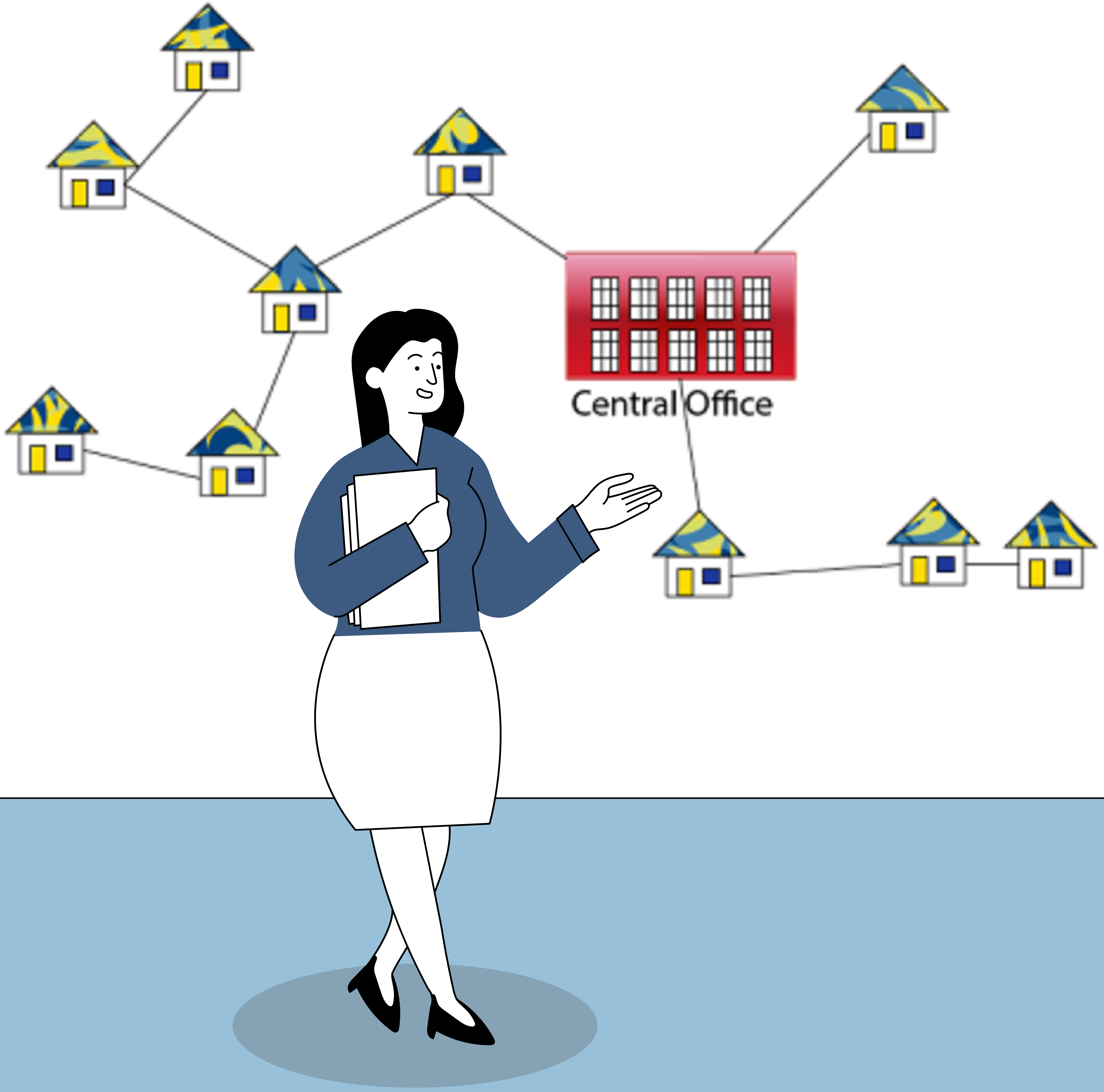
The minimal spanning tree problem

Group 1

GROUP'S MEMBER:

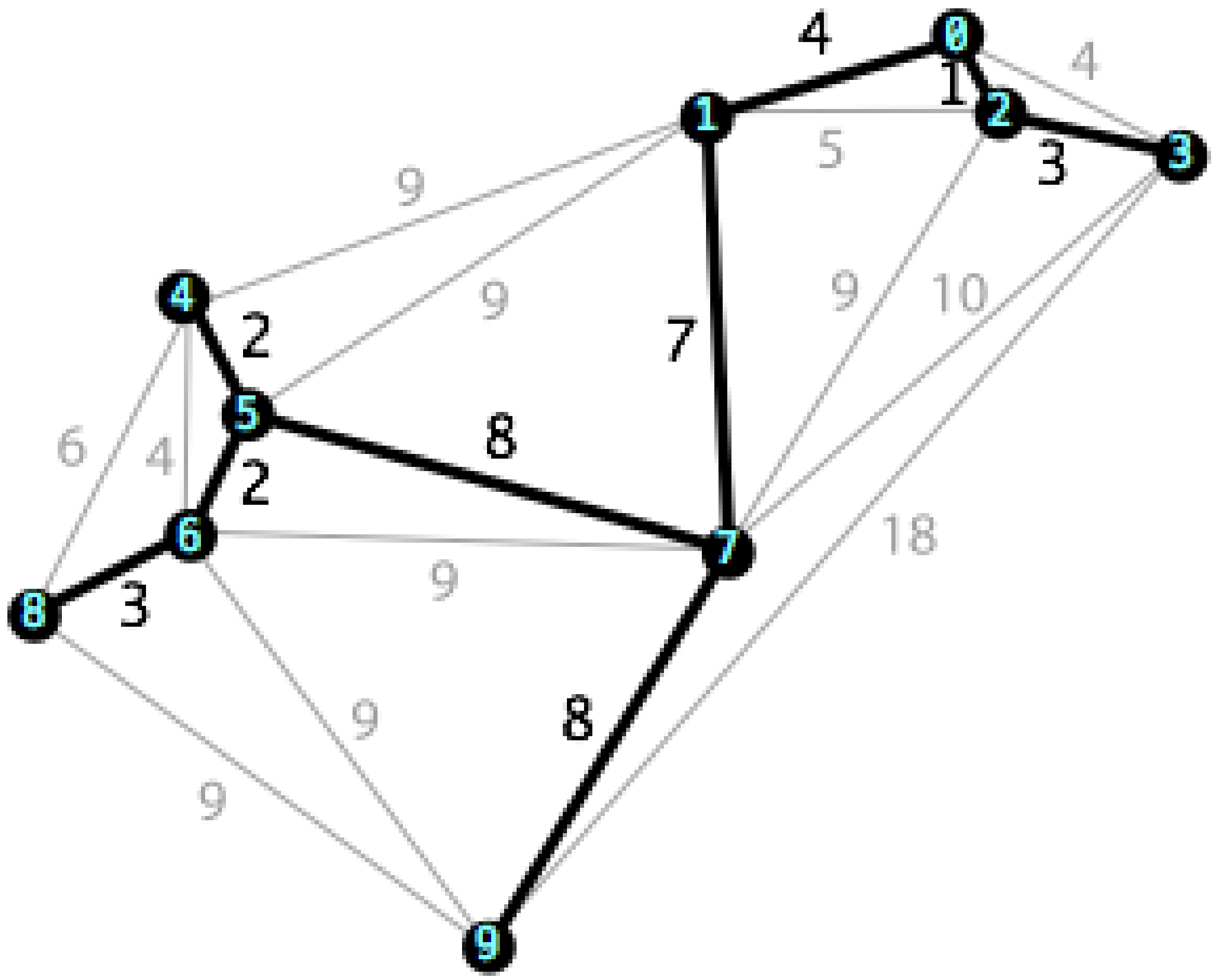
1. Trần Châu Thạnh An - MAMAIU19001
2. Trần Hoàng Duy - MAMAIU19005
3. Trần Việt Hằng - MAMAIU18079
4. Trần Thanh Hiếu - MAMAIU19006
5. Lê Minh Hoàng - MAMAIU19051

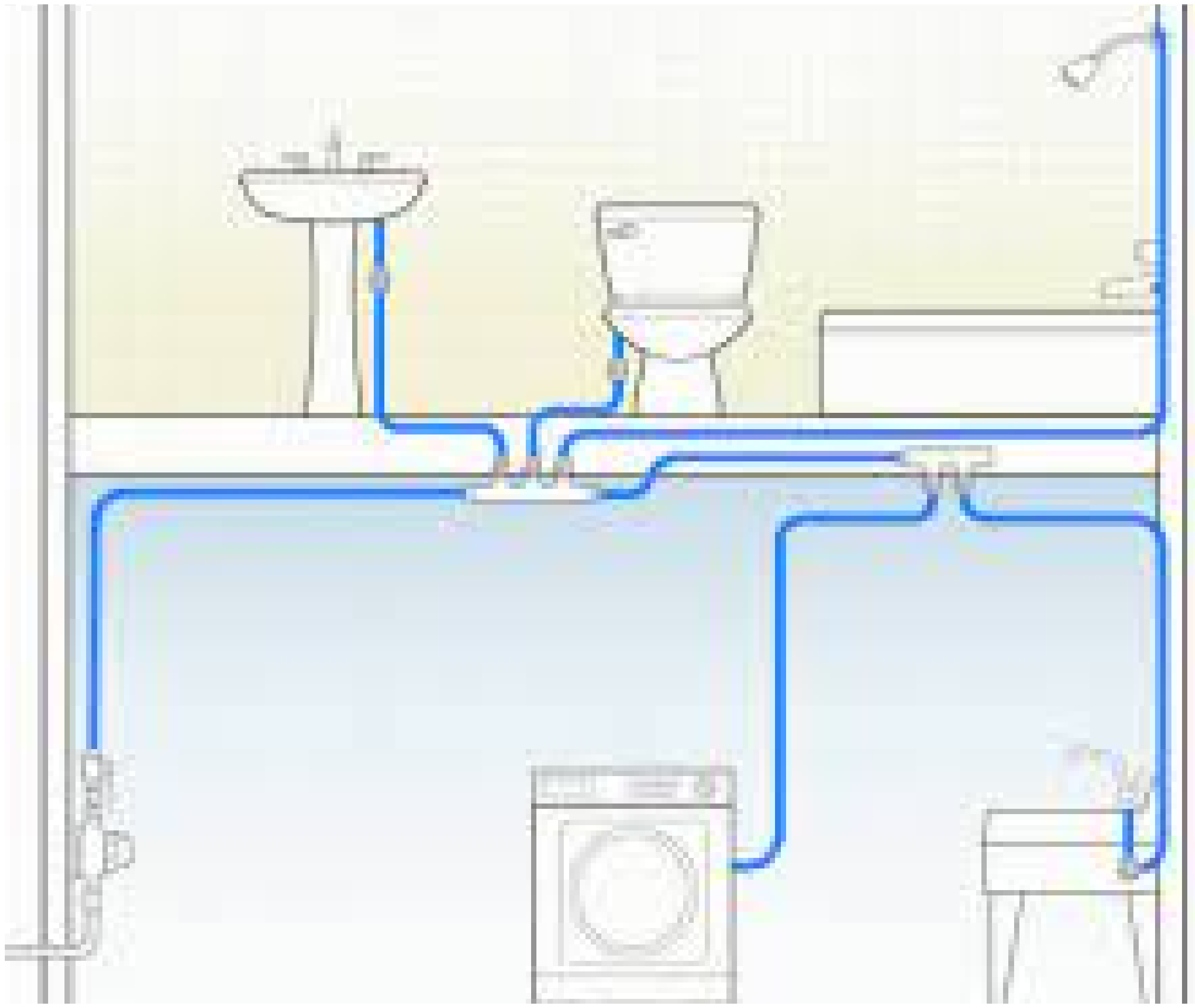
1. DEFINITION AND APPLICATION OF MINIMAL SPANNING TREE PROBLEM:



A minimum spanning tree (MST):

- A subset of the edges of a connected, edge-weighted, undirected graph.
- Connects all the vertices together, without any cycles.
- With the minimum possible total arcs length.

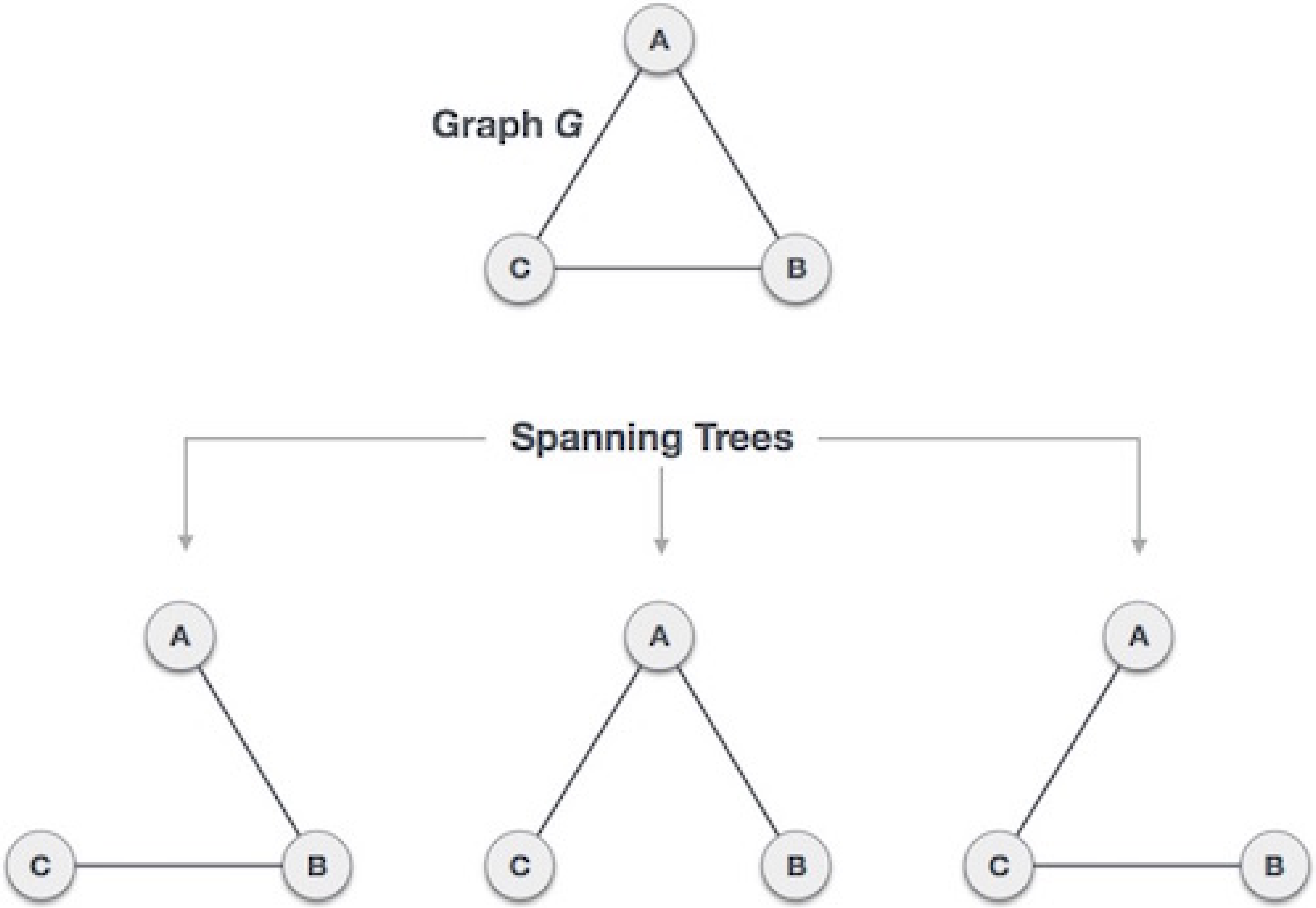




MST problems in real life:

- Finding the shortest way to connect all the warehouses.
- Routing protocol goal.
- Supply water to all the equipments in the house

Finding the spanning tree



-Consider a network G : G = {N, A} with n nodes and a arcs called potential links.

-Let (i;j) be an arc ∈ A, associated with a length Xij>0.

-Our goal is to find a spanning tree T=(N, A'), a connected acycle subnetwork linking all the nodes in G with the minimum length.

$$L(T)=\sum_{(i;j)\in A'} Xij$$

with L(T) is the total length of the tree

2. PRIM'S ALGORITHM

2. PRIM'S ALGORITHM

Denote G = {N, A} the initial network

- Step 1: Pick any node i in the network and set up an initial tree $\mathcal{T}_0 = (\mathcal{N}', \mathcal{A}') = (\{i\}, \emptyset)$.
- Step 2: If $\mathcal{N}' = \mathcal{N}$, the algorithm stops with the optimal total length $\ell(\mathcal{T})$. Otherwise, go to step 3.
- Step 3: Choose any pair of nodes $j_0 \in \mathcal{N} \setminus \mathcal{N}'$ and $i_0 \in \mathcal{N}'$ that satisfy

$$x_{i_0 j_0} = \min\{x_{ij} : (i,j) \in \mathcal{A}, i \in \mathcal{N}', j \in \mathcal{N} \setminus \mathcal{N}'\}.$$

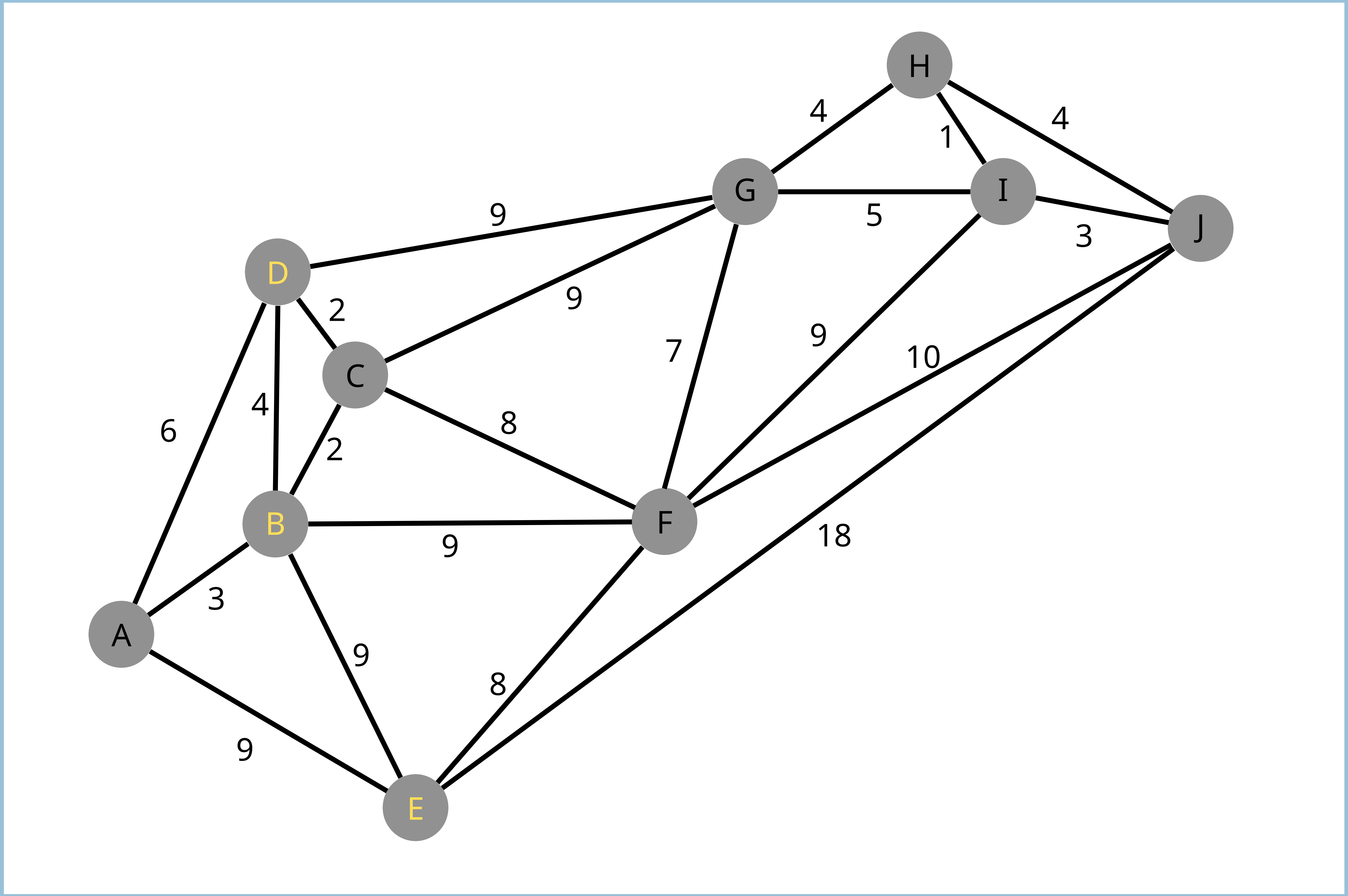
  If there are two or more satisfied pairs of nodes, then choose any one of them.

- Step 4: Update the new tree

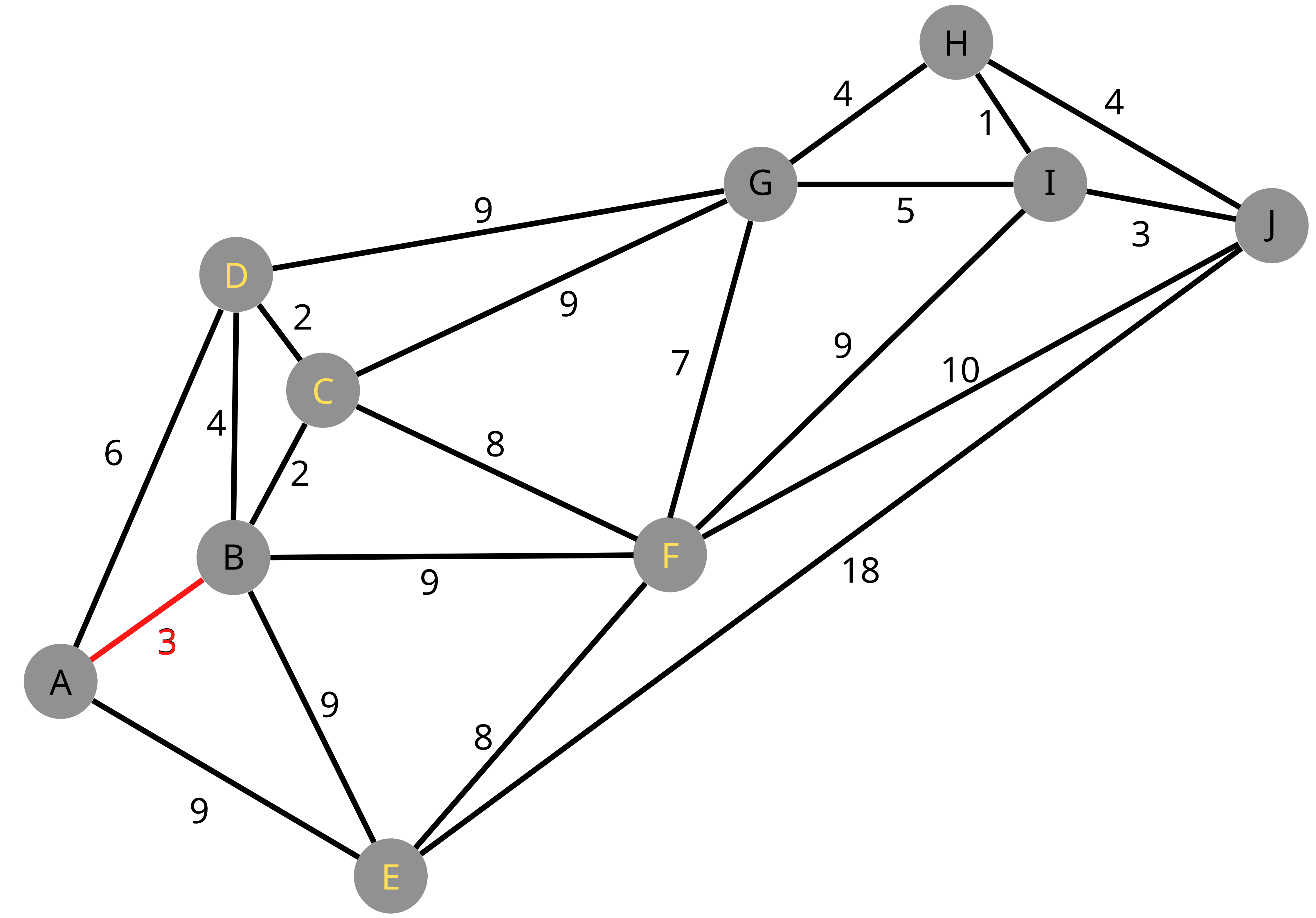
$$\overline{\mathcal{T}} = (\overline{\mathcal{N}'} = \mathcal{N}' \cup \{j_0\}, \overline{\mathcal{A}'} = \mathcal{A}' \cup \{(i_0, j_0)\}).$$

  Return to step 2.

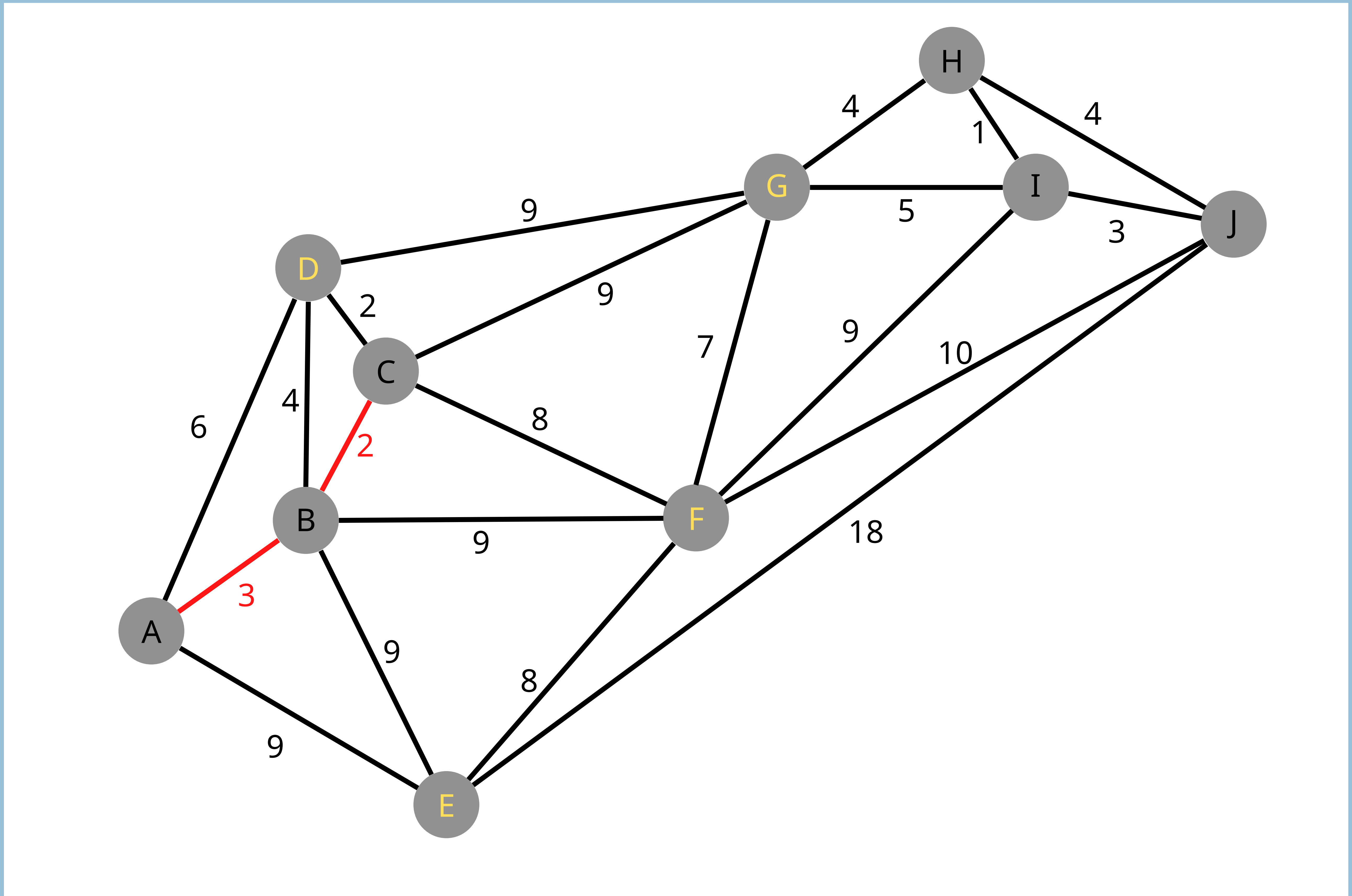
3. SOLVING THE PROBLEM



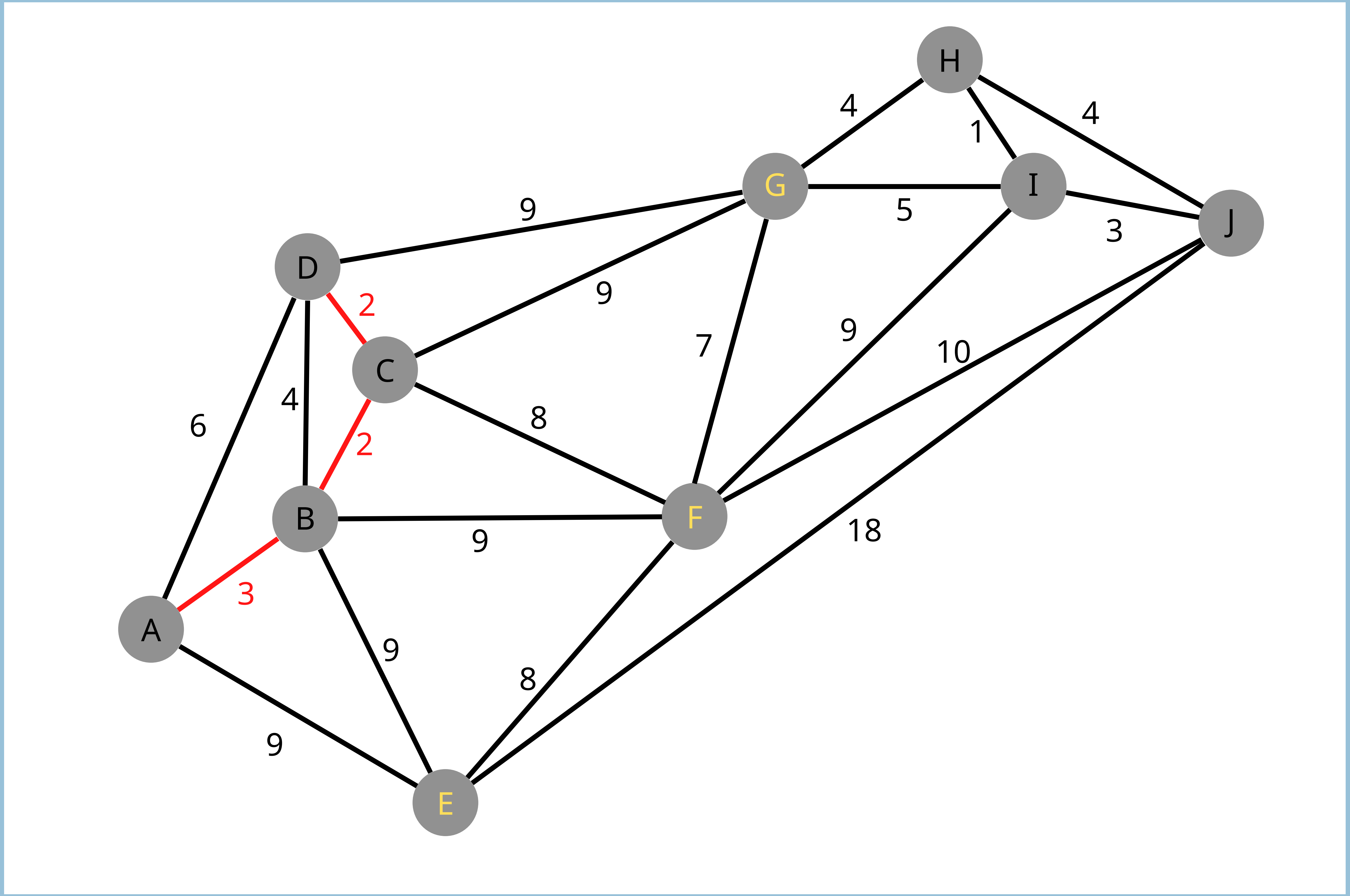
Initial tree: $T_0 = (\{A\}, \emptyset)$



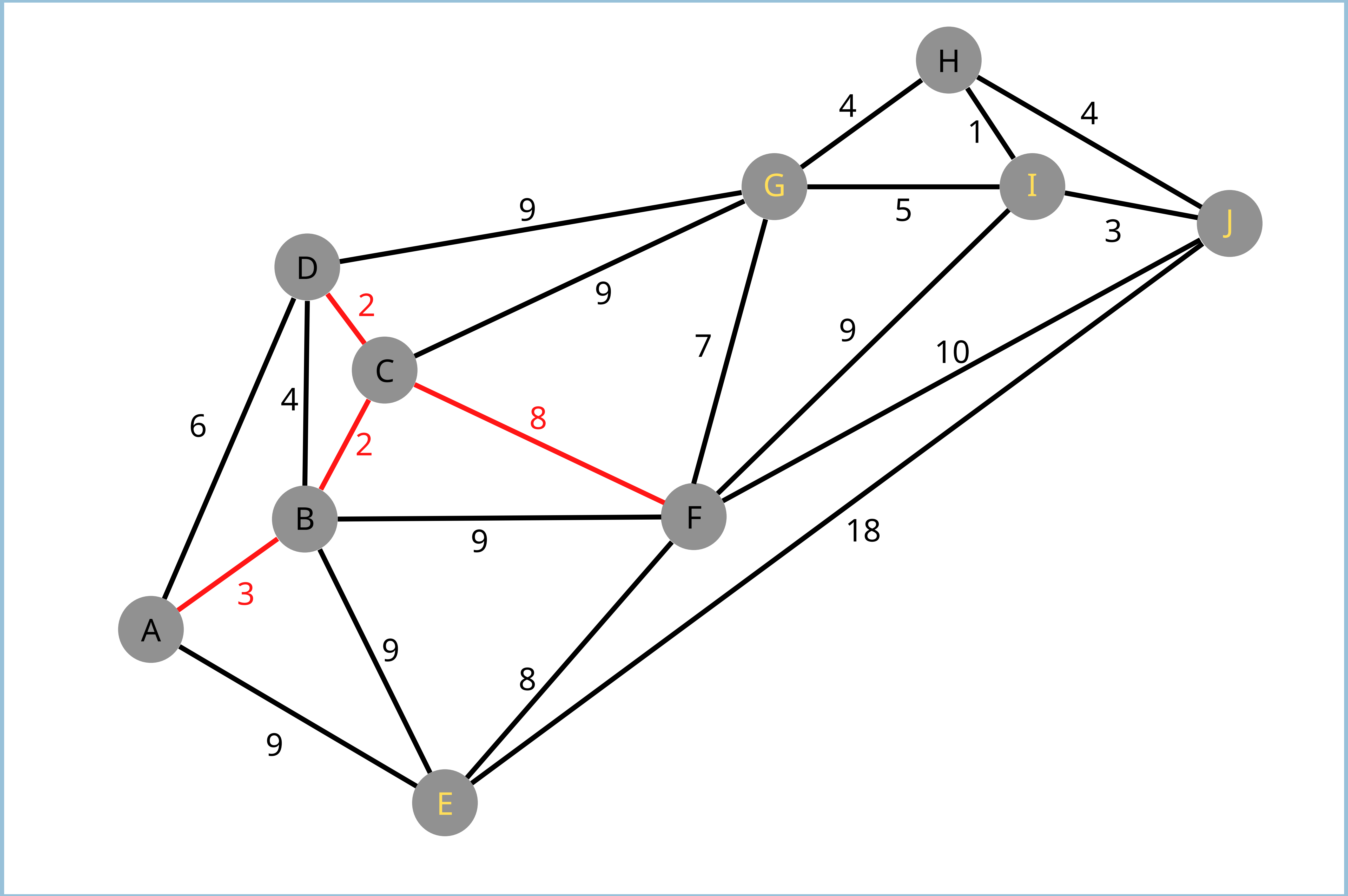
Iteration 1: T_1 = ({A,B} , { (A,B) }).



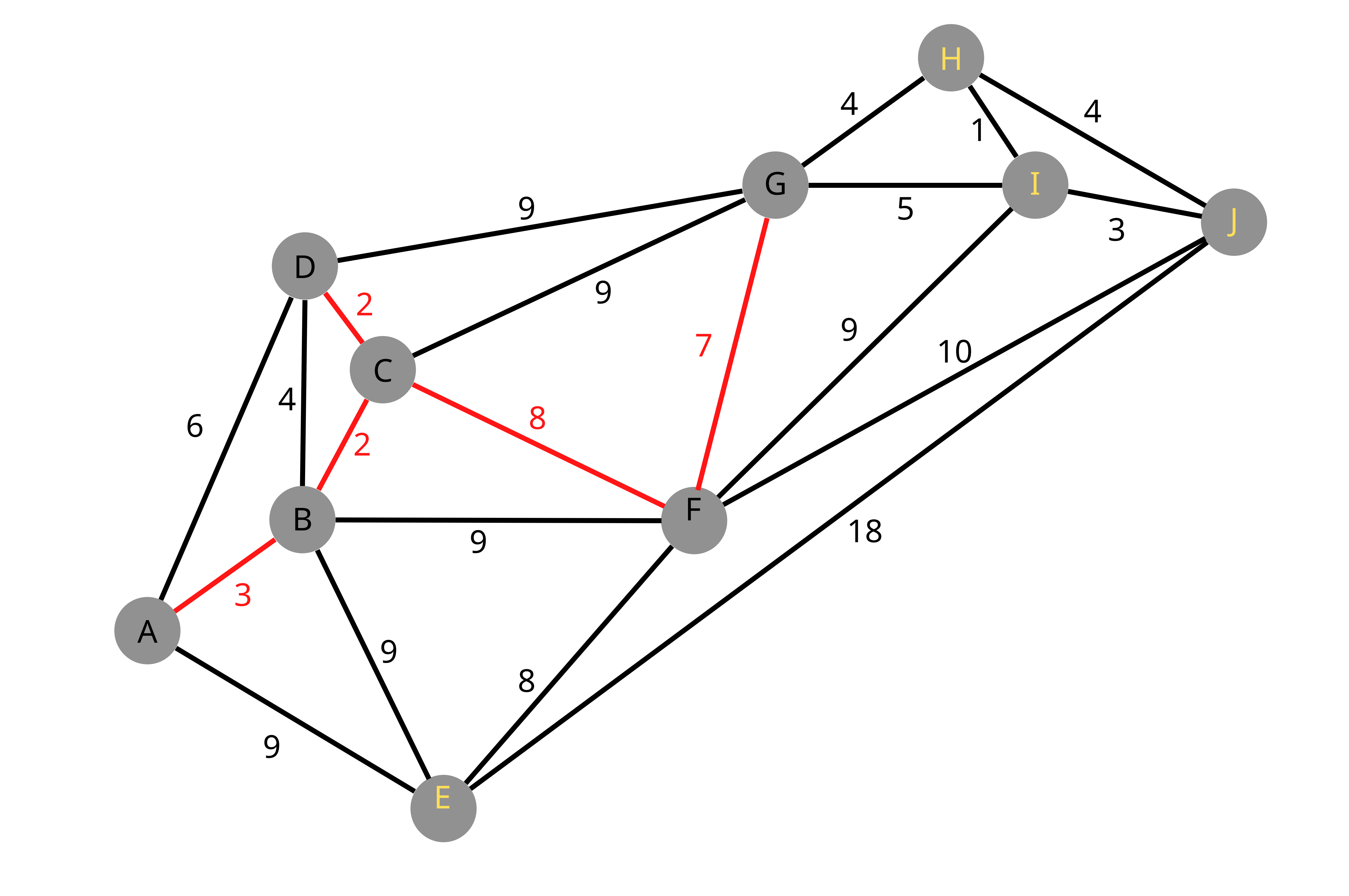
Iteration 2: $T_2 = (\ \{A,B,C\}\ ,\ \{\ (A,B)\ ,\ (B,C)\ \}\ ).$



Iteration 3: $T_3 = (N_3 = \{A,B,C,D\} ; A_3 = \{ (A,B) , (B,C) , (C,D) \})$.

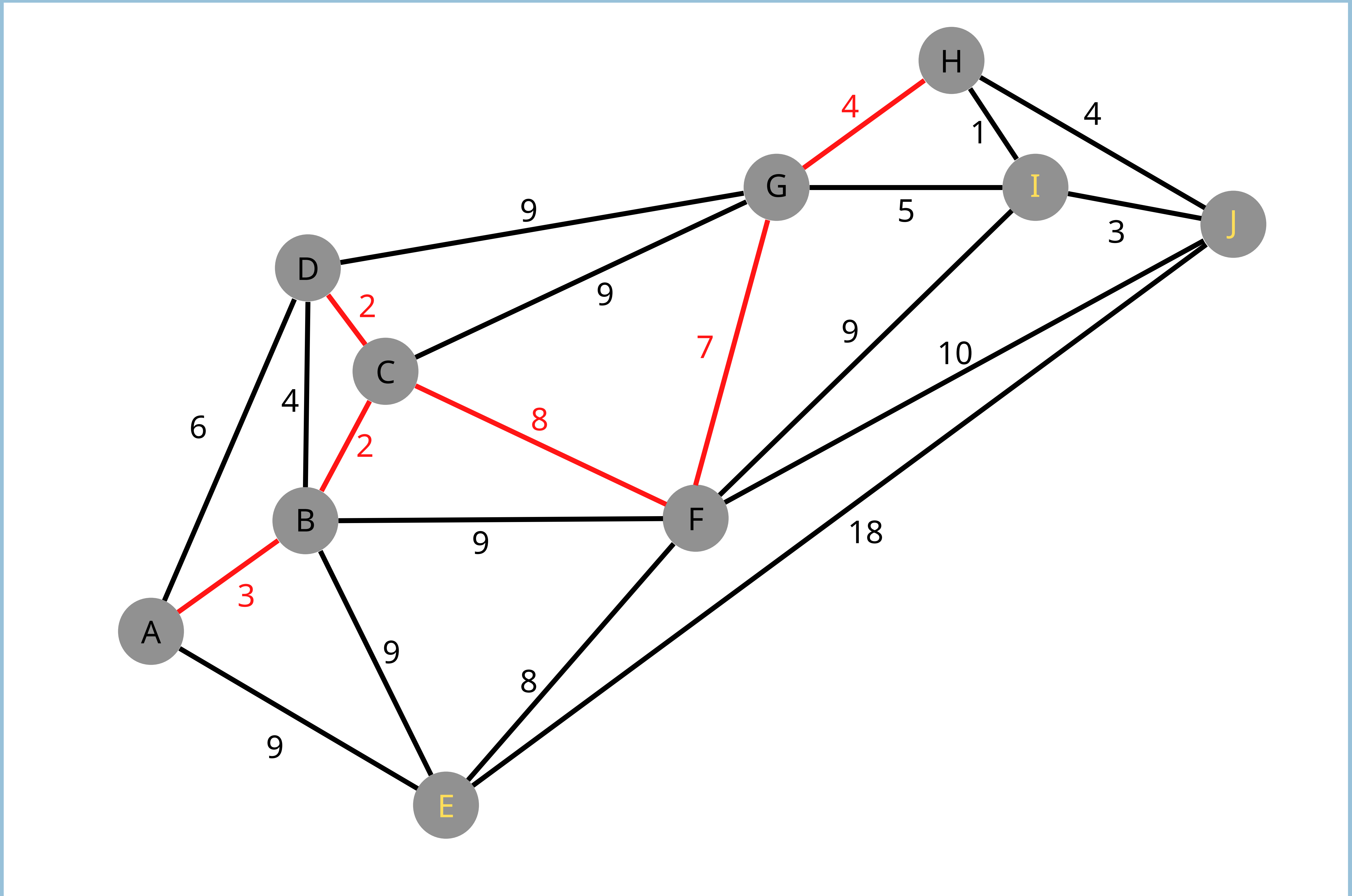


Iteration 4: T_4 = (N_4 = {A,B,C,D,F} ; A_4 = { (A,B) , (B,C) , (C,D) , (C,F) }).

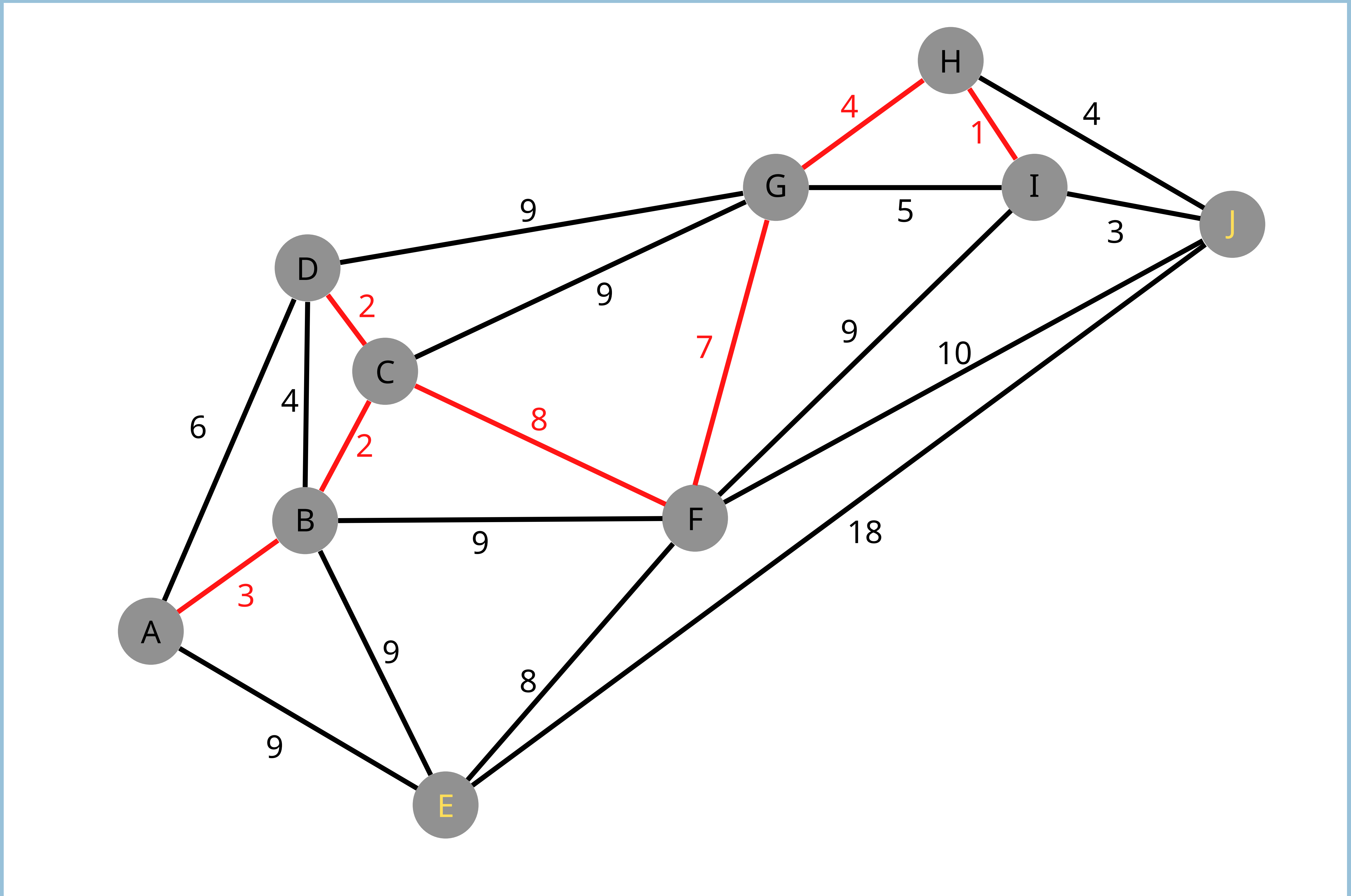


Iteration 5: T_5 = (N_5 = {A,B,C,D,F,G} ;

A_5 = { (A,B) , (B,C) , (C,D) , (C,F) , (F,G) }).

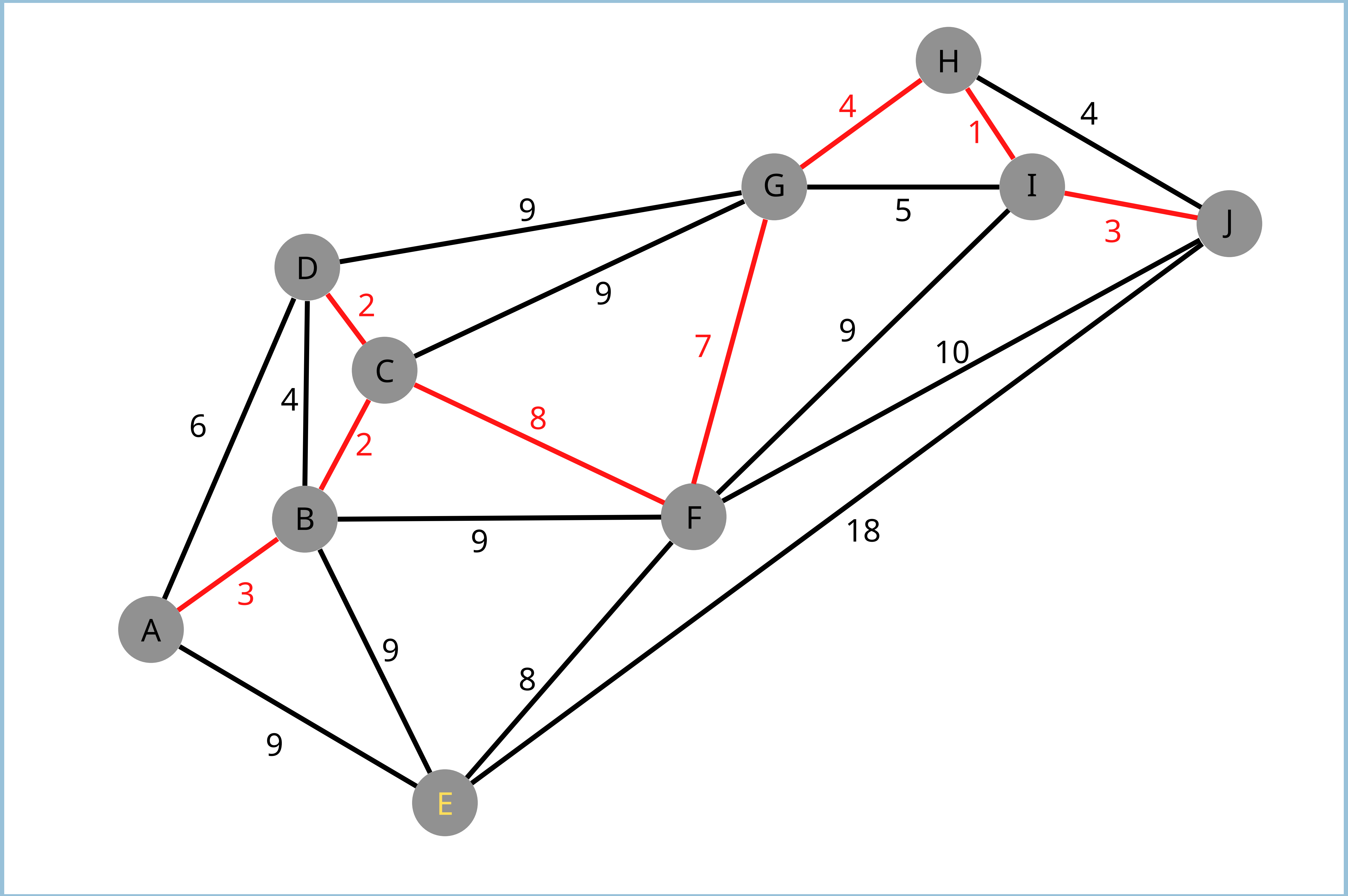


Iteration 6: T_6 = (N_6 = {A,B,C,D,F,G,H} ;
A_6 = { (A,B) , (B,C) , (C,D) , (C,F) , (F,G) , (G,H) }).



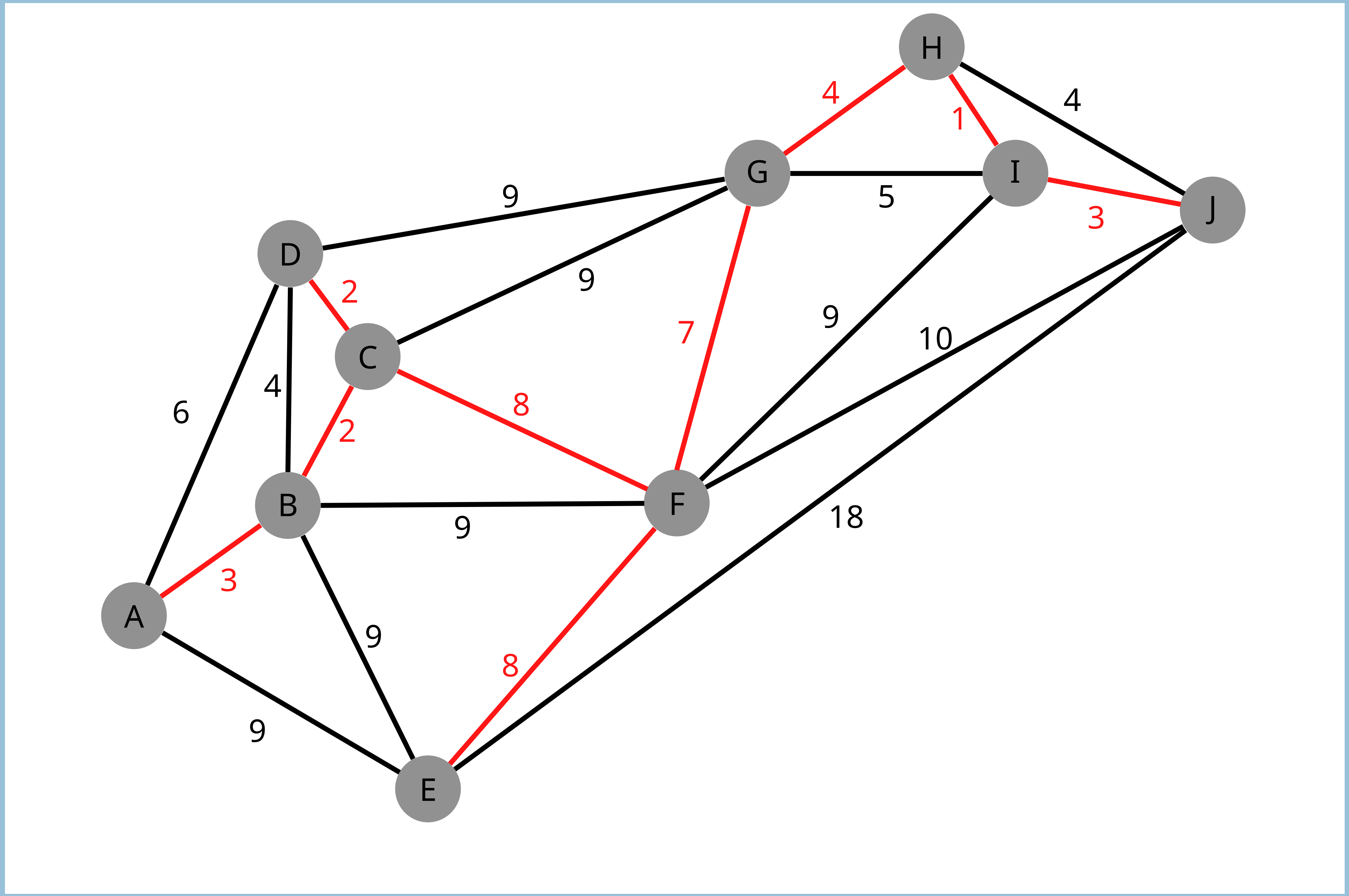
Iteration 7: $T_7 = (N_7 = \{A,B,C,D,F,G,H,I\}$;

$A_7 = \{ (A,B) , (B,C) , (C,D) , (C,F) , (F,G) , (G,H) , (H,I) \})$.



Iteration 8: $T_8 = \left(N_8 = \{A,B,C,D,F,G,H,I,J\} \right.$;

$\left. A_8 = \{ (A,B) , (B,C) , (C,D) , (C,F) , (F,G) , (G,H) , (H,I) , (I,J) \} \right)$.



Iteration 9: T_9 = (N_9 = {A,B,C,D,F,G,H,I,J,E} ;

A_9 = { (A,B) , (B,C) , (C,D) , (C,F) , (F,G) , (G,H) , (H,I) , (I,J), (F,E) }).

So $N \setminus N_9 = \varnothing$

=>The algorithm stops with the optimal total arc length of 38.

Group 1

Thank you for listening!