - v_3 - v_4 - v_9$
   - $\min\{6-0, 6-0, 7-0, 9-0\} = 6$
2. $v_1 - v_3 - v_4 - v_9$
   - $\min\{8-0, 7-6, 9-6\} = 1$
3. $v_1 - v_5 - v_9$
   - $\min\{6-0, 5-0\} = 5$
4. $v_1 - v_7 - v_8 - v_9$

- $\min\{10-0, 9-3, 16-3\} = 6$

5. $v_1 - v_6 - v_7 - v_8 - v_9$

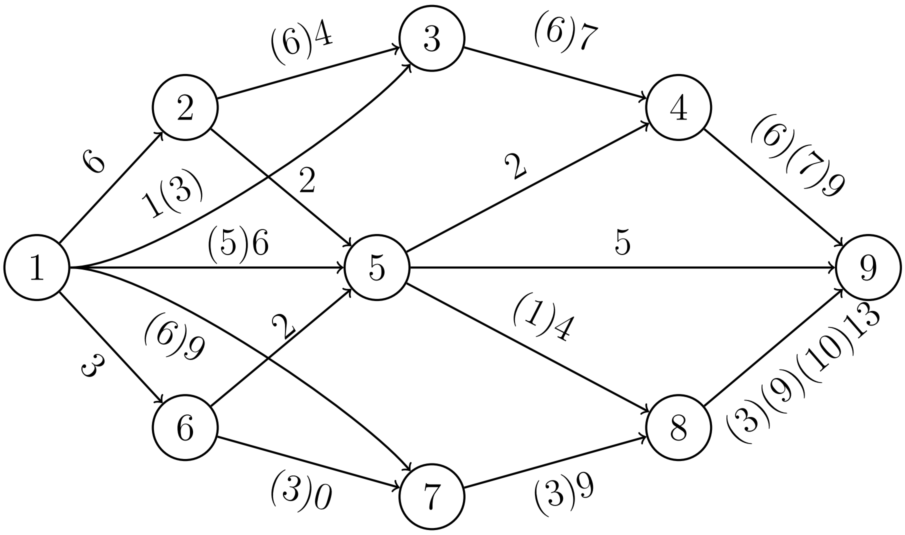
   - $\min\{3-0, 3-0, 9-0, 16-0\} = 3$

6. $v_1 - v_5 - v_8 - v_9$

   - $\min\{6-5, 6-0, 16-9\} = 1$

Величина полного потока $\Phi_{\text{пол.}} = 7 + 5 + 10 = 22$

Построение максимального потока



1. $v_1 - v_3 - v_2 - v_5 - v_4 - v_9$

   - $\Delta_1 = \min\{8-1, 2-0, 4, 3-0, 9-7\} = 2$

2. $v_1 - v_7 - v_6 - v_5 - v_8 - v_9$

   - $\Delta_2 = \min\{10-6, 7-0, 3, 6-1, 16-10\} = 3$

Величина максимального потока $\Phi_{\text{макс.}} = 9 + 5 + 13 = 27$

Следовательно, величина потока увеличилась на 5

Задание №8

Раскраска вершин гиперграфа

1. Основные понятия и определения

Гиперграф представляет собой пару (V, E), где V - непустое множество объектов, называемых вершинами гиперграфа, а E - семейство непустых (необязательно различных) подмножеств множества V, называемых рёбрами гиперграфа.

Задача раскраски вершин гиперграфа - это задача, в которой необходимо раскрасить вершины гиперграфа таким образом, чтобы любые две смежные вершины имели разные цвета. Цель задачи - найти минимальное количество цветов, необходимых для раскраски всех вершин гиперграфа.

2. Описание алгоритма

Инициализация:

- Создается фигура и оси для отрисовки гиперграфа с размером $(10, 10)$.
- Ограничения по осям устанавливаются таким образом, чтобы гиперграф целиком поместился в отрисовку.
- Определяются доступные цвета для раскраски вершин.

Определение функции available_colors(vertex):

- Данная функция принимает вершину vertex и возвращает множество доступных цветов для нее.
- Она проверяет все рёбра, инцидентные данной вершине, и добавляет цвета соседних вершин в множество adjacent_colors.
- Затем она возвращает разность всех доступных цветов и множества соседних цветов.

Цикл по вершинам:

- Для каждой вершины vertex в диапазоне от 0 до $n - 1$ выполняется следующее:
  - Вычисляются доступные цвета для вершины с помощью функции
  - Если доступные цвета не пусты, выбирается первый цвет из множества доступных цветов. В противном случае, цвет выбирается из списка доступных цветов по модулю от количества уже раскрашенных вершин.
  - Раскрашенный цвет сохраняется в словаре vertex_colors.

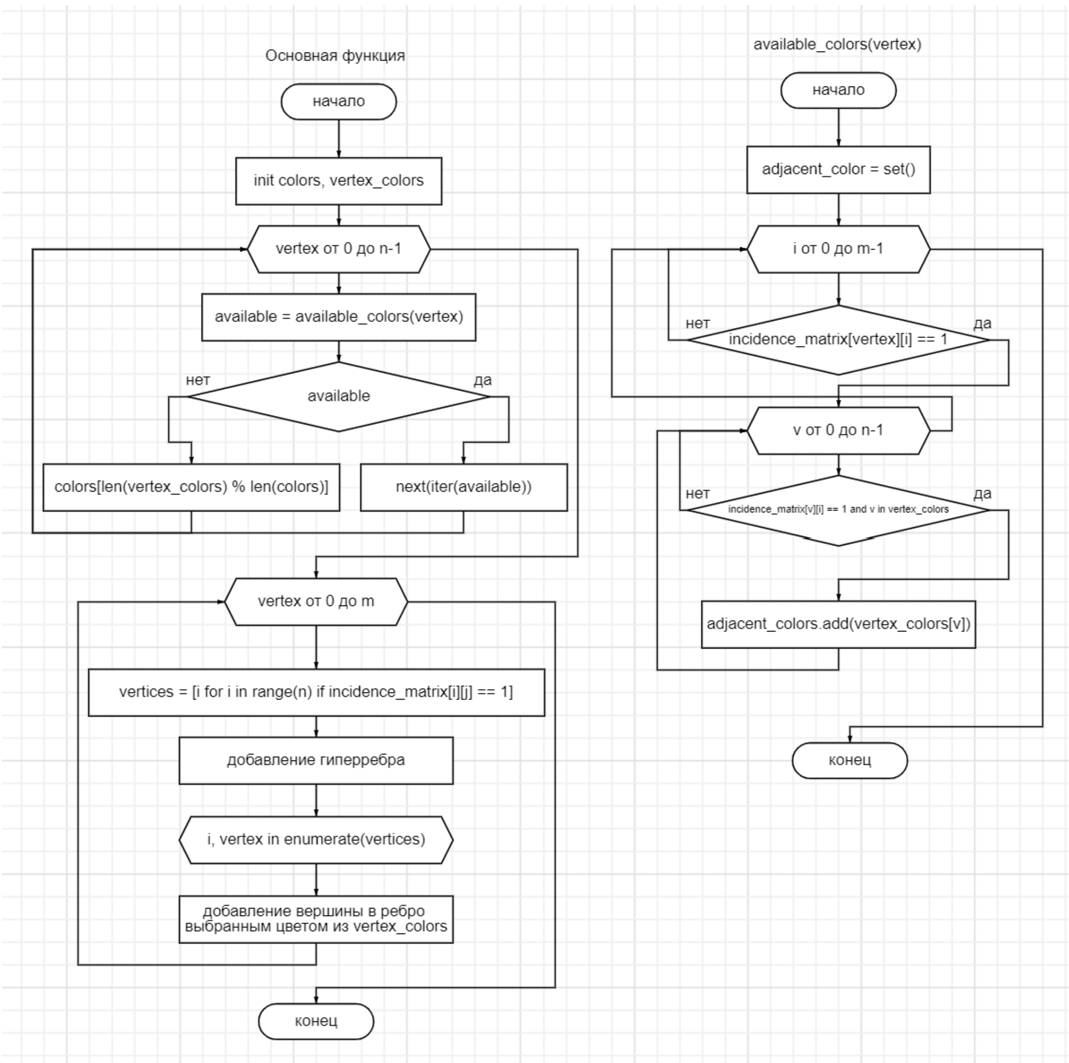
Цикл по рёбрам:

- Для каждого гиперребра j в диапазоне от 0 до $m - 1$ выполняется следующее:
- Определяются все вершины, инцидентные текущему гиперребру, и располагаются внутри прямоугольника, представляющего гиперребро.
- Для каждой вершины внутри гиперребра создается круг с соответствующим цветом и номером вершины, который добавляется на отрисовку.

Завершение:

- Устанавливается соотношение сторон для сохранения пропорций.
- Ось отрисовки скрывается.
- Отрисовывается фигура с гиперграфом.

3. Логическая блок схема



4. Описание работы и инструкция по работе с ней

Описание программы:

Программа представляет собой графический интерфейс, позволяющий пользователю вводить матрицу инцидентности для гиперграфа и визуализировать его с использованием библиотеки matplotlib. Пользователь может ввести количество вершин и ребер, а затем ввести матрицу инцидентности для каждого ребра. Программа автоматически раскрашивает вершины гиперграфа и визуализирует гиперребра и вершины на графике.

Инструкция по работе с программой:

1. Запустите программу.
2. В открывшемся диалоговом окне введите количество вершин ‘n‘ и количество ребер ‘m‘.
3. Откроется новое окно, где вам нужно будет ввести матрицу инцидентности для каждого ребра.
4. После ввода матрицы инцидентности нажмите кнопку "Submit".
5. Программа выведет матрицу инцидентности в консоль.
6. После закрытия консоли откроется графическое окно с визуализацией гиперграфа.
7. Вы можете наблюдать раскраску вершин и гиперребер на графике.

5. Вычисление сложности алгоритма

Внешний цикл `for vertex in range(n)` проходит по каждой вершине ровно один раз, поэтому его сложность составляет $O(n)$. Внутренняя функция `available_colors(vertex)` также проходит по каждой вершине, поэтому ее сложность также $O(n)$. Внутри функции `available_colors(vertex)` есть вложенные циклы, которые проходят по всем ребрам и вершинам, поэтому их сложность $O(m \cdot n)$. Общая сложность раскраски вершин:

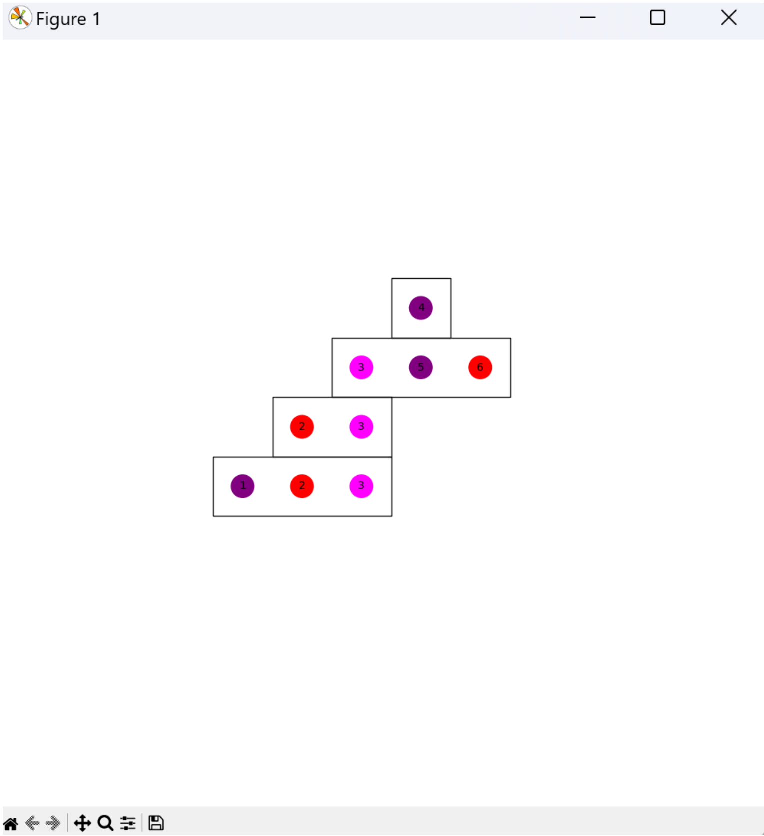
$$O(n \cdot (n + m)) = O(n^2 + nm) \approx O(n^2)$$

6. Тестовый пример

Дана матрица инцидентности

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

7. Скриншоты программы для данного примера



8. Прикладная задача

Распределение ресурсов в проекте

Предположим, у вас есть проект, состоящий из множества задач, которые должны быть выполнены определенными сотрудниками. Каждый сотрудник может быть занят выполнением других задач, и некоторые задачи могут быть зависят от других. Вы хотите распределить задачи между сотрудниками таким образом, чтобы каждый сотрудник работал над задачами, которые не зависят от других, и чтобы задачи, которые зависят от других, были выполнены после завершения их зависимостей.

Вы можете представить это в виде гиперграфа, где вершины представляют сотрудников, а гиперребра - задачи. Если задача зависит от другой задачи, то между этими задачами есть гиперребро. Затем вы можете использовать алгоритм раскраски гиперграфа для распределения задач между сотрудниками таким образом, чтобы каждый сотрудник работал над задачами, которые не зависят от других, и чтобы задачи, которые зависят от других, были выполнены после завершения их зависимостей.