Вариант 04

Задача 1. Постройте таблицу истинности для заданной формулы.
$(x_1 \vee x_2)x_2 \to (\overline{x_1} \vee x_2)\overline{x_3}$

Решение.

Обозначим: $F = (x_1 \vee x_2)x_2 \to (\overline{x_1} \vee x_2)\overline{x_3}$. Строим таблицу:

x_1	x_2	x_3	$\overline{x_1}$	$\overline{x_3}$	$x_1 \vee x_2$	$(x_1 \vee x_2)x_2$	$\overline{x_1} \vee x_2$	$(\overline{x_1} \vee x_2)\overline{x_3}$	F
0	0	0	1	1	0	0	1	1	**1**
0	0	1	1	0	0	0	1	0	**1**
0	1	0	1	1	1	1	1	1	**1**
0	1	1	1	0	1	1	1	0	**0**
1	0	0	0	1	1	0	0	0	**1**
1	0	1	0	0	1	0	0	0	**1**
1	1	0	0	1	1	1	1	1	**1**
1	1	1	0	0	1	1	1	0	**0**

Задача 2. Преобразовать данную формулу так, чтобы она содержала только операции тесного отрицания, дизъюнкции и конъюнкции. Пользуясь свойствами операций дизъюнкции и конъюнкции, привести формулу к виду, не содержащему скобок.
$(\overline{x_3} \sim x_2) \to (x_1 \vee x_2)\overline{x_3}$

Решение. Будем использовать известные тождества
$x \to y = \overline{x} \vee y$, $\overline{\overline{x}} = x$,
$x \sim y = (x \cdot y) \vee (\overline{x} \cdot \overline{y})$, $\overline{x \sim y} = (x \cdot \overline{y}) \vee (\overline{x} \cdot y)$,
законы Моргана (двойственности): $\overline{(x \vee y)} = (\overline{x} \cdot \overline{y})$, $\overline{(x \cdot y)} = (\overline{x} \vee \overline{y})$.

Получаем:
$$F = (\overline{x_3} \sim x_2) \to (x_1 \vee x_2)\overline{x_3} = \overline{(\overline{x_3} \sim x_2)} \vee (x_1 \vee x_2)\overline{x_3} =$$
$$= \left[(\overline{x_3} \cdot \overline{x_2}) \vee (\overline{\overline{x_3}} \cdot x_2)\right] \vee (x_1 \vee x_2)\overline{x_3} = \left[\overline{x_3} \cdot \overline{x_2} \vee x_3 \cdot x_2\right] \vee (x_1 \vee x_2)\overline{x_3} =$$

Теперь упростим полученное выражение, которое содержит только операции тесного отрицания, дизъюнкции и конъюнкции:

$$=\overline{x_2}\cdot\overline{x_3}\vee x_2\cdot x_3\vee\left(x_1\cdot\overline{x_3}\vee x_2\cdot\overline{x_3}\right)=\overline{x_2}\cdot\overline{x_3}\vee x_2\cdot x_3\vee x_1\cdot\overline{x_3}\vee x_2\cdot\overline{x_3}=$$
$$=\overline{x_2}\cdot\overline{x_3}\vee\left(x_2\cdot x_3\vee x_2\cdot\overline{x_3}\right)\vee x_1\cdot\overline{x_3}=\overline{x_2}\cdot\overline{x_3}\vee x_2\cdot\left(x_3\vee\overline{x_3}\right)\vee x_1\cdot\overline{x_3}=$$
$$=\overline{x_2}\cdot\overline{x_3}\vee x_2\vee x_1\cdot\overline{x_3}=\left(\overline{x_2}\vee x_2\right)\cdot\left(\overline{x_3}\vee x_2\right)\vee x_1\cdot\overline{x_3}=\left(\overline{x_3}\vee x_2\right)\vee x_1\cdot\overline{x_3}=$$
$$=\overline{x_3}\vee x_2\vee x_1\cdot\overline{x_3}=x_2\vee\left(\overline{x_3}\vee x_1\cdot\overline{x_3}\right)=x_2\vee\overline{x_3}\left(1\vee x_1\right)=x_2\vee\overline{x_3}.$$

Получили $F=x_2\vee\overline{x_3}$.

Задача 3. Из колоды в 36 карт вынимают n карт. Указать число наборов, содержащих ровно m карт бубновой масти и k карт пиковой масти. Рассмотреть случаи выбора с возвращением и без возвращения.

Производится упорядоченный выбор. $n=8, m=3, k=2$.

Решение.
Выбирается 8 карт из колоды в 36, из них должно быть 3 бубновых, 2 пиковых, 3 другой масти.

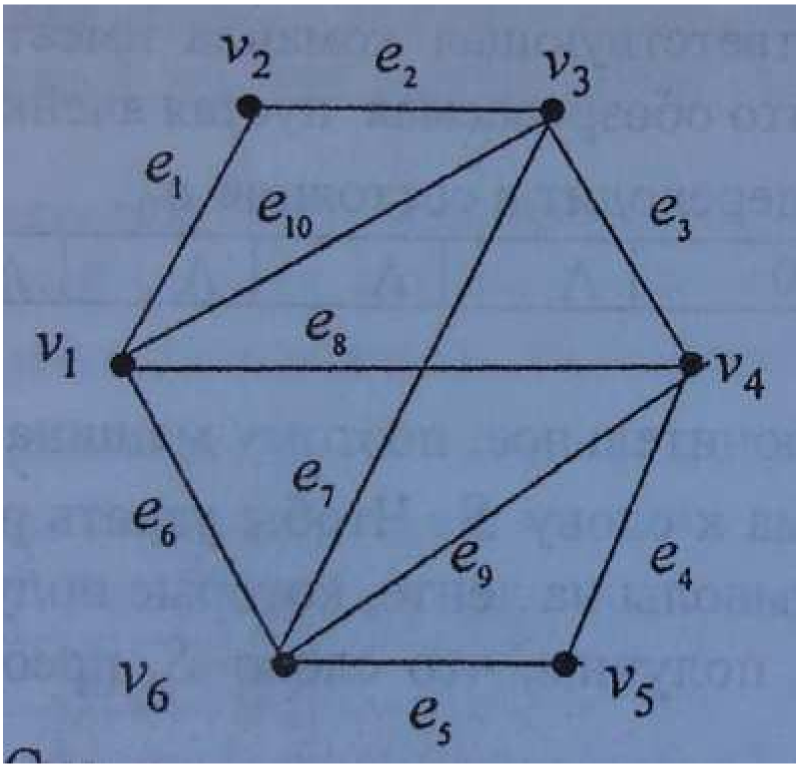
Случай 1. Выбор с возвращением.
Так как карта после выбора возвращается в колоду, количество бубновых (9 карт), пиковых (9 карт) и других карт (18 карт) в колоде не меняются.

Тогда количество способов выбрать 8 карт из 36 так, чтобы из них 3 были бубновой масти будет $9\cdot9\cdot9$, 2 карты пиковой масти будет $9\cdot9$ и еще 3 другой масти будет $18\cdot18\cdot18$. При этом, так как выбор упорядоченный, карты можно переставлять между собой, число различных перестановок равно $P_8(3,2,3)=\frac{8!}{3!\cdot2!\cdot3!}$.

Итого получаем $N_1=9^5\cdot18^3\cdot\frac{8!}{3!\cdot2!\cdot3!}$. способов.

Случай 2. Выбор без возвращения.
Нужно выбрать 6 карт из 36 так, чтобы из них 3 были бубновой масти (всего таких карт 9), 2 карты пиковой масти (всего таких карт 9) и еще 3 другой масти (всего их 18), откуда получаем по правилу умножения $N_2=C_9^3\cdot C_9^2\cdot C_{18}^3\cdot\frac{8!}{3!\cdot2!\cdot3!}$ способов (снова умножили на число способов упорядочить выбранные карты).

Задача 4. Пользуясь алгоритмом Дейкстры, найти кратчайшие маршруты из вершины v_1 неориентированного взвешенного графа в другие вершины графа. Указать кратчайший маршрут из вершины v_1 в вершину v_4.



e_1	e_2	e_3	e_4	e_5	e_6	e_7	e_8	e_9	e_{10}
3	2	3	1	2	2	1	7	1	1

Решение. Найдем кратчайший путь от вершины v_1 до всех вершин, используя *алгоритм Дейкстры*. Он заключается в том, что вершинам графа присваиваются временные метки, которые затем по определенным правилам заменяются на постоянные метки. Будем использовать обозначения:

$L^*(v_i)$ - постоянная метка вершины v_i,

$L^н(v_i)$ - новая временная метка вершины v_i,

$L^c(v_i)$ - старая временная метка вершины v_i,

R_{ij} - вес ребра, соединяющего вершины v_i и v_j.

Новая временная метка вычисляется по формуле:

$$L^н(v_j) = \min\left\{L^c(v_i), R_{ij} + L^*(v_i)\right\}$$

После этого из всех временных меток выбирается наименьшая, и она становится постоянной меткой. Действия продолжаются, пока не будут найдены постоянные метки для всех вершин графа. Результаты действий на каждом шаге будем заносить в таблицу. В предпоследний столбец заносим вершину, получившую постоянную метку, в последний столбец – величину этой метки (для данного шага).

Шаг 1. Начальная вершина v_1, имеет постоянную метку $L^*(v_1)=0$, остальные вершины имеют временную метку ∞.

Шаг 2. Определяем множество последователей вершины $\Gamma(v_1)=\{v_2,v_3,v_4,v_6\}$. Пересчитываем их временные метки по основной формуле. $L^i(v_2)=\min\{\infty,0+3\}=3$, $L^i(v_3)=\min\{\infty,0+1\}=1$, $L^i(v_4)=\min\{\infty,0+7\}=7$, $L^i(v_6)=\min\{\infty,0+2\}=2$. Берем вершину v_3 с минимальной временной меткой 1, присваиваем этой вершине постоянную метку $L^*(v_3)=1$.

Шаг 3. Определяем множество последователей вершины $\Gamma(v_3)=\{v_1,v_2,v_6,v_4\}$. Пересчитываем временные метки по основной формуле. $L^i(v_4)=\min\{7,1+3\}=4$, $L^i(v_2)=\min\{3,1+2\}=3$, $L^i(v_6)=\min\{2,1+1\}=2$ Берем вершину v_6 с минимальной временной меткой 2, присваиваем этой вершине постоянную метку $L^*(v_6)=2$.

Шаг 4. Определяем множество последователей вершины $\Gamma(v_6)=\{v_1,v_3,v_4,v_5\}$. Пересчитываем временные метки $L^i(v_4)=\min\{4,2+1\}=3$, $L^i(v_5)=\min\{\infty,2+2\}=4$. Берем вершину v_4 с минимальной временной меткой 3, присваиваем этой вершине постоянную метку $L^*(v_4)=3$.

Нашли длину кратчайшего пути от вершины v_1 до вершины v_4, он равен 3. Проходит через вершины: $v_1-v_6-v_4$.

Шаг 5. Определяем множество последователей вершины $\Gamma(v_4)=\{v_1,v_3,v_6,v_5\}$. Пересчитываем временные метки по основной формуле. $L^i(v_5)=\min\{4,3+1\}=4$ Берем вершину v_2 с минимальной временной меткой 3, присваиваем этой вершине постоянную метку $L^*(v_2)=3$.

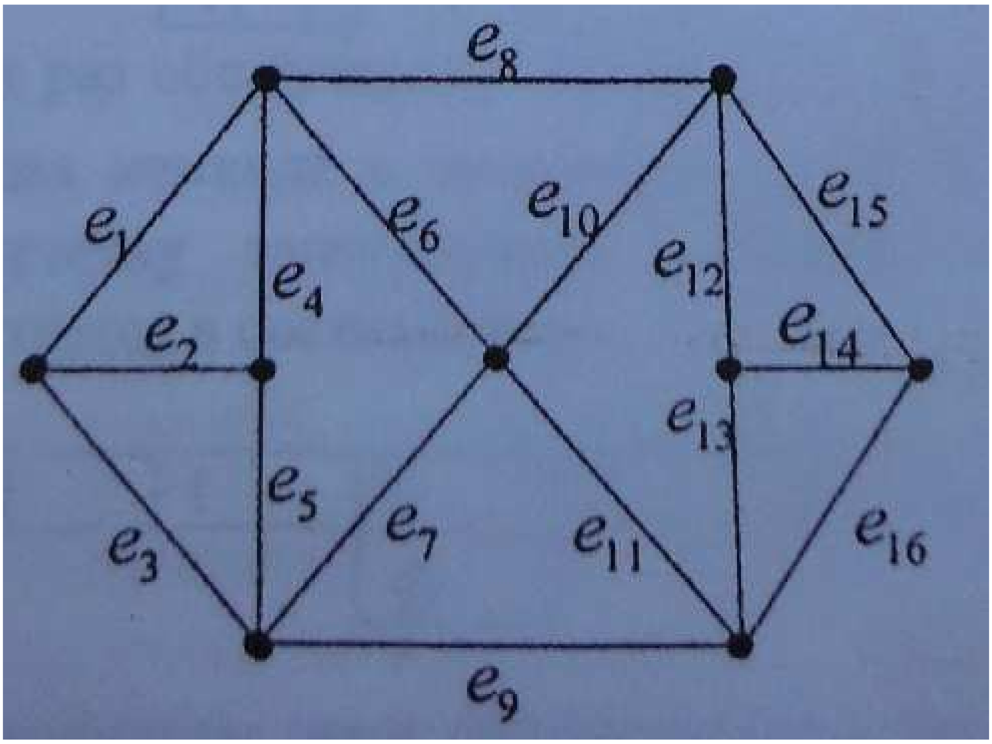
Шаг 4. Определяем множество последователей вершины $\Gamma(v_2)=\{v_1,v_3\}$. Берем вершину v_5 с минимальной временной меткой 4, присваиваем этой вершине постоянную метку $L^*(v_5)=4$.

Все вершины получили постоянную пометку, алгоритм завершен. Кратчайшие пути до каждой вершины указаны в последнем столбце расчетной таблицы.

Шаги	**Вершины**						v_i	$L^*(v_i)$
	v_1	v_2	v_3	v_4	v_5	v_6		
1	**0**	∞	∞	∞	∞	∞	v_1	0
2		3	**1**	7	∞	2	v_3	1

3		3		4	∞	**2**	v_6	2
4		3		**3**	4		v_4	3
5		**3**			4		v_2	3
6					**4**		v_5	4

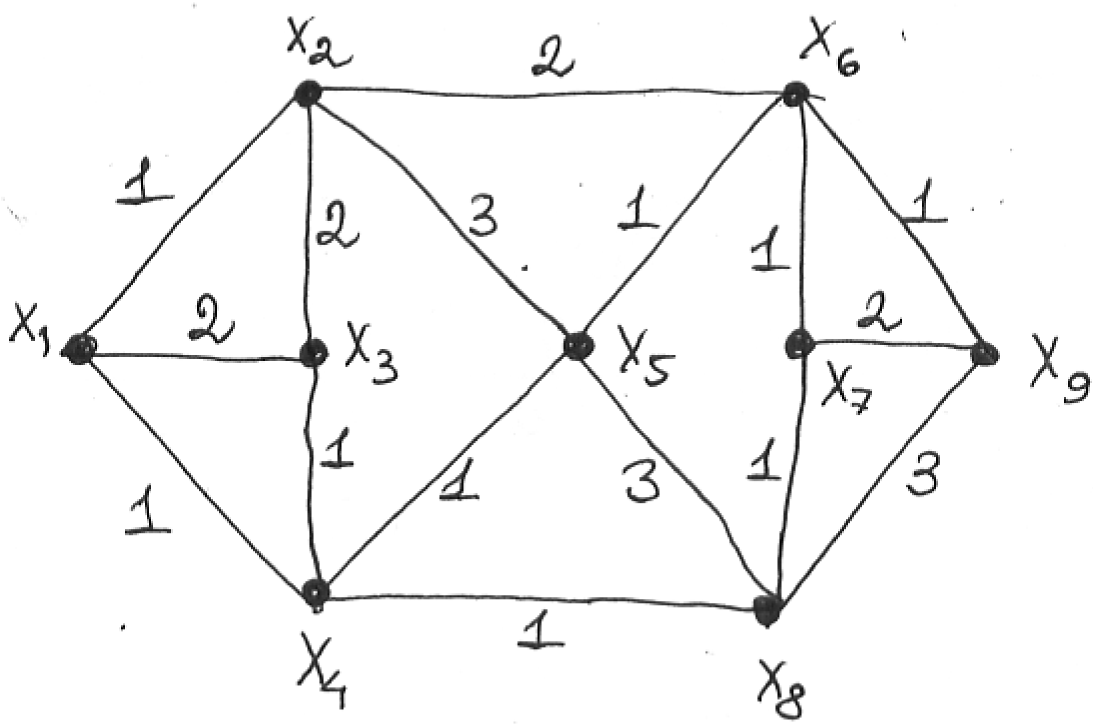
Задача 5. Схема дорог, соединяющих населенные пункты, задана графом, показанным на рисунке. В таблице каждому ребру графа поставлен в соответствие вес, характеризующий стоимость прокладки дороги, соединяющей данные населенные пункты. При помощи алгоритма Краскала построить схему дорог, соединяющую данные населенные пункты при наименьшей стоимости проекта.



e_1	e_2	e_3	e_4	e_5	e_6	e_7	e_8	e_9	e_{10}	e_{11}	e_{12}	e_{13}	e_{14}	e_{15}	e_{16}
1	2	1	2	1	3	1	2	1	1	3	1	1	2	1	3

Решение. Найдем остовное дерево минимального веса (то есть схему дорог наименьшей стоимости) с помощью алгоритма Краскала.

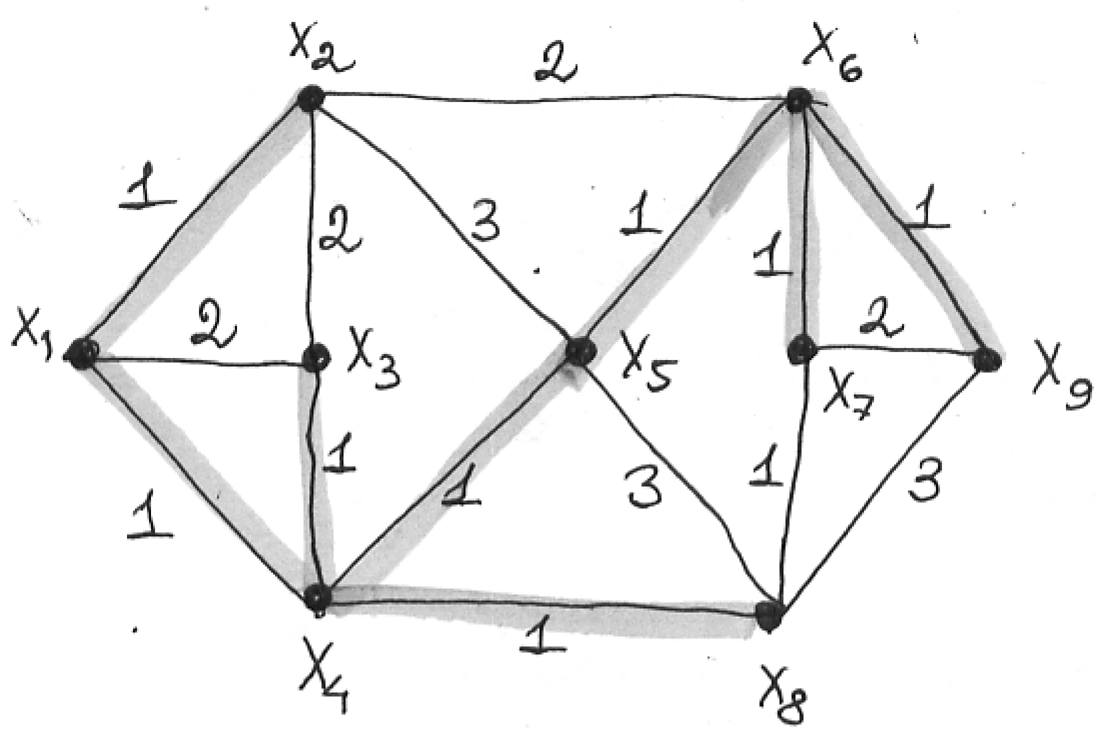
Пронумеруем вершины графа для удобства, ребра пометим весом (стоимостью) согласно варианту.



Начинаем с ребра минимального веса (x_1, x_2), вес 1.
Затем по очереди добавляем ребра минимального веса, так, чтобы они не образовали цикла.

Порядок добавления:
ребро (x_1, x_2), вес 1.
ребро (x_1, x_4), вес 1.
ребро (x_3, x_4), вес 1.
ребро (x_5, x_4), вес 1.
ребро (x_4, x_8), вес 1.
ребро (x_5, x_6), вес 1.
ребро (x_6, x_7), вес 1.
ребро (x_6, x_9), вес 1.

Все вершины добавлены в дерево. Минимальное остовное дерево построено. Обведем его на чертеже:



Общий вес (минимальная стоимость проекта) 1+1+1+1+1+1+1+1=8.

Задача 6. Выяснить, применима ли машина Тьюринга, заданная программой P, к слову S, и если применима, то указать результат применения машины Тьюринга к данному слову.

$$P=\begin{cases} q_1 0 \to 0 q_2 \\ q_1 1 \to 1 R q_2 \\ q_2 0 \to 0 R q_3 \\ q_2 1 \to 1 R q_3 \\ q_3 0 \to 1 R q_0 \\ q_3 1 \to 1 L q_2 \end{cases}$$

$S = 110101$.

Решение. Начальное состояние машины q_1. Начнем применять машину, заданную программой P, к слову S. Двигаемся слева направо.

Шаг 1. Начальное слово $...0q_1 110101...$, состояние $q_1 1$. Многоточием обозначили бесконечную ленту (заполненную вообще говоря нулями).

Шаг 2. Применяем правило $q_1 1 \to 1 R q_2$, то есть сдвигаемся вправо на 1, единица сохраняется, переходим в состояние q_2. Получаем слово: $...0q_1 110101... \to ...01q_2 10101...$

Состояние $q_2 1$.
Шаг 3. Применяем правило $q_2 1 \to 1Rq_3$, то есть сдвигаемся вправо на 1, единица сохраняется, переходим в состояние q_3. Получаем слово:
$...01q_2 10101... \to ...011q_3 0101...$
Состояние $q_3 0$.

Шаг 4. Применяем правило $q_3 0 \to 1Rq_0$, то есть сдвигаемся вправо на 1, нуль меняется на 1, переходим в состояние q_0. Получаем слово:
$...011q_3 0101... \to ...0111q_0 101...$
Состояние $q_0 1$.

Так как мы пришли к конечному состоянию, программа машины Тьюринга применима к данному слову и переводит его в слово $T = 111101$ (меняет первый нуль на 1).