Exercise 1. Let D be a loopless digraph, F a spanning forest of D, and $\mathbf{C}_F$ the basis matrices of the cycle space $\mathcal{C}(D)$ corresponding to F. Show that $\mathbf{C}_F$ is unimodular.

pf: Let P be an abitrary full order submatrix of C_F,
R be the edge set of P in D,
F_1 be the induced graph $D[R]$.
Therefore, $P = C_F|_{\bar{F}_1}$.

① If $\bar{F}_1$ contains a bond,
then the columns of $P = C_F|_{\bar{F}_1}$ are linearly dependent.
Thus $\det P = 0$.

② If $\bar{F}_1$ contains no bonds,
then the columns of P are linearly independent.
Obviously, P is an $(m-n+w)\times(m-n+w)$ matrix.
Thus $|R| = m-n+w$, F_1 is a spanning forest in D.
By the transition matrix's equation,
$C_F|_{\bar{F}_1} \cdot C_{F_1} = C_F$.
Under the restriction of $\bar{F}$,
$C_F|_{\bar{F}_1} \cdot C_{F_1}|_{\bar{F}} = C_F|_{\bar{F}} = I_{m-n+w}$.
Calculate the determinant,
$\det C_F|_{\bar{F}_1} \cdot \det C_{F_1}|_{\bar{F}} = 1$
The coefficients of those determinant are integers.

Thus $\det P = \det C_{F|\bar{F}} = \pm 1$,

C_F is unimodular.

□.

Exercise 2. Let D be a nontrivial connected loopless digraph and a any edge of D. Prove:

$$\tau(D) = \tau(D \cdot a) + \tau(D - a).$$

Here $D \cdot a$ is the digraph obtained from D by contracting a: delete a, identify its endpoints u and v as a single vertex, and retain all other edges, with their directions preserved.

Pf: Let $T(D) = \{F : F \text{ is a spanning tree of } D\}$.

$T_a(D) = \{F : F \text{ is a spanning tree of } D \text{ and } a \in E(F)\}$

$T_{\bar{a}}(D) = \{F : F \text{ is a spanning tree of } D \text{ and } a \notin E(F)\}$.

Thus,

$\tau(D) = |T_a(D)| + |T_{\bar{a}}(D)|$

Obviously, $|T_{\bar{a}}(D)| = \tau(D - a)$.

Define a mapping from $T_a(D)$ to $T(D \cdot a)$

$\phi : T_a(D) \longrightarrow T(D \cdot a)$

$\phi(F) = F \cdot a$

Obviously ϕ is well defined because $F \cdot a \in T(D \cdot a)$

① ϕ is injective, because

$F_1 \cdot a = F_2 \cdot a \Rightarrow F_1 = F_2$.

② ϕ is surjective, because

$\forall F' \in T(D \cdot a)$, let v be the vertex shrinked by a, restore v to x and y in a, denote the graph as $\widetilde{F'}$. Then $\widetilde{F'} + a$ is still a tree, $\widetilde{F'} + a \in T_a(D)$, $\phi(\widetilde{F'} + a) = F'$.

Hence, $|T_a(D)| = |T(D \cdot a)|$.

Above all, $T(D) = |T(D \cdot a)| + T(D - a)$

$= T(D \cdot a) + T(D - a)$.

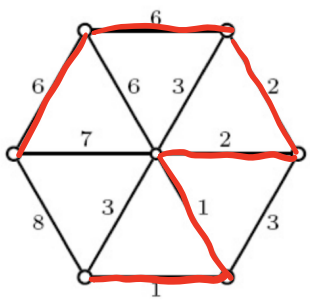
□.

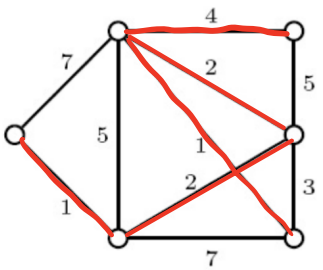
Exercise 3. The Kruskal algorithm for a minimum tree in a connected weighted simple graph (G, w) is as follows:

1. Choose $e_1 \in E(G)$ with $w(e_1)$ as small as possible;
2. once $e_1, e_2, \ldots, e_i$ are chosen, pick $e_{i+1} \in E(G) \setminus \{e_1, \ldots, e_i\}$ so that $G[\{e_1, \ldots, e_i, e_{i+1}\}]$ is acyclic and $w(e_{i+1})$ is as small as possible;
3. if $i < n - 1$, go to step 2; if $i = n - 1$, stop.

(a) Prove that the subgraph of G built by Kruskal's algorithm is a minimum tree of G.

(b) Use Kruskal's algorithm to find a minimum tree in the following graph.





(a) Pf:

It is equivalent to prove:

For each i, there exists a minimum spanning tree T_i, s.t. $F_i = \{e_j\}_{j=1}^{i} \subset E(T_i)$.

① If $i = 0$, then $F_0 = \phi$. Obviously it's correct.

② If it's true for i, then we choose a T_i which contains F_i.

Let $e = e_{i+1}$.

1. If $e \in E(T_i)$, then $E(T_i) \supset F_i \cup \{e\} = F_{i+1}$. we still choose T_i and it's correct.
2. If $e \notin E(T_i)$, then $T_i + e$ contains a

foundmental cycle C. Since e connects 2 connected components in this step,
then there exists $e' \in C \setminus F_i$, which is also a candidate of kruskal algorithm in step i+1.
By the minimality, $w(e) \le w(e')$.
Let $T_{i+1} = (T_i + e) - e'$, T_{i+1} is also a spanning tree, and $w(T_{i+1}) = w(T_i) + w(e) - w(e')$
$\le w(T_i)$.

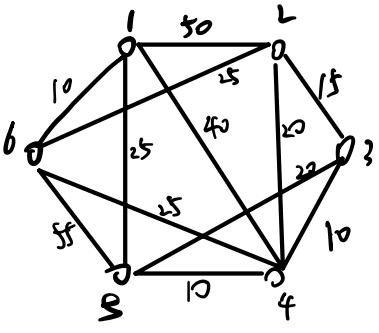
Thus T_{i+1} is also a minimum spanning tree.
Hence $F_{i+1} = F_i \cup \{e\} \subset E(T_{i+1})$.

□.

(b) The result is the red edges in the figure above.

Exercise 4. A company has branches in six cities $x_1, x_2, \ldots, x_6$. The direct airfare from x_i to x_j is the (i,j)-entry of the matrix below (∞ means no direct flight). Construct a table of the cheapest route between every pair of cities for the company.

$$A = \begin{pmatrix} 0 & 50 & \infty & 40 & 25 & 10 \\ 50 & 0 & 15 & 20 & \infty & 25 \\ \infty & 15 & 0 & 10 & 20 & \infty \\ 40 & 20 & 10 & 0 & 10 & 25 \\ 25 & \infty & 20 & 10 & 0 & 55 \\ 10 & 25 & \infty & 25 & 55 & 0 \end{pmatrix}$$

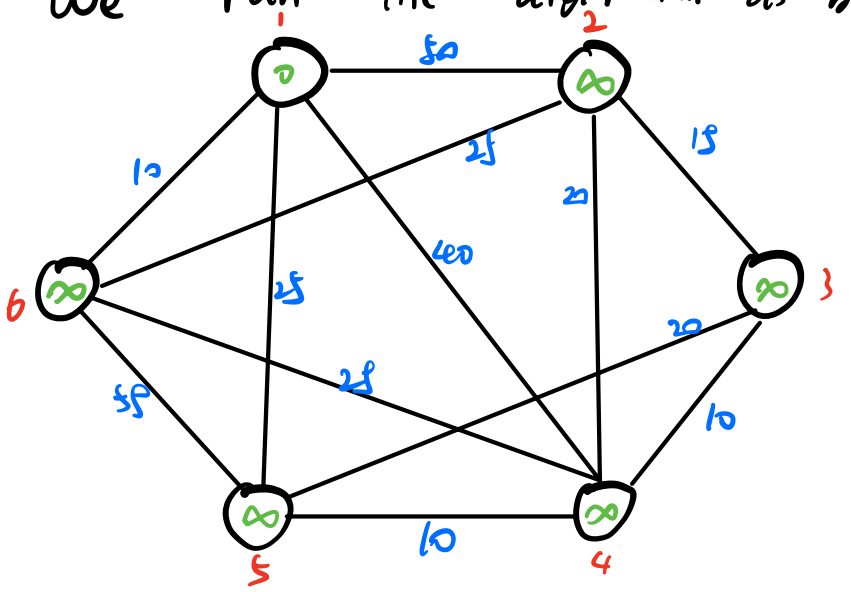


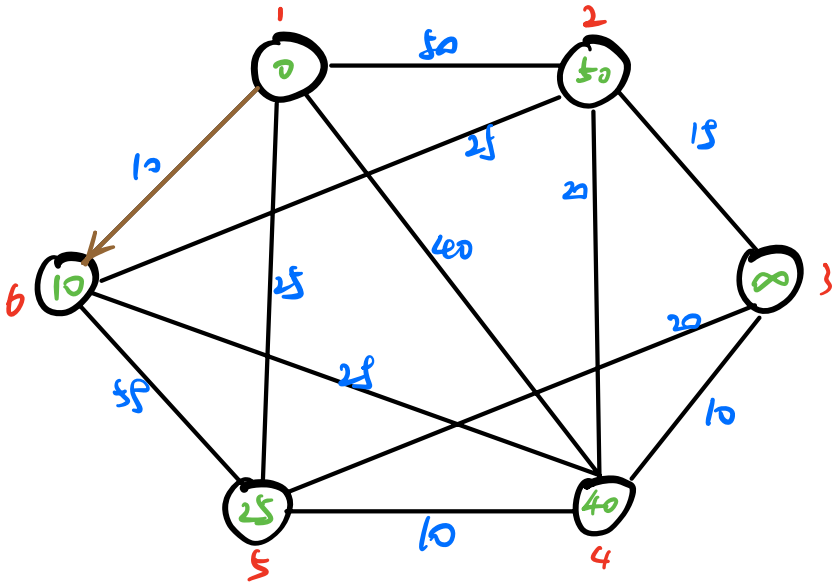
Let G be the graph such that
$V(G) = \{x_i\}_{i=1}^{6}$.
$E(G) = \{x_i x_j : A_{ij} \neq \infty, i \neq j\}$.
$w \in |E(G)|$, $w(x_i x_j) = A_{ij}$.

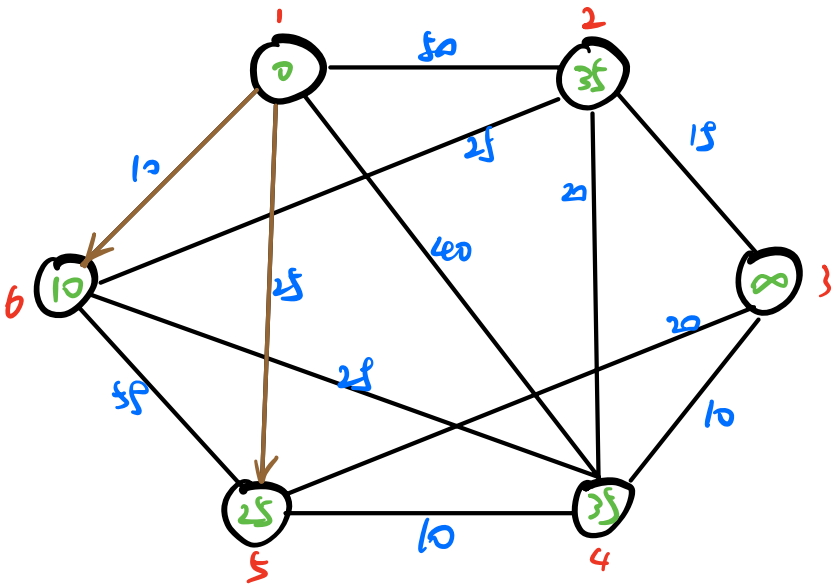
By Moore - Dijkstra Algorithm,

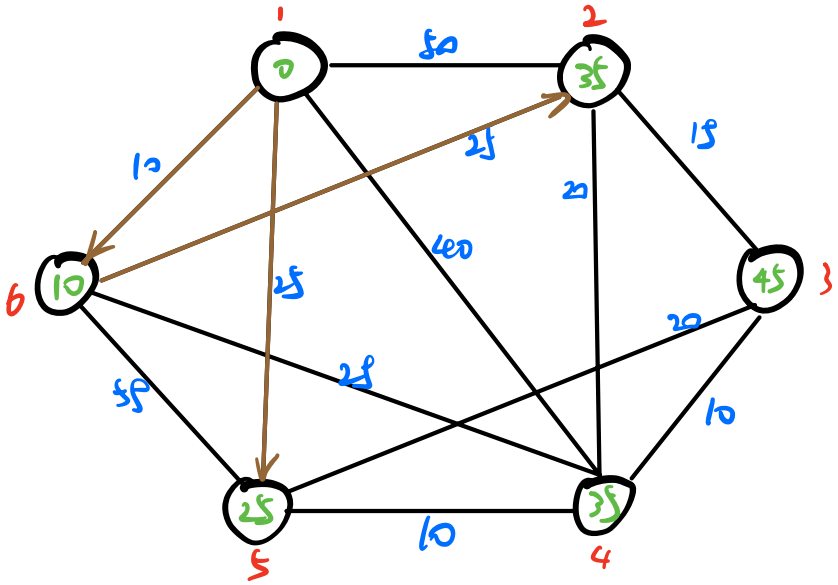
① If x_1 is the start vertex.

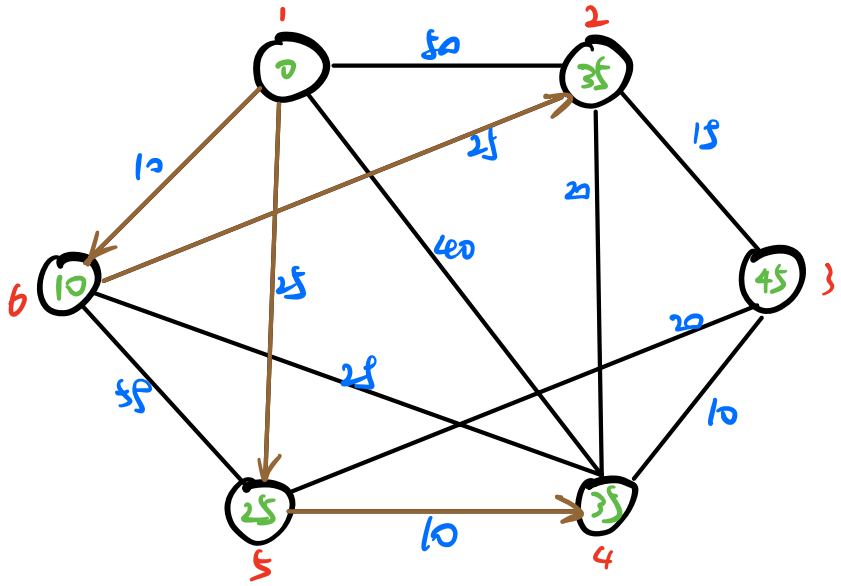
We run the algorithm as below.

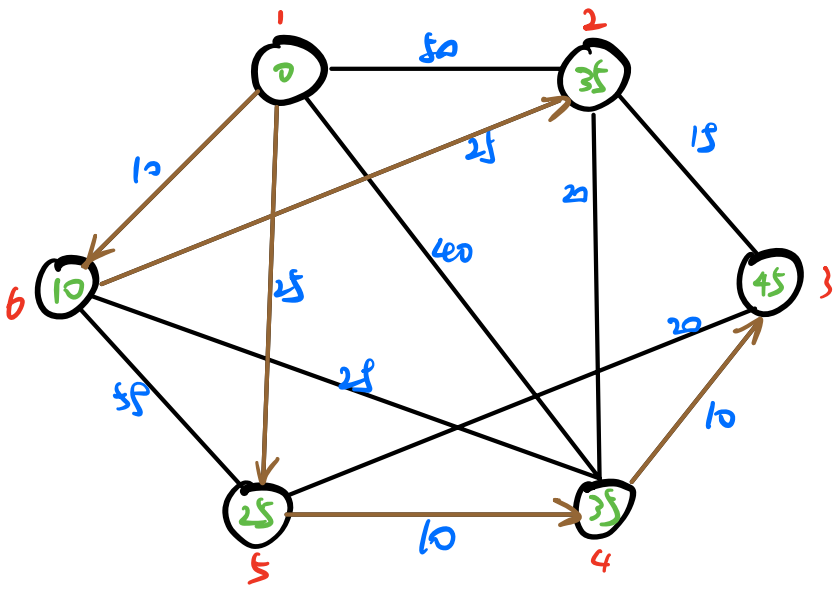












above all,

$d_G(x_1, x_2) = 35$ $P_G(x_1, x_2) = x_1 x_6 x_2$

$d_G(x_1, x_3) = 45$ $P_G(x_1, x_3) = x_1 x_5 x_4 x_3$

$d_G(x_1, x_4) = 35$ $P_G(x_1, x_4) = x_1 x_5 x_4$

$d_G(x_1, x_5) = 25$ $P_G(x_1, x_5) = x_1 x_5$

$d_G(x_1, x_6) = 10$. $P_G(x_1, x_6) = x_1 x_6$

Similarly, run the algorithm when the start vertex is $x_2, \cdots, x_6$, we obtain the distance matrix and route table.

$$\begin{pmatrix} 0 & 35 & 45 & 35 & 25 & 10 \\ 35 & 0 & 15 & 20 & 30 & 25 \\ 45 & 15 & 0 & 10 & 20 & 35 \\ 35 & 40 & 10 & 0 & 10 & 25 \\ 25 & 30 & 20 & 10 & 0 & 35 \\ 10 & 25 & 35 & 45 & 35 & 0 \end{pmatrix}$$

By symetry, we give half of the table.

i	j	$P_G(x_i, x_j)$
1	2	$x_1 x_6 x_2$
1	3	$x_1 x_5 x_4 x_3$
1	4	$x_1 x_5 x_4$
1	5	$x_1 x_5$
1	6	$x_1 x_6$
2	3	$x_2 x_3$
2	4	$x_2 x_4$
2	5	$x_2 x_4 x_5$
2	6	$x_2 x_6$
3	4	$x_3 x_4$
3	5	$x_3 x_5$
3	6	$x_3 x_4 x_6$
4	5	$x_4 x_5$

4 6 $x_4 x_6$

5 6 $x_5 x_1 x_6$

(Optional) **Exercise 5.** The following exercises concern the two conjectures stated in Section 7, and ask for a proof of the special cases in which they are known to hold. A *matching* of a graph $G = (V, E)$ is an edge subset in which no two edges are adjacent; a matching with exactly $\frac{|V(G)|}{2}$ edges is a *perfect matching* of G. A graph is *cubic* if every vertex has degree 3.

> **Conjecture (Berge–Fulkerson, 1971).** Every bridgeless cubic graph admits six perfect matchings such that every edge is covered exactly twice.
>
> **Conjecture (Berge, 1979).** Every bridgeless cubic graph admits five perfect matchings such that every edge belongs to at least one of them.

Prove the following special cases of the above conjectures:

(1) Both conjectures hold for cubic graphs that are 3-edge-colorable.

(2) Both conjectures hold for cubic Hamiltonian graphs (graphs containing a Hamilton cycle). A *Hamilton cycle* of G is a cycle containing every vertex of G.

(3) Do the above conjectures hold for cubic hypohamiltonian graphs? A graph G is *hypohamiltonian* if G itself is not Hamiltonian, but deleting any single vertex from G yields a Hamiltonian graph.

Conjecture (Berge-Fulkerson, 1971) $\Rightarrow$ Conjecture (Berge, 1979).
We just need to prove the first conjecture under the restriction.

(1)

Pf: Suppose G is 3-edge-colorable.
Let M_1, M_2, M_3 be the three color classes.
Each vertex has degree 3, and its 3 incident edges have different colors.
Hence each M_i meets every vertex exactly once.
Thus, M_1, M_2, M_3 are perfect matchings, and they partition $E(G)$.
Consider 6 perfect matchings:
$M_1, M_1, M_2, M_2, M_3, M_3$.
Each edge belongs to exactly one color class, so it occurs exactly twice. □.

(2)

pf: Suppose G is Hamilton.

Let C be a Hamilton cycle of G.

Since $3|V(G)| = 2|E(G)|$,

$|V(G)|$ is even.

Alternating edges of C form two perfect matchings M_1, M_2.

Let $M_3 = E(G) \setminus E(C)$.

Thus M_1, M_2, M_3 are a 3-edge-coloring of G.

By (1), the statement is true.

(3) Not sure.