Laborator 3 - Proiectarea Algoritmilor

METODA GREEDY

1. Algoritmul lui Kruskal.
2. Algoritmul lui Prim.
3. Algoritmul Dijkstra.
4. Algoritmul Ford-Fulkerson.
5. Algoritmul Edmonds-Karp.
6. Fiind dată o hartă cu n tari, se cere o soluţie de colorare a hărţii, utilizând cel mult patru culori, astfel încât două ţări care au frontiera comună să fie colorate diferit.

APLICATII REZOLVATE - METODA GREEDY

MATRICEA COSTURILOR PENTRU UN GRAF PONDERAT (G,c) este definita astfel:

$$A_{i,j} = \begin{cases} 0 & \text{dacă } i = j, \\ \text{costul arcului } (i,j) & \text{dacă există arc de la } i \text{ la } j, \\ \infty & \text{dacă nu există arc de la } i \text{ la } j \end{cases}$$

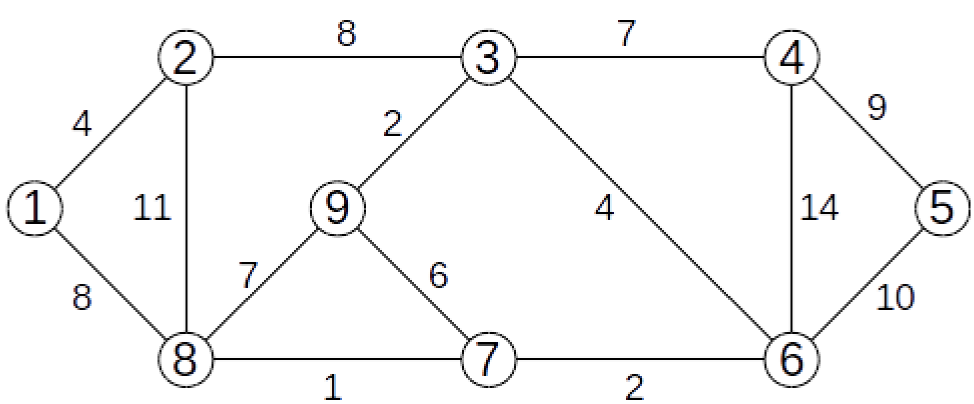
1. *Arbori partiali de cost minim*

APLICATIA 1: Fie graful neorientat ponderat (G, c) reprezentat prin matricea costurilor:

$$A = \begin{pmatrix} 0 & 4 & \infty & \infty & \infty & \infty & \infty & 8 & \infty \\ 4 & 0 & 8 & \infty & \infty & \infty & \infty & 11 & \infty \\ \infty & 8 & 0 & 7 & \infty & 4 & \infty & \infty & 2 \\ \infty & \infty & 7 & 0 & 9 & 14 & \infty & \infty & \infty \\ \infty & \infty & \infty & 9 & 0 & 10 & \infty & \infty & \infty \\ \infty & \infty & 4 & 14 & 10 & 0 & 2 & \infty & \infty \\ \infty & \infty & \infty & \infty & \infty & 2 & 0 & 1 & 6 \\ 8 & 11 & \infty & \infty & \infty & \infty & 1 & 0 & 7 \\ \infty & \infty & 2 & \infty & \infty & \infty & 6 & 7 & 0 \end{pmatrix}$$

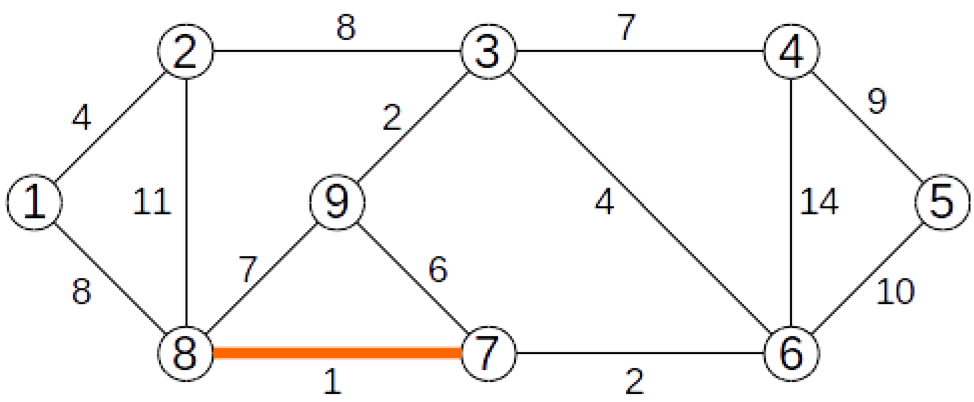
Determinati, folosind *Algoritmul lui Kruskal*, arborele parțial de cost minim pentru acest graf.

SOLUTIE: Reprezentam graful:



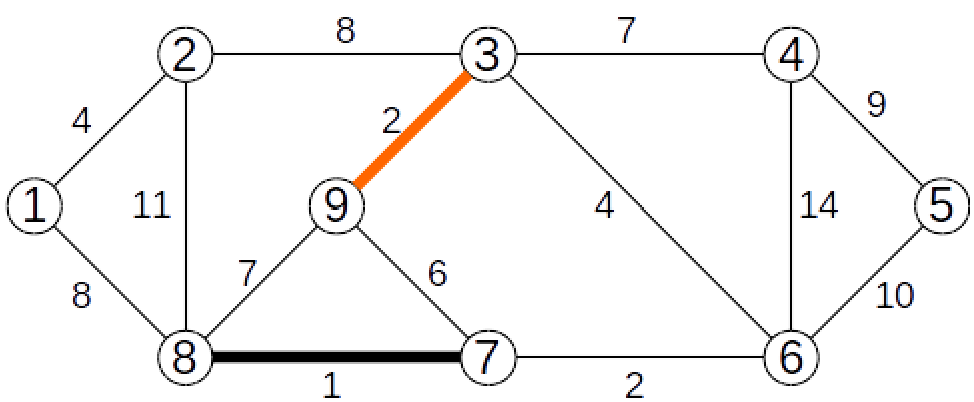
Vom determina, folosind Algoritmul lui Kruskal, arborele parțial de cost minim pentru acest graf. Muchiile se analizeaza în ordine crescatoare. Sunt adaugate doar cele care nu formeaza cicluri cu muchiile deja selectate (cele adaugate sunt reprezentat ingrosat, iar cele care formeaza cicluri se ignora si sunt reprezentate punctat in figuri).

1.



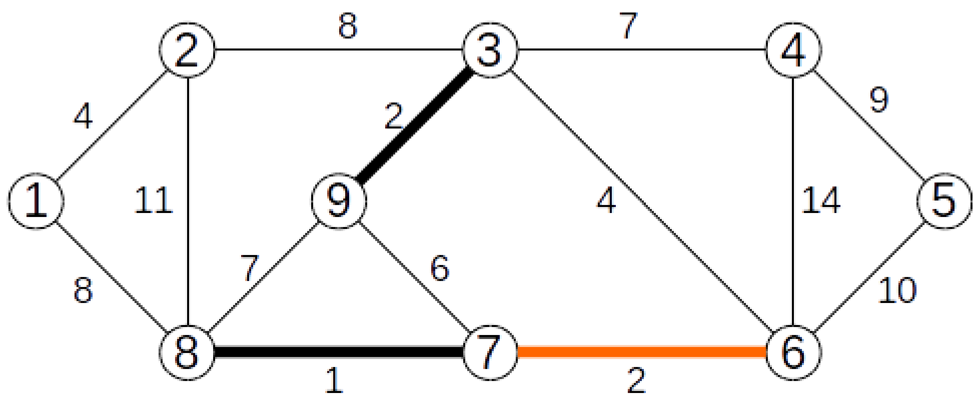
Se adaugă muchia (7,8) de cost 1.

2.



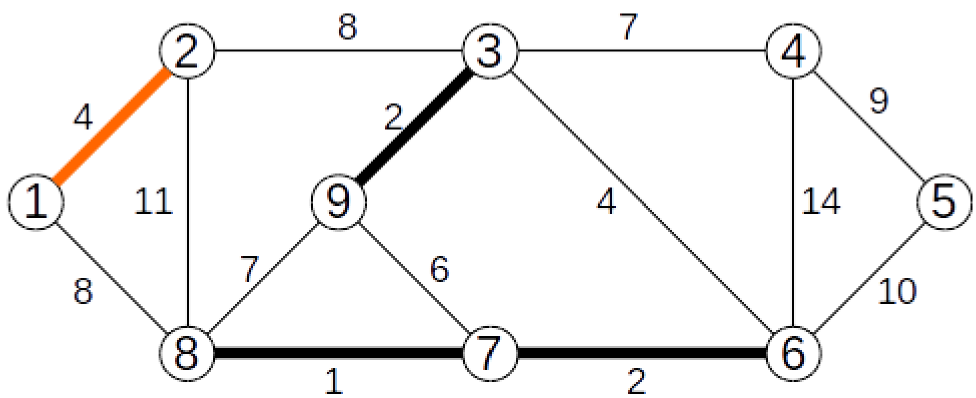
Se adaugă muchia (3,9) de cost 2

3.



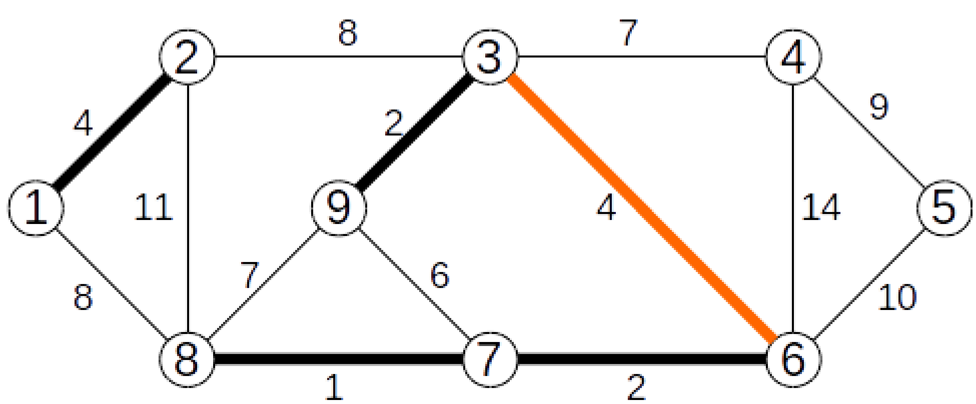
Se adaugă muchia (6,7) de cost 2

4.



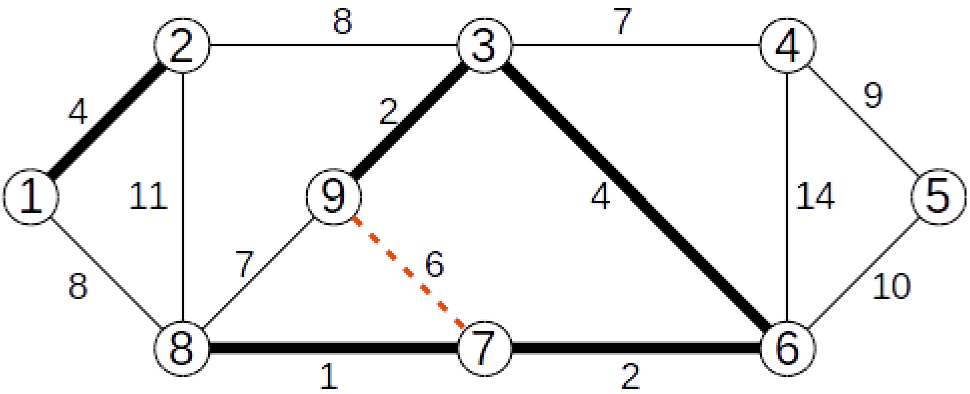
Se adaugă muchia (1,2) de cost 4

5.



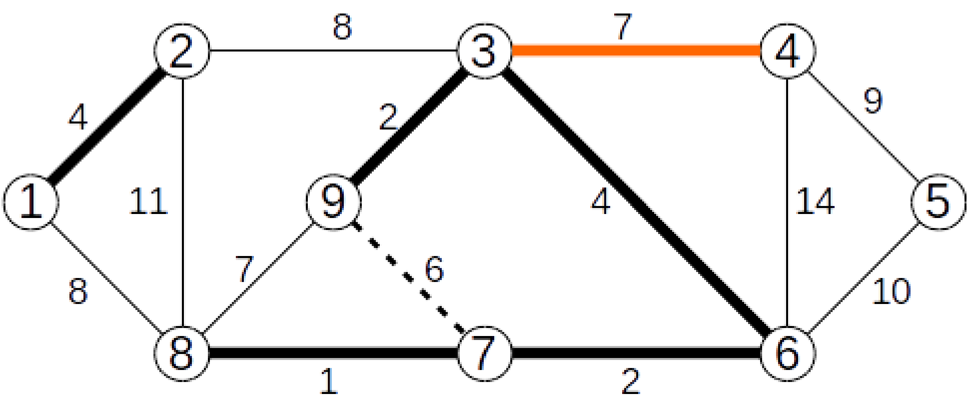
Se adaugă muchia (3,6) de cost 4

6.



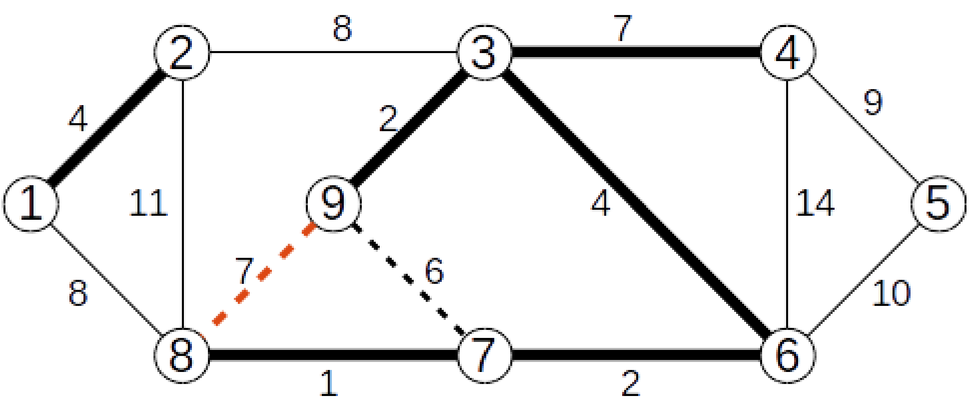
Se ignoră muchia (7,9) de cost 6

7.



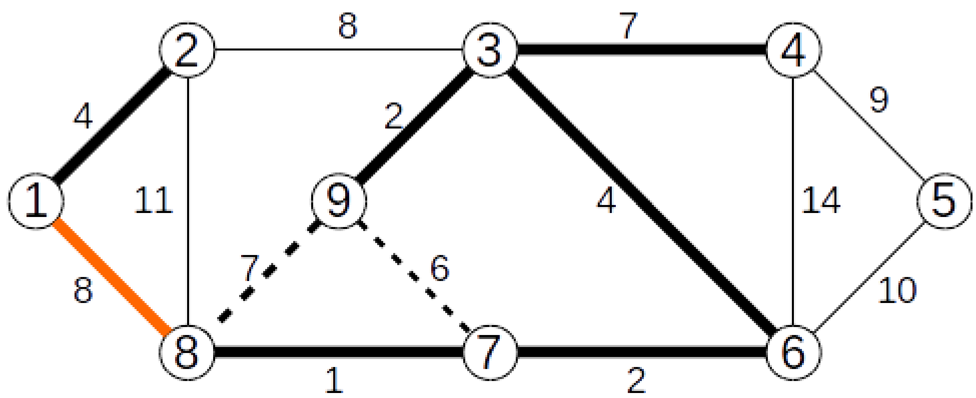
Se adaugă muchia (3,4) de cost 7

8.



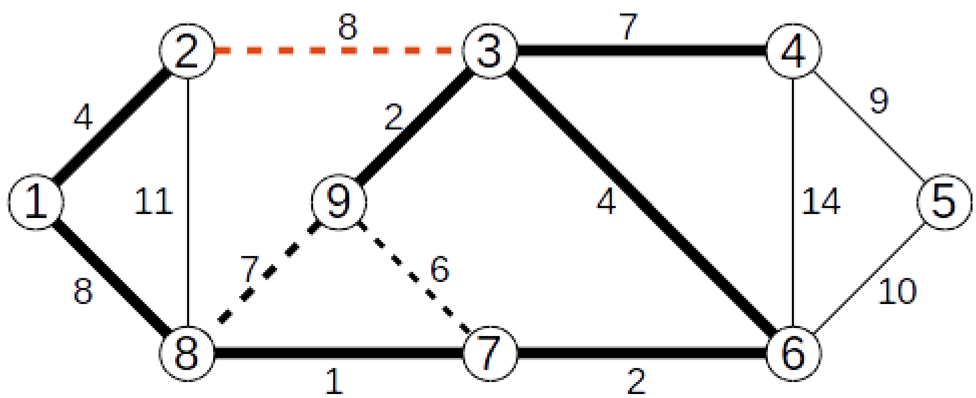
Se ignoră muchia (8,9) de cost 7

9.



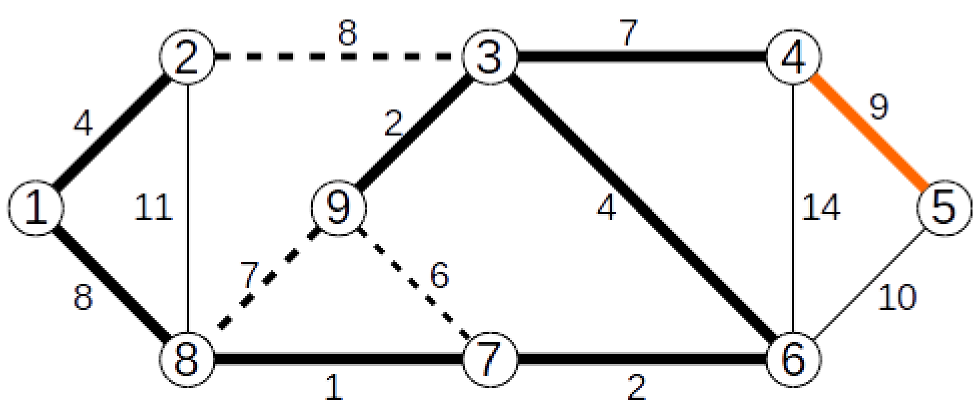
Se adaugă muchia (1,8) de cost 8

10.



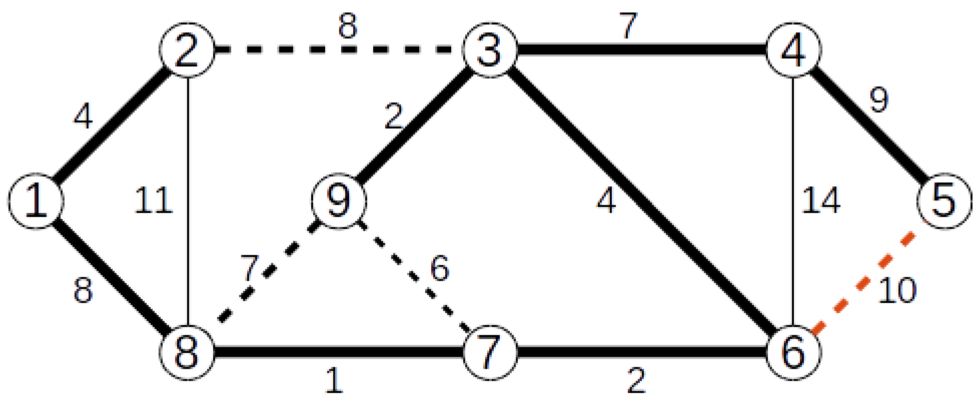
Se ignoră muchia (2,3) de cost 8

11.



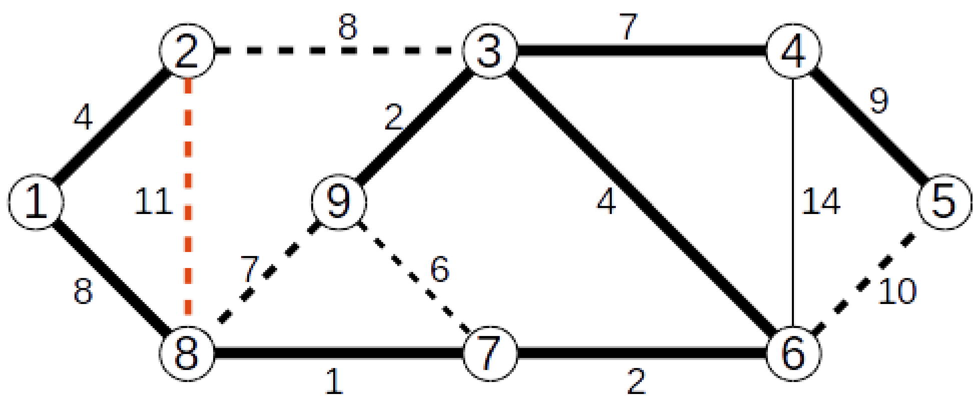
Se adaugă muchia (4,5) de cost 9

12.



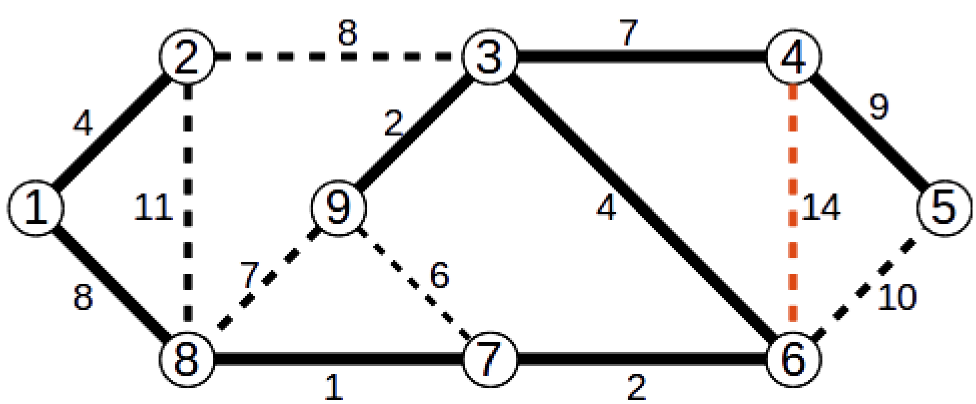
Se ignoră muchia (5,6) de cost 10

13.



Se ignoră muchia (2,8) de cost 11

14.



Se ignoră muchia (4,6) de cost 14.

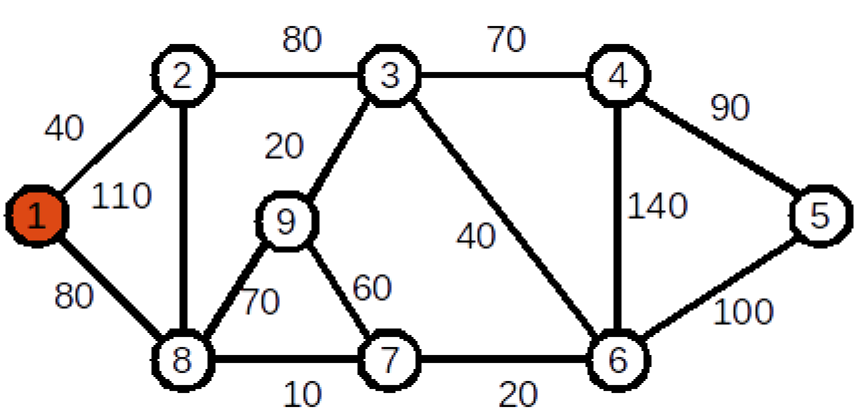
Costul total este 37.

APLICATIA 2: Fie graful neorientat ponderat (G, c) reprezentat prin matricea costurilor:

$$A = \begin{pmatrix} 0 & 40 & \infty & \infty & \infty & \infty & \infty & 80 & \infty \\ 40 & 0 & 80 & \infty & \infty & \infty & \infty & 110 & \infty \\ \infty & 80 & 0 & 70 & \infty & 40 & \infty & \infty & 20 \\ \infty & \infty & 70 & 0 & 90 & 140 & \infty & \infty & \infty \\ \infty & \infty & \infty & 90 & 0 & 100 & \infty & \infty & \infty \\ \infty & \infty & 40 & 140 & 100 & 0 & 20 & \infty & \infty \\ \infty & \infty & \infty & \infty & \infty & 20 & 0 & 10 & 60 \\ 80 & 110 & \infty & \infty & \infty & \infty & 10 & 0 & 70 \\ \infty & \infty & 20 & \infty & \infty & \infty & 60 & 70 & 0 \end{pmatrix}$$

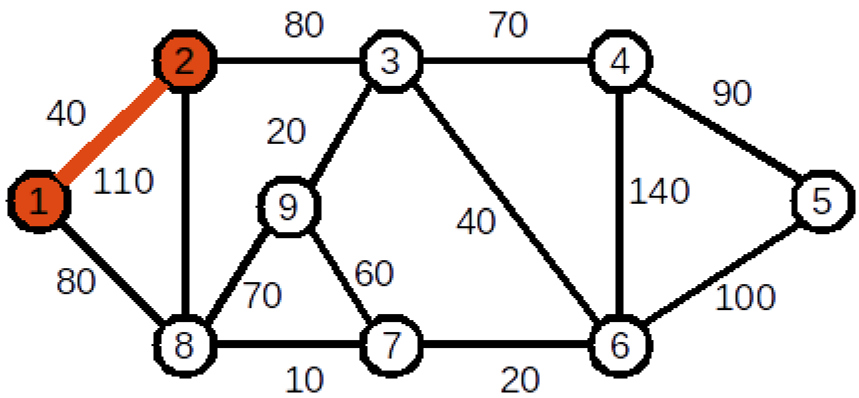
Determinati, folosind *Algoritmul lui Prim*, arborele parțial de cost minim pentru acest graf.

SOLUTIE: Reprezentam graful. Mai jos este descris modul în care se aleg nodurile care se adaugă în arbore pentru acest graf ponderat. Selectam de fiecare data o muchie de cost minim care leaga un nod selectat de altul neselectat care devine acum selectat.



Nodul inițial este 1.

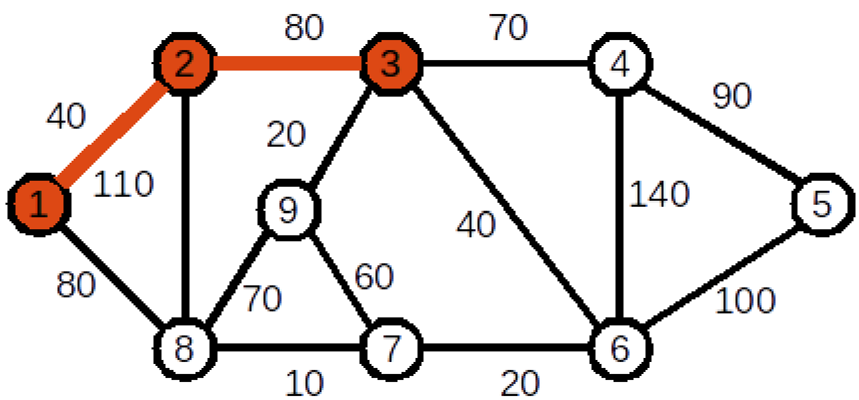
Costul curent al APM este 0



Se adaugă nodul 2.

Muchia folosită este (1,2).

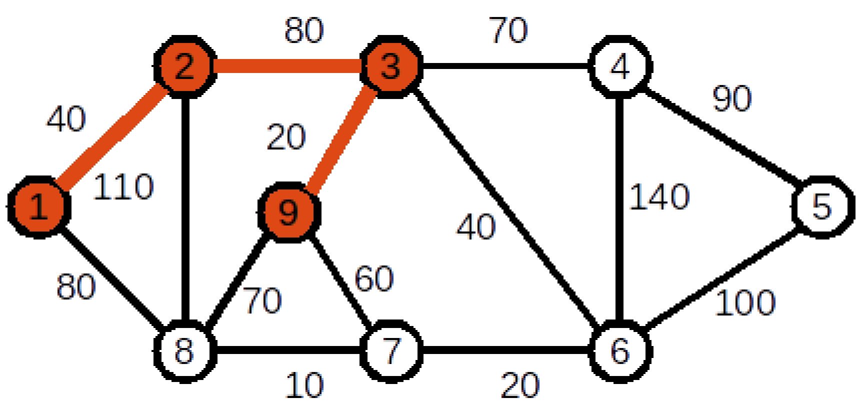
Costul curent al APM este 40



Se adaugă nodul 3.

Muchia folosită este (2,3).

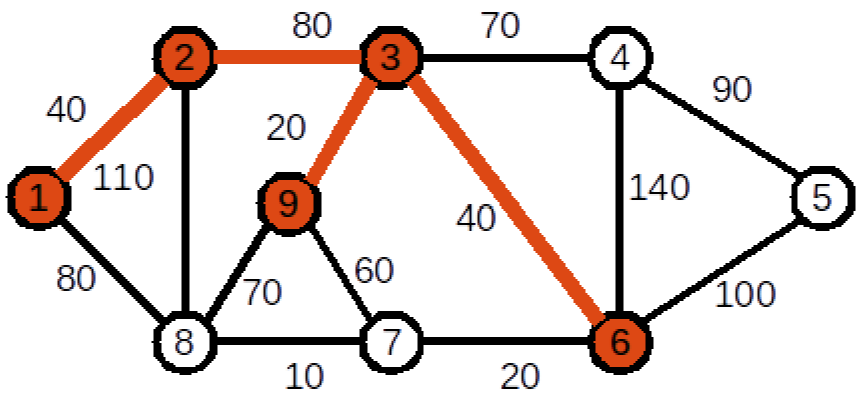
Costul curent al APM este 120



Se adaugă nodul 9.

Muchia folosită este (3,9).

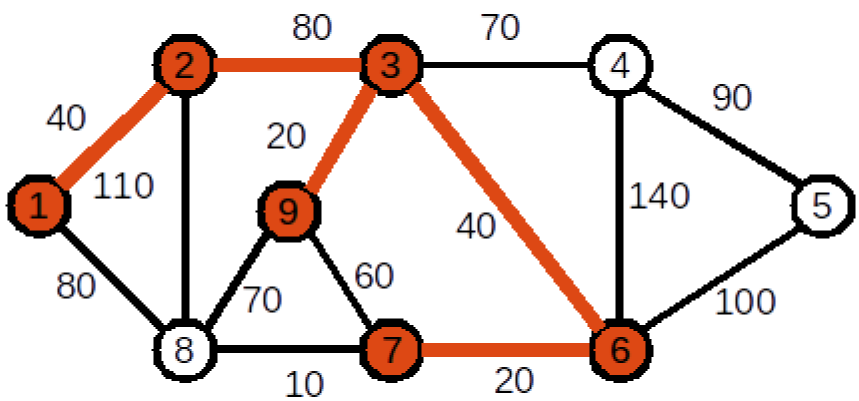
Costul curent al APM este 140



Se adaugă nodul 6.

Muchia folosită este (3,6).

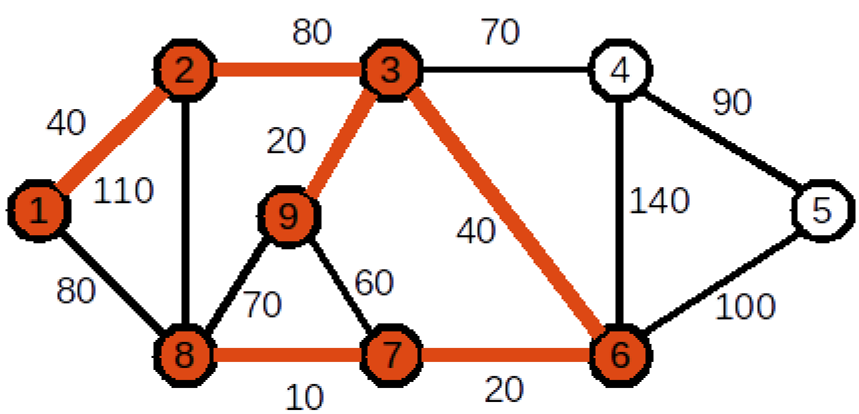
Costul curent al APM este 180



Se adaugă nodul 7.

Muchia folosită este (6,7).

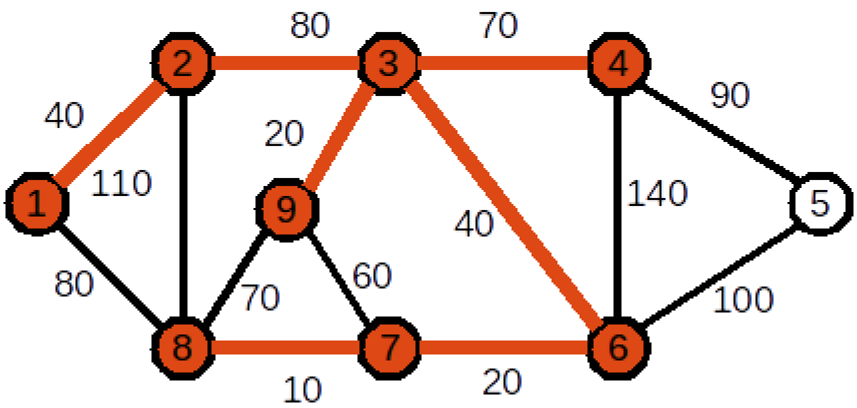
Costul curent al APM este 200



Se adaugă nodul 8.

Muchia folosită este (7,8).

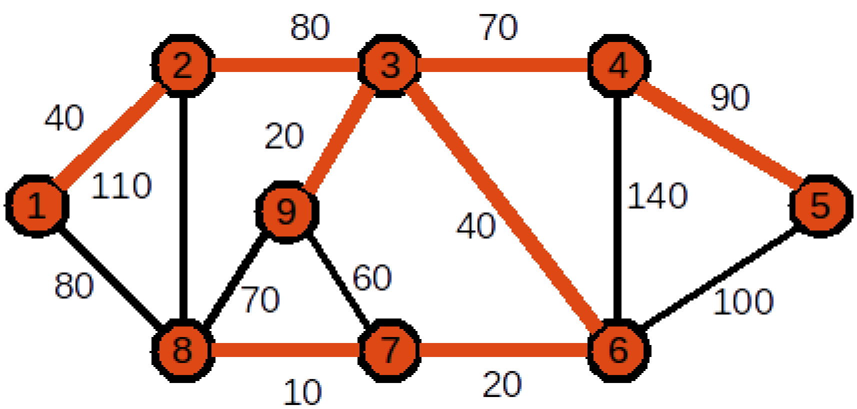
Costul curent al APM este 210



Se adaugă nodul 4.

Muchia folosită este (3,4).

Costul curent al APM este 280



Se adaugă nodul 5.

Muchia folosită este (4,5).

Costul curent al APM este 370.

2. *Distante si drumuri minime.*

<u>APLICATIE:</u> Pentru graful orientat ponderat (G, c) reprezentat prin matricea costurilor:

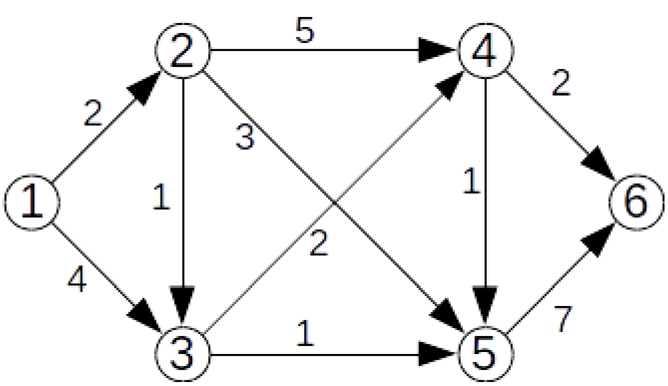
$$A = \begin{pmatrix} 0 & 2 & 4 & \infty & \infty & \infty \\ \infty & 0 & 1 & 5 & 3 & \infty \\ \infty & \infty & 0 & 2 & 1 & \infty \\ \infty & \infty & \infty & 0 & 1 & 2 \\ \infty & \infty & \infty & \infty & 0 & 7 \\ \infty & \infty & \infty & \infty & \infty & 0 \end{pmatrix}$$

luand ca nod sursa nodul s = 1, aplicati *Algoritmului Dijkstra*.

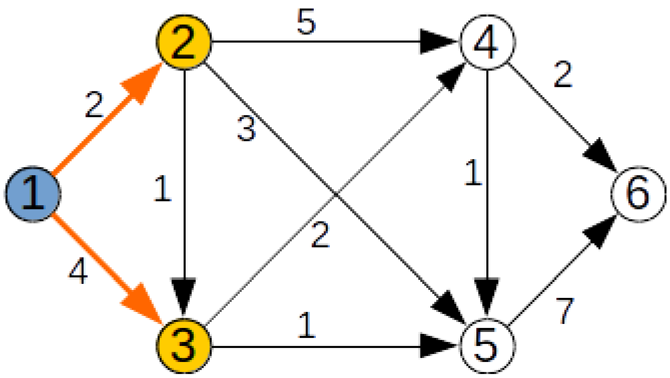
<u>SOLUTIE:</u> Reprezentam graful. Folosim următoarele structuri de date:

- un vector t[], în care t[k] reprezintă costul minim curent al drumului de la nodul sursă s=1 la k;
- un vector caracteristic S[], în care S[k]=1 dacă pentru nodul k s-a determinat costul minim final, respectiv S[k]=0 dacă pentru nodul k nu s-a determinat (încă) acest cost;
- Pentru determinarea drumurilor minime de la nodul s la nodurile grafului vom utiliza si un vector T AT A[] avand semnificatia T AT A[k] = nodul j ce este predecesorul direct al nodului k pe drumul minim de la s la k, $\forall$ k $\in$ {1, . . . , n}.

Graful dat este:

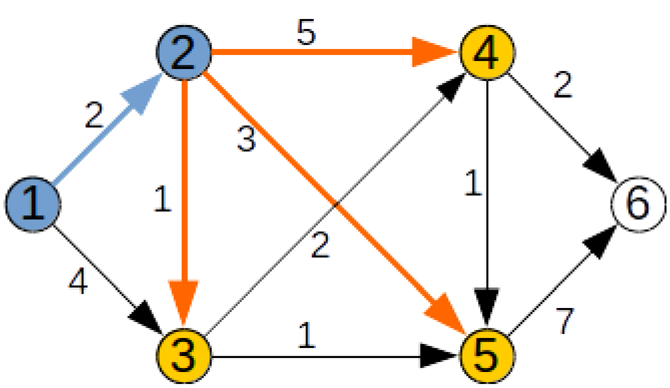


Pasul 0: Initializăm vectorii, ca mai jos. Inițial în mulțimea S se află doar nodul sursă s=1.



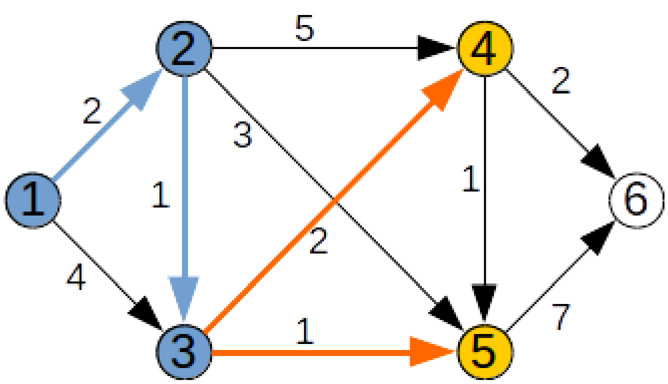
k	1	2	3	4	5	6
S[k]	1	0	0	0	0	0
t[k]	0	2	4	∞	∞	∞
TATA[k]	0	∞	∞	∞	∞	∞

Pasul 1: Alegem un vârf k din afara lui S, pentru care t[k] este finit și minim. Acesta este k=2. Îl adăugăm în S și analizăm nodurile x pentru care (k,x) este arc. Se vor relaxa nodurile 3 4 5, adica pentru succesorii nodului k reactualizam vectorul t.



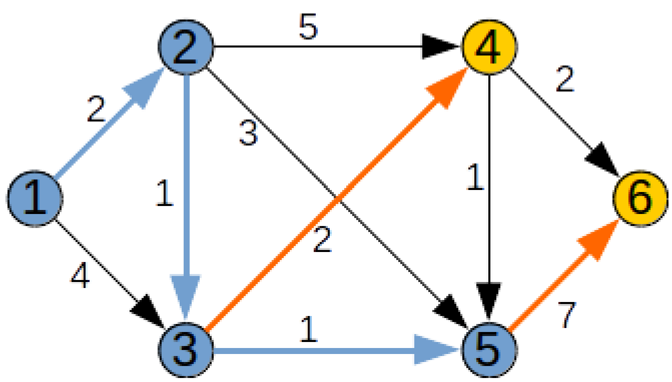
k	1	2	3	4	5	6
S[k]	1	1	0	0	0	0
t[k]	0	2	3	7	5	∞
TATA[k]	0	1	∞	∞	∞	∞

Pasul 2: Alegem un vârf k din afara lui S, pentru care d[k] este finit şi minim. Acesta este k=3. Îl adăugăm în S şi analizăm nodurile x pentru care (k,x) este arc. Se vor relaxa nodurile 4 5.



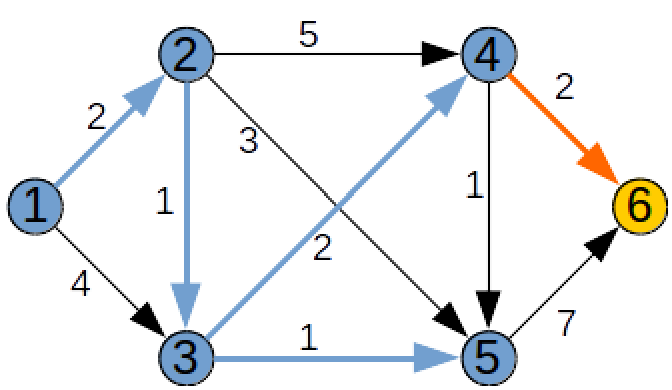
k	1	2	3	4	5	6
S[k]	1	1	0	0	0	0
t[k]	0	2	3	5	4	∞
TATA[k]	0	1	2	∞	∞	∞

Pasul 3: Alegem un vârf k din afara lui S, pentru care t[k] este finit şi minim. Acesta este k=5. Îl adăugăm în S şi analizăm nodurile x pentru care (k,x) este arc. Se va relaxa nodul 6.



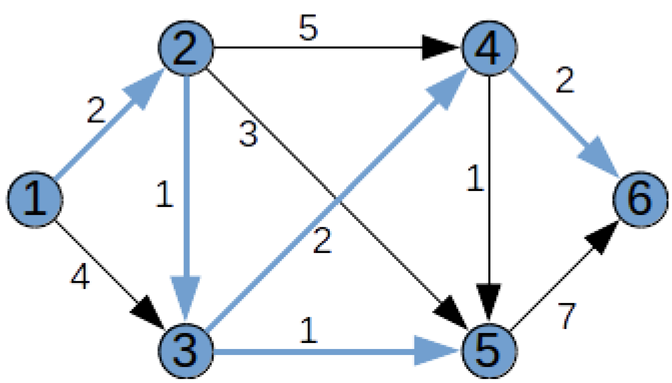
k	1	2	3	4	5	6
S[k]	1	1	0	0	0	0
t[k]	0	2	3	5	4	11
TATA[k]	0	1	2	∞	3	∞

Pasul 4: Alegem un vârf k din afara lui S, pentru care t[k] este finit și minim. Acesta este k=4. Îl adăugăm în S și analizăm nodurile x pentru care (k,x) este arc. Se va relaxa nodul 6.



k	1	2	3	4	5	6
S[k]	1	1	0	0	0	0
t[k]	0	2	3	5	4	7
TATA[k]	0	1	2	3	3	∞

Pasul 5: Alegem un vârf k din afara lui S, pentru care t[k] este finit și minim. Acesta este k=6. Îl adăugăm în S și analizăm nodurile x pentru care (k,x) este arc. Nu mai există asemenea arce, niciun nod nu se mai relaxează.



k	1	2	3	4	5	6
S[k]	1	1	0	0	0	0
t[k]	0	2	3	5	4	7
TATA[k]	0	1	2	3	3	4

Algoritmul lui Dijkstra s-a încheiat. Valorile finale din vectorul t[] – distanțele minime de la nodul s=1 la toate celelalte sunt cele de mai sus.

Drumurile minime se gasesc pentru fiecare nod mergand inapoi de-a lungul vectorului TATA[ ] pana ajungem in nodul sursa.

De exemplu pt nodul x=6 : 6-4-3-2-1, deci drumul minim determinat de algoritm de la 1 la 6 este [1,2,3,4,6]. Analog, drumurile minime determinate de algoritm sunt:

- de la 1 la 1 : [1]
- de la 1 la 2: [1,2]
- de la 1 la 3: [1,2,3]
- de la 1 la 4: [1,2,3,4]
- de la 1 la 5: [1,2,3,5]

<u>APLICATII PROPUSE:</u>

Determinaţi un arbore parţial de cost minim pentru graful ponderat din Figura 1, aplicând:

- Algoritmul lui Kruskal;
- Algoritmul lui Prim.

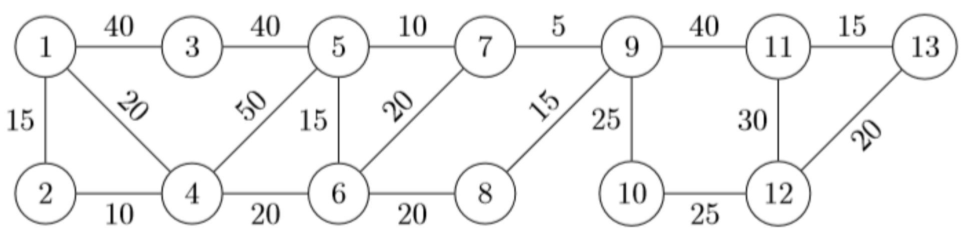


Figura 1:

Pentru graful neorientat ponderat din Figura 2, determinaţi distanţele minime şi drumurile minime de la nodul sursă 1 la fiecare dintre nodurile grafului. Aceeaşi cerinţă dacă nodul sursă este 10.

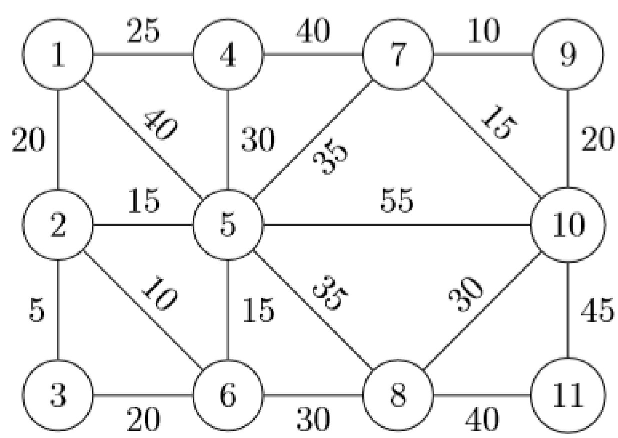


Figura 2: