

Attività di Alternanza Scuola-Lavoro
Roma, 20 Febbraio 2020

Il problema del minimo albero ricoprente

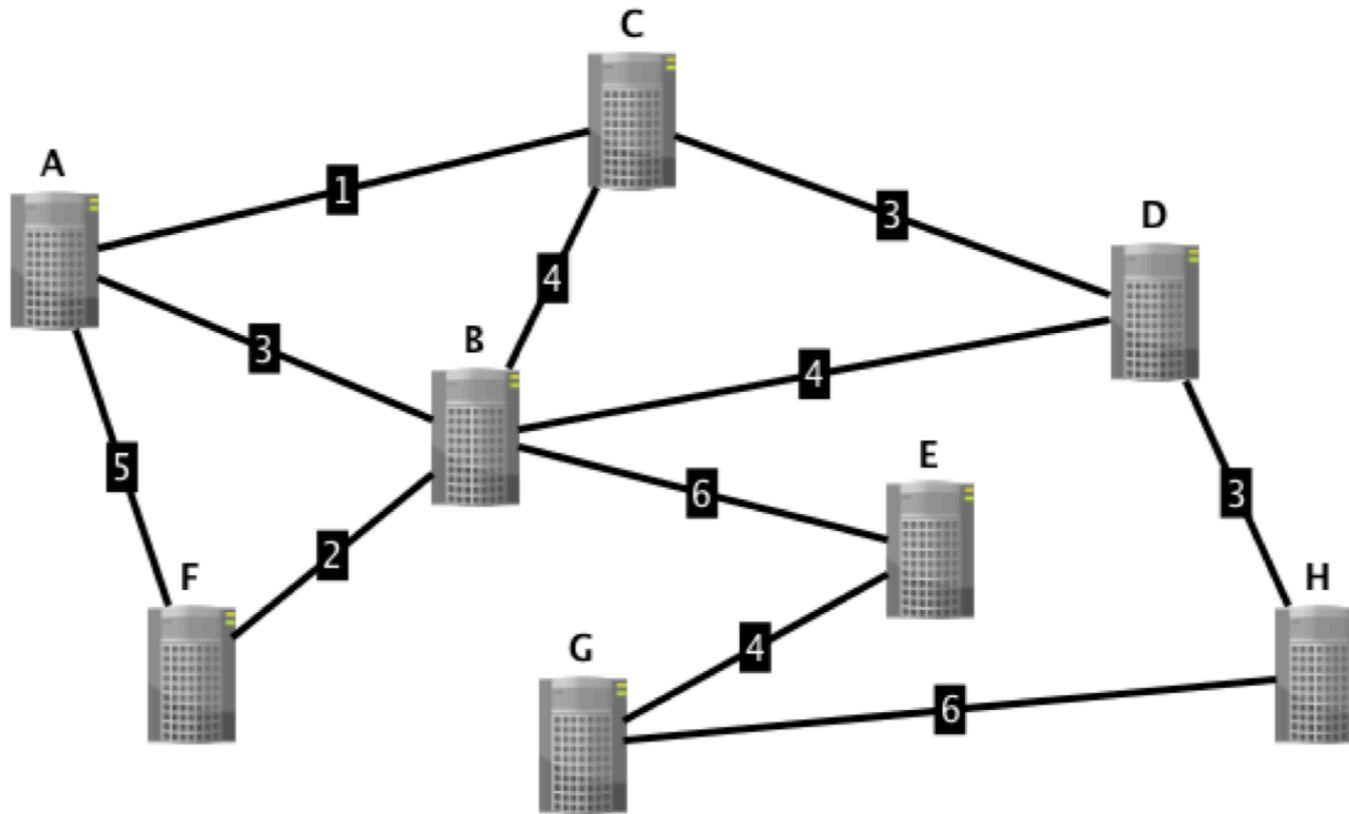


Angelo Monti

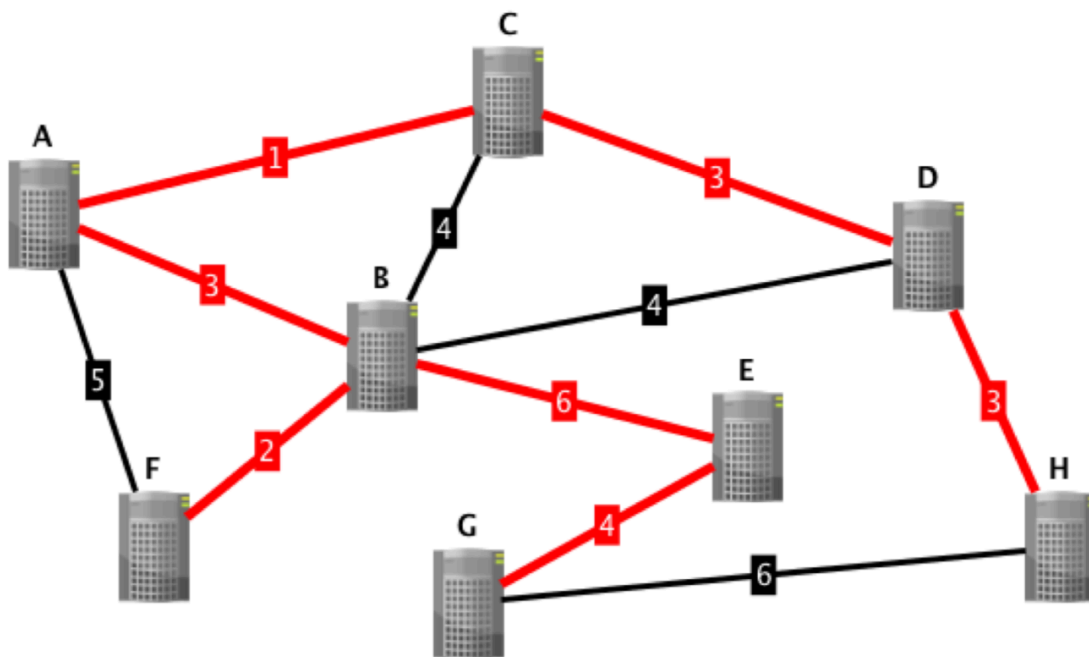
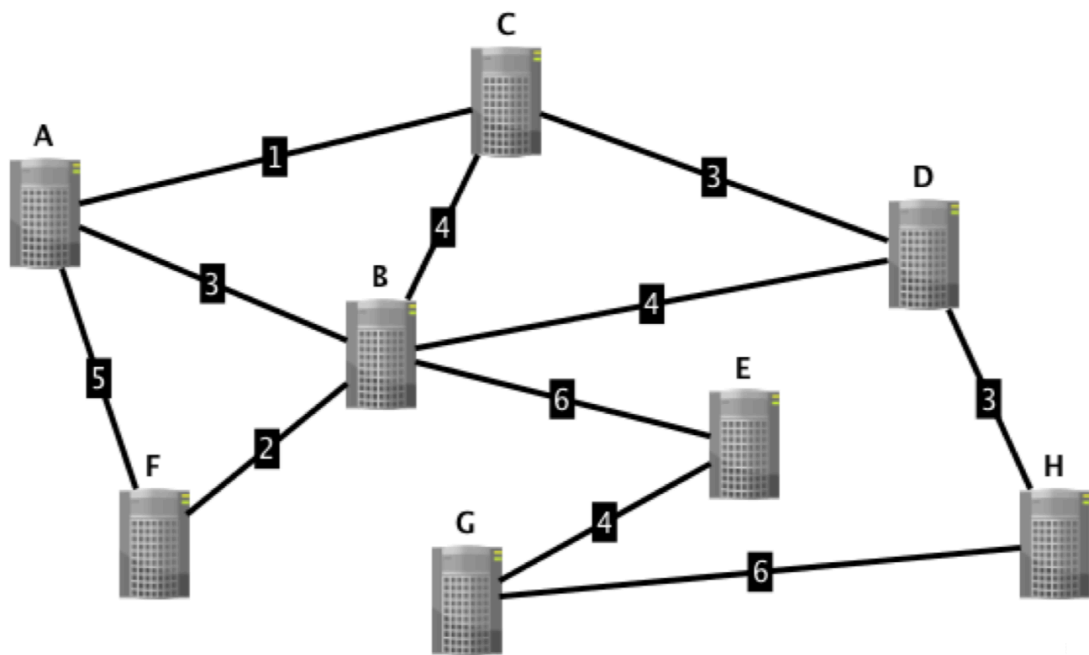
Professore Associato del Dipartimento di Informatica
Sapienza Università di Roma

Il problema del minimo albero di copertura

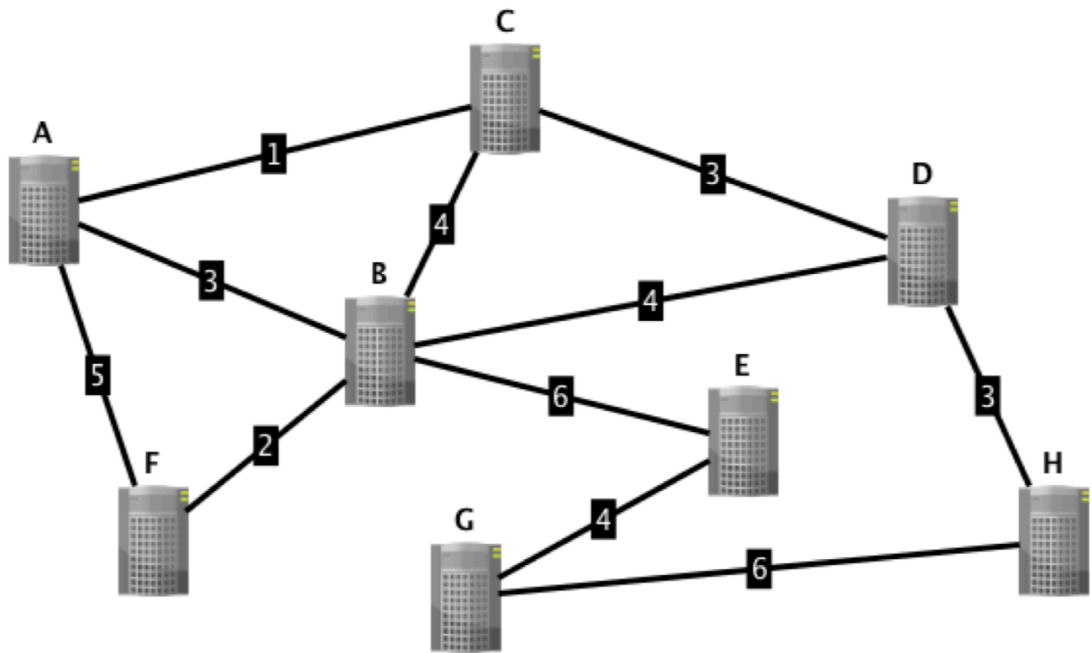
Consideriamo un insieme di computer (server e/o router) che devono essere connessi tramite cavi a formare una rete in modo che ogni computer possa comunicare con ogni altro o tramite un cavo che li collega direttamente o passando per altri computer delle rete. Ogni possibile collegamento tramite cavo ha un costo. Un esempio è mostrato nella figura seguente:



Quindi vogliamo installare alcuni dei collegamenti possibili in modo tale da garantire la connessione della rete e al contempo minimizzare il costo totale.

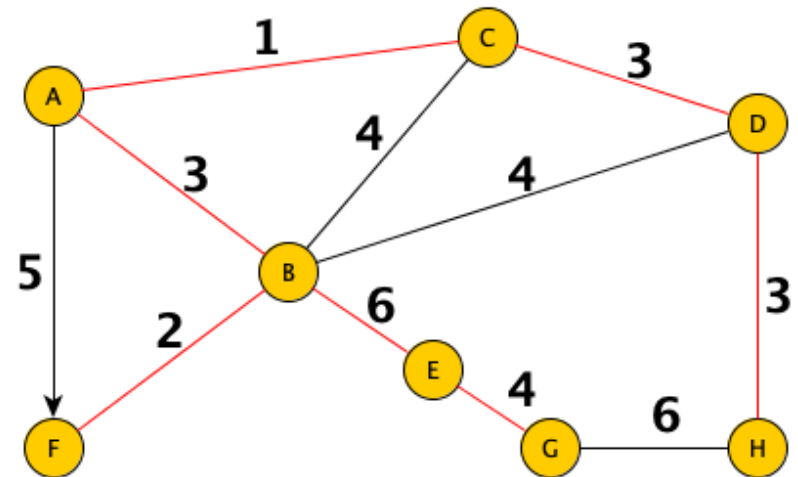
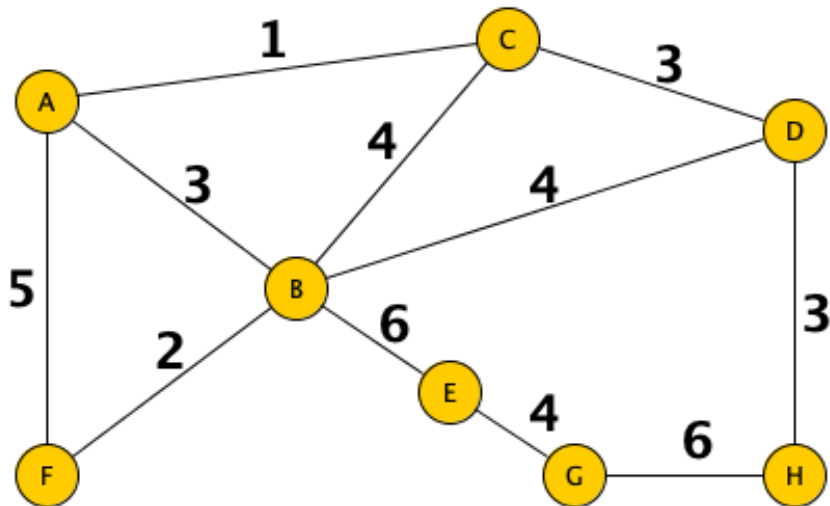


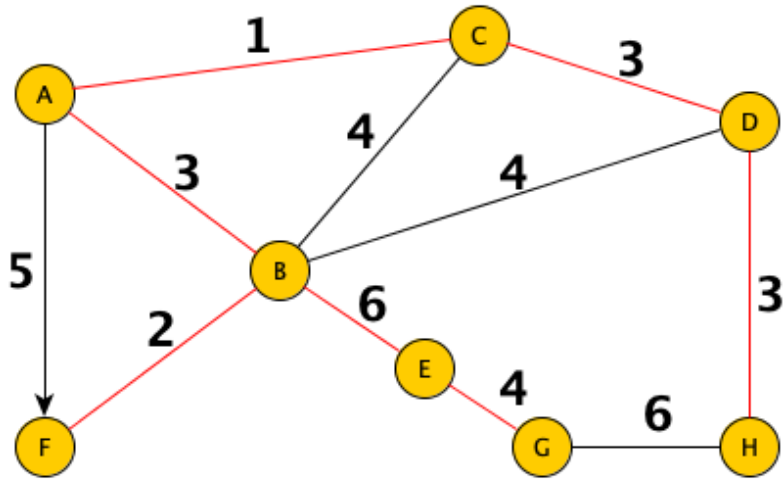
Una possibile soluzione (di costo 22):



Il problema può essere rappresentato tramite un grafo G i cui nodi sono i computer, gli archi sono i possibili collegamenti con i loro costi.

Il requisito della connessione è tradotto nel requisito che l'insieme degli archi da selezionare T deve connettere tutti i nodi.





Nota che nel grafo soluzione (con gli archi in rosso in figura) non sono mai presenti circuiti (l'eliminazione di un qualunque arco del circuito non farebbe perdere la connessione e diminuirebbe il costo).

Un sottoinsieme degli archi del grafo in grado di mantenere i nodi connessi e privo di circuiti è detto **albero di copertura**.

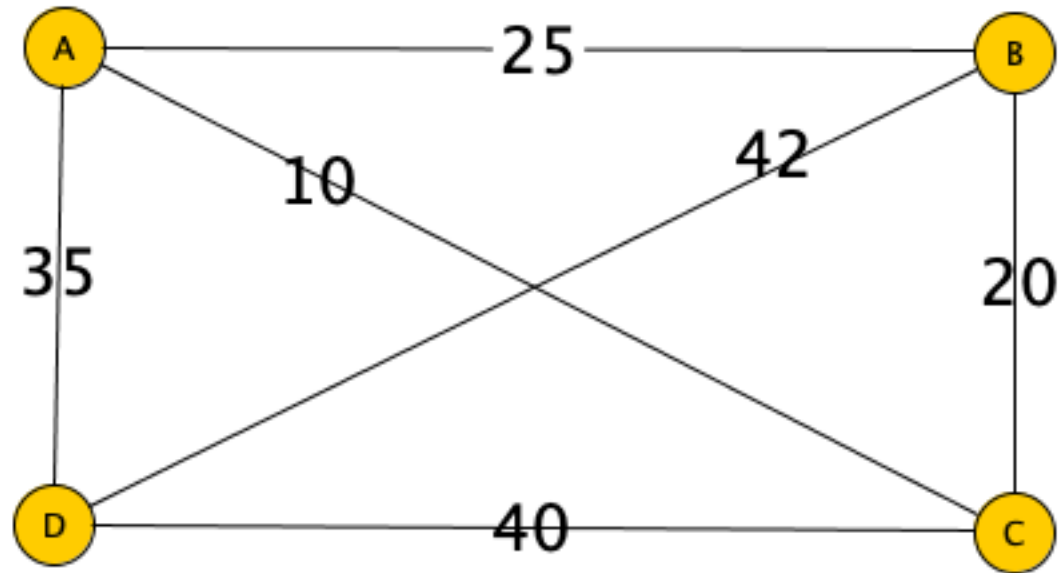
Il problema consiste dunque nel cercare in G un albero di copertura di costo minimo

Dato un grafo G connesso e con archi pesati cerchiamo dunque un suo minimo albero di copertura

Esistono diversi algoritmi per risolvere questo problema. Nel seguito ne vedremo tre:

- 1) Algoritmo di Kruskal
- 2) Versione “rovesciata” dell’algoritmo di Kruskal
- 3) Algoritmo di Prim

Algoritmo di Kruskal



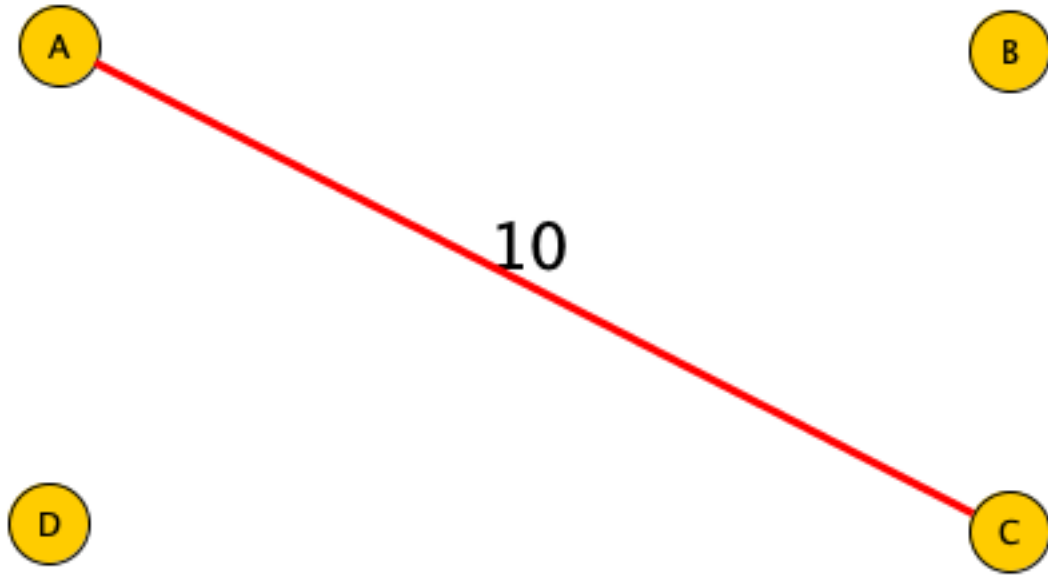
Parti con il grafo che contiene tutti i nodi e nessun arco.

Considera gli archi del grafo in sequenza dopo averli ordinati per costo crescente.

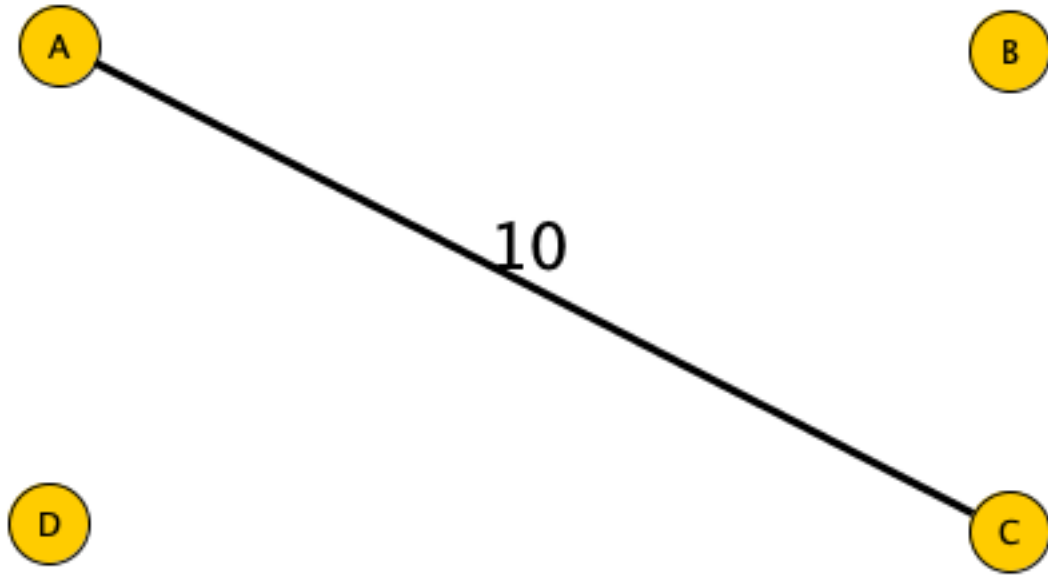
Se l'arco forma ciclo con archi presi allora non prenderlo altrimenti inseriscilo nella soluzione.



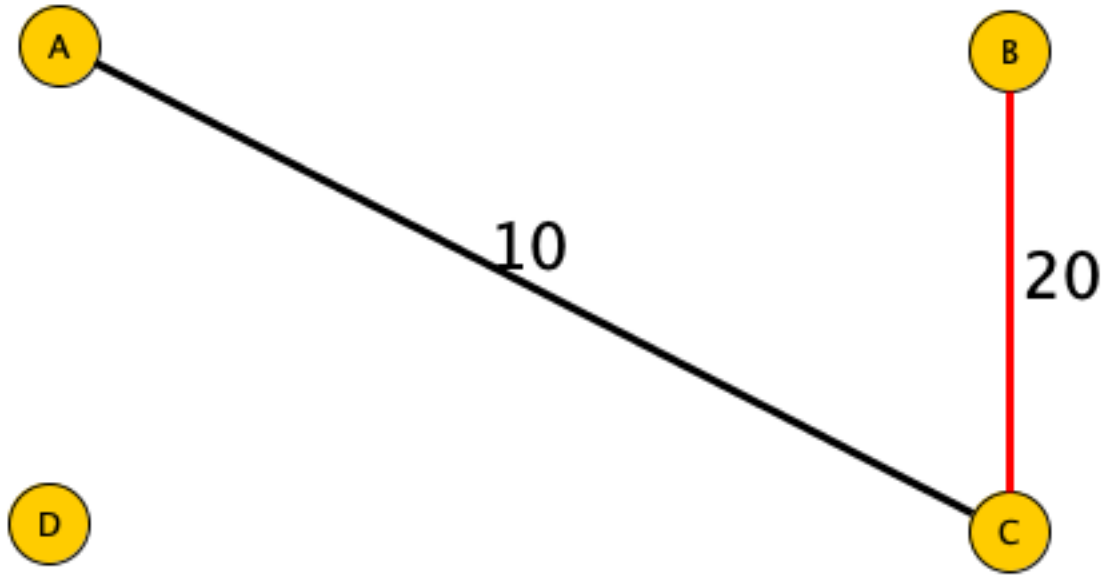
- AC 10
- BC 20
- AB 25
- AD 35
- DC 40
- BD 42



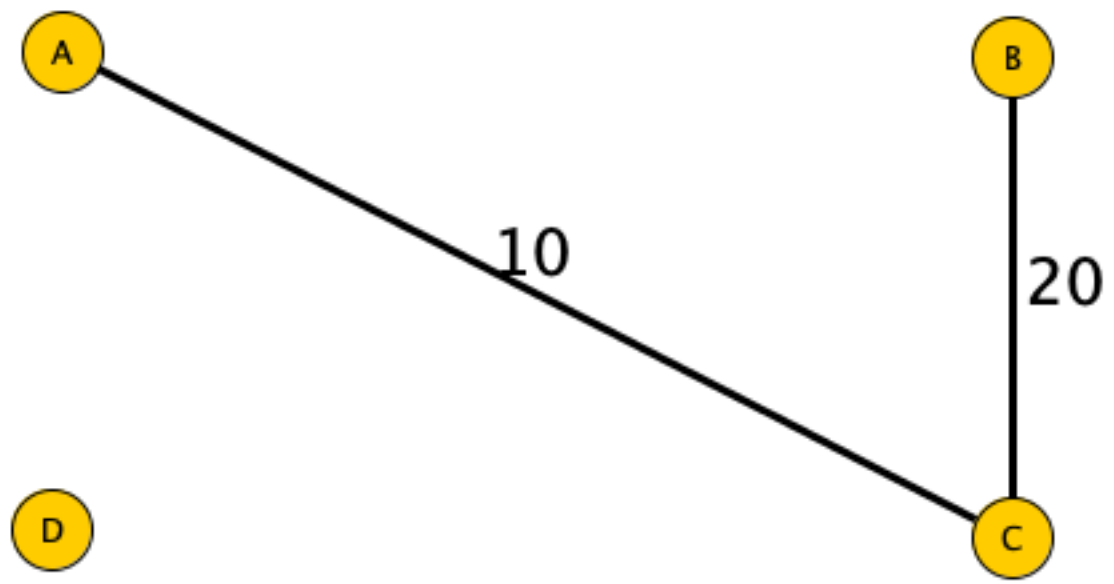
- AC 10
- BC 20
- AB 25
- AD 35
- DC 40
- BD 42



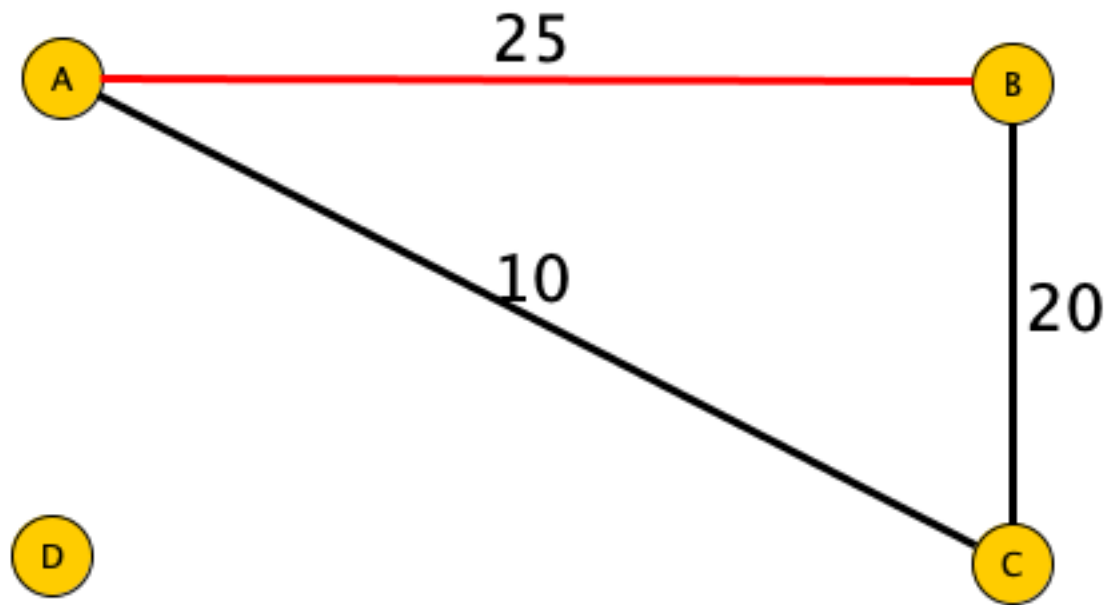
- *AC* 10
- *BC* 20
- *AB* 25
- *AD* 35
- *DC* 40
- *BD* 42



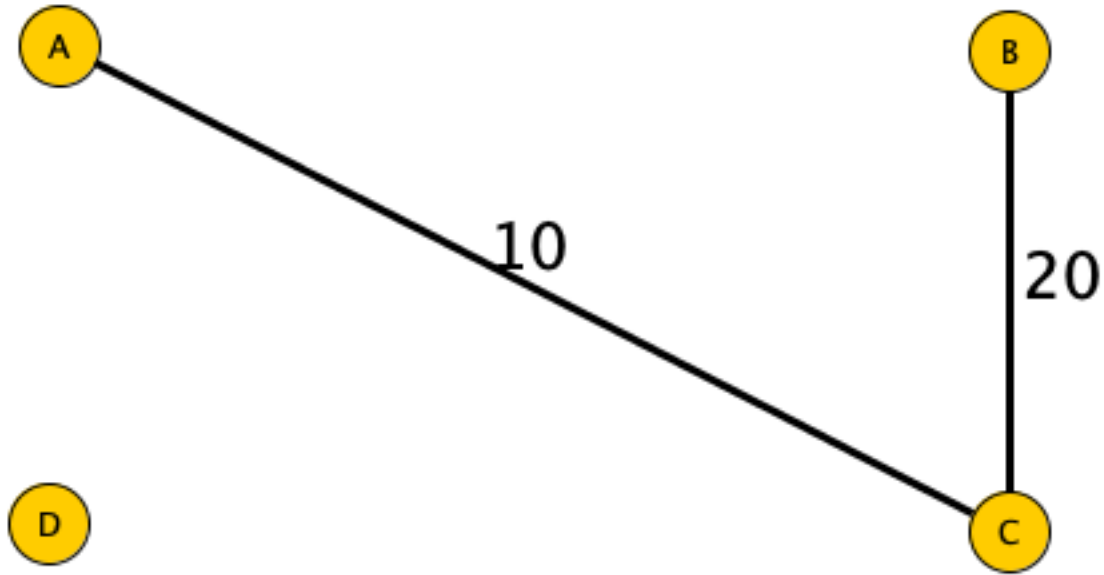
- *AC* 10
- *BC* 20
- *AB* 25
- *AD* 35
- *DC* 40
- *BD* 42



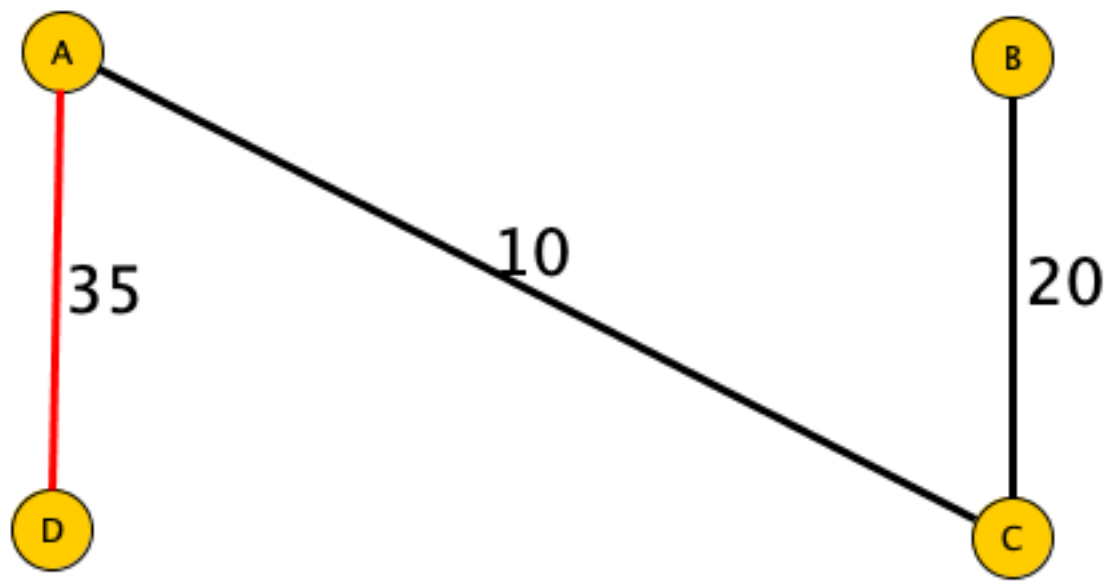
- *AC* 10
- *BC* 20
- *AB* 25
- *AD* 35
- *DC* 40
- *BD* 42



- *AC* 10
- *BC* 20
- *AB* 25
- *AD* 35
- *DC* 40
- *BD* 42



- *AC* 10
- *BC* 20
- *AB* 25
- *AD* 35
- *DC* 40
- *BD* 42



- *AC* 10

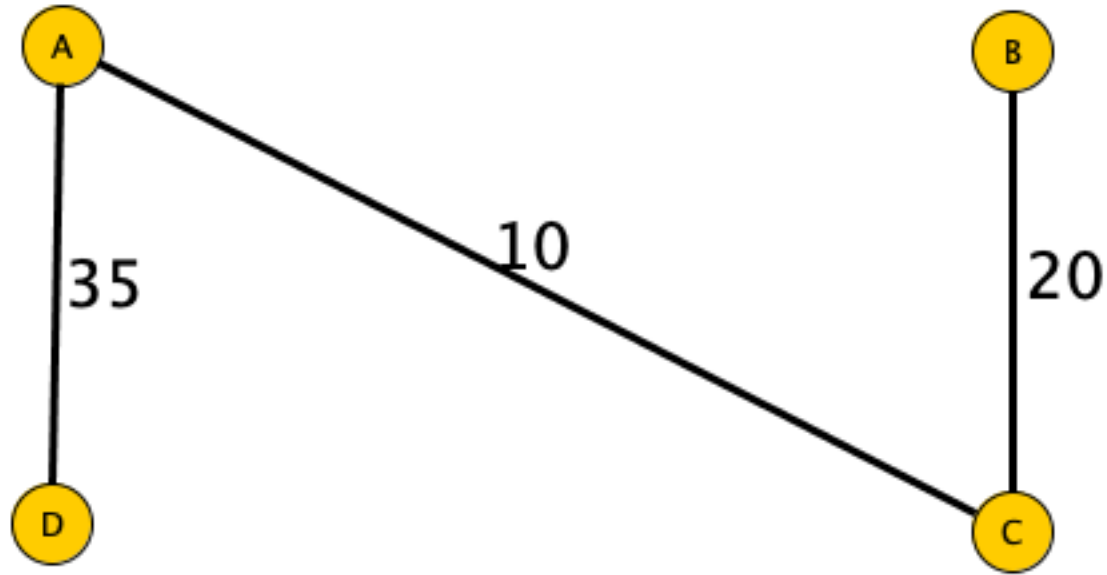
- *BC* 20

- *AB* 25

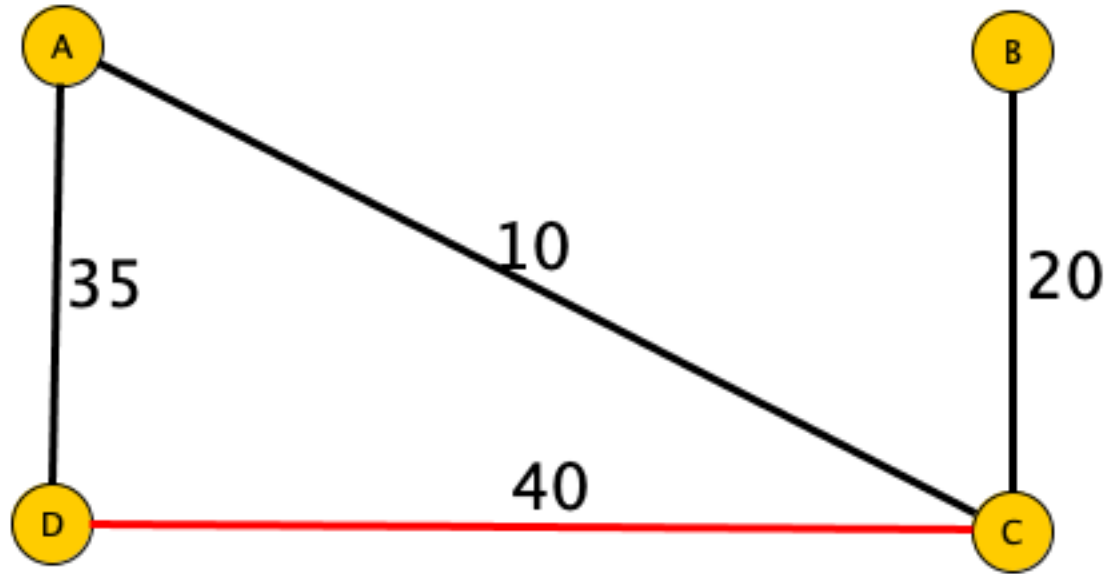
- *AD* 35

- *DC* 40

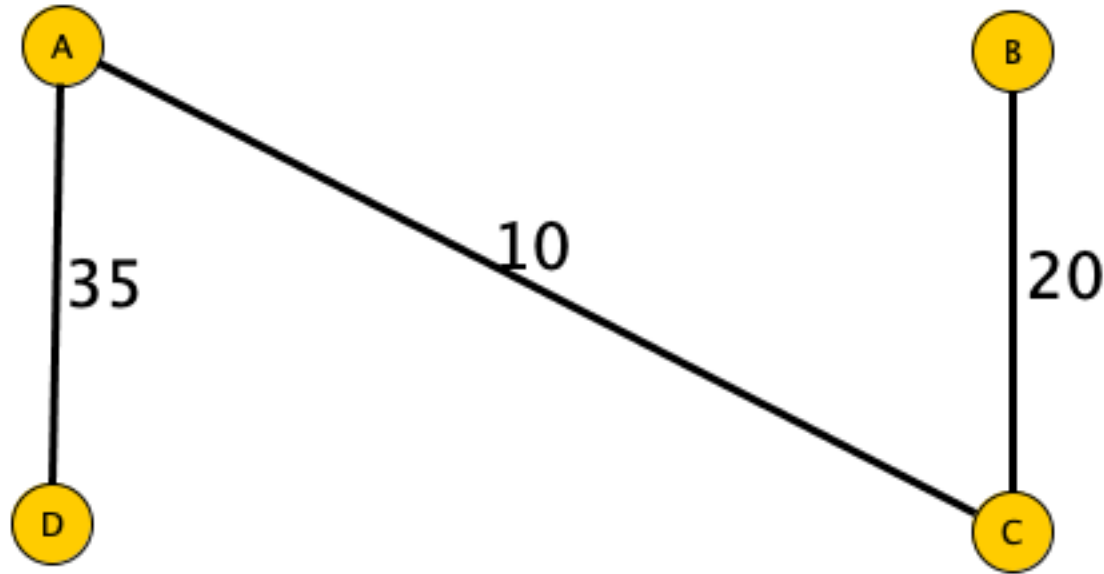
- *BD* 42



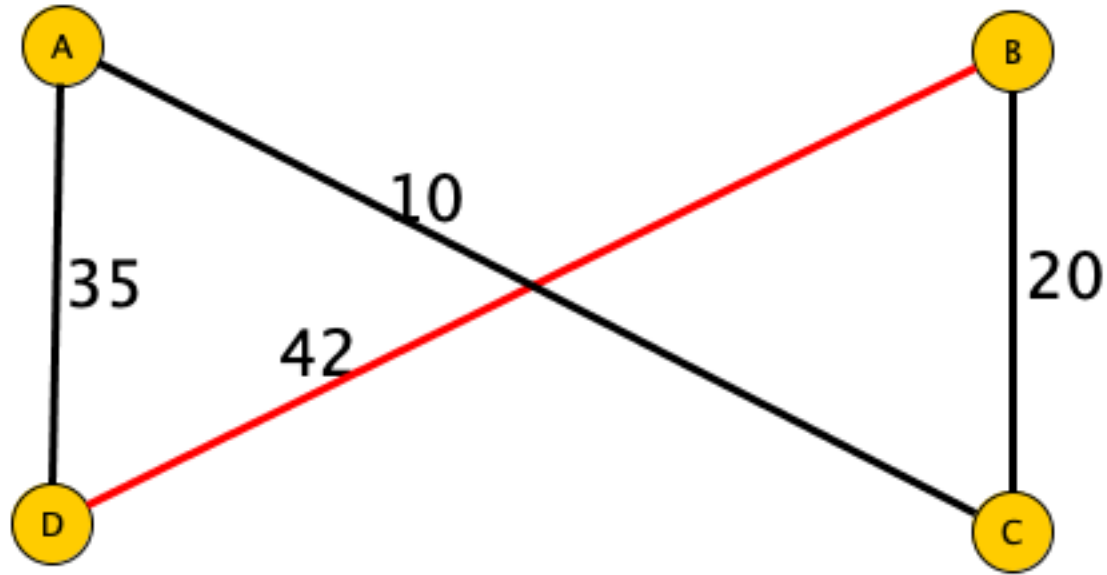
- AC 10
- BC 20
- AB 25
- AD 35
- DC 40
- BD 42



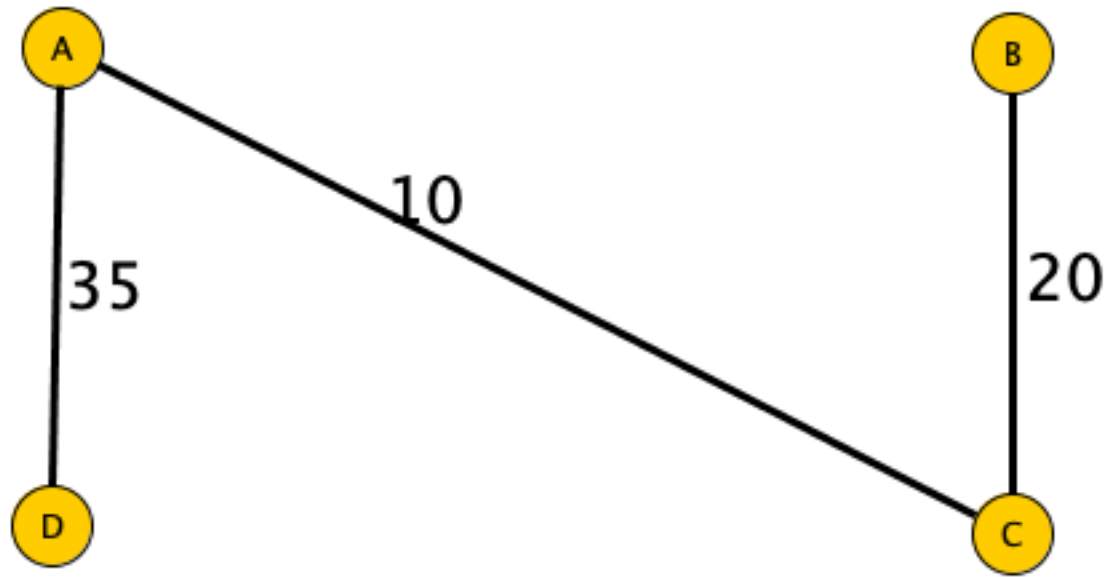
- AC 10
- BC 20
- AB 25
- AD 35
- DC 40
- BD 42



- AC 10
- BC 20
- AB 25
- AD 35
- DC 40
- BD 42

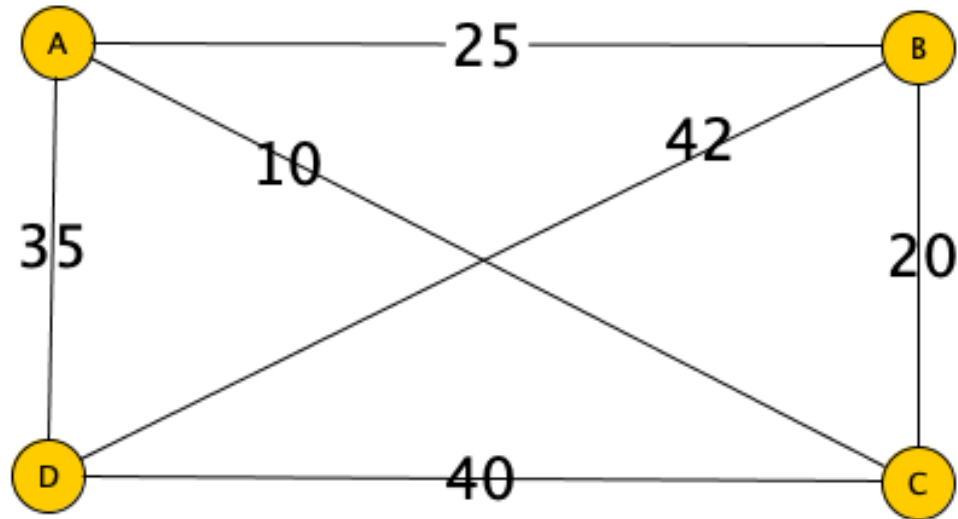


- *AC* 10
- *BC* 20
- *AB* 25
- *AD* 35
- *DC* 40
- *BD* 42



- AC 10
- BC 20
- AB 25
- AD 35
- DC 40
- BD 42

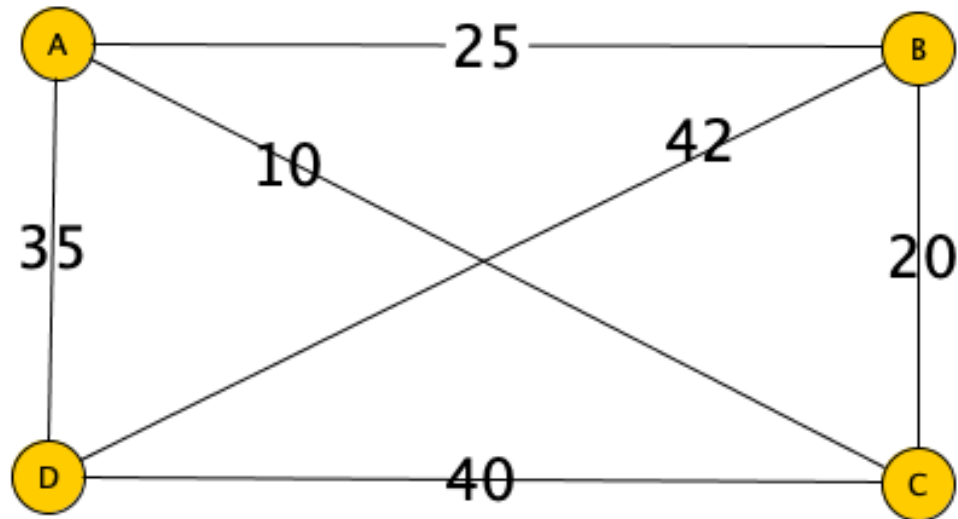
Versione rovesciata dell' algoritmo di Kruskal



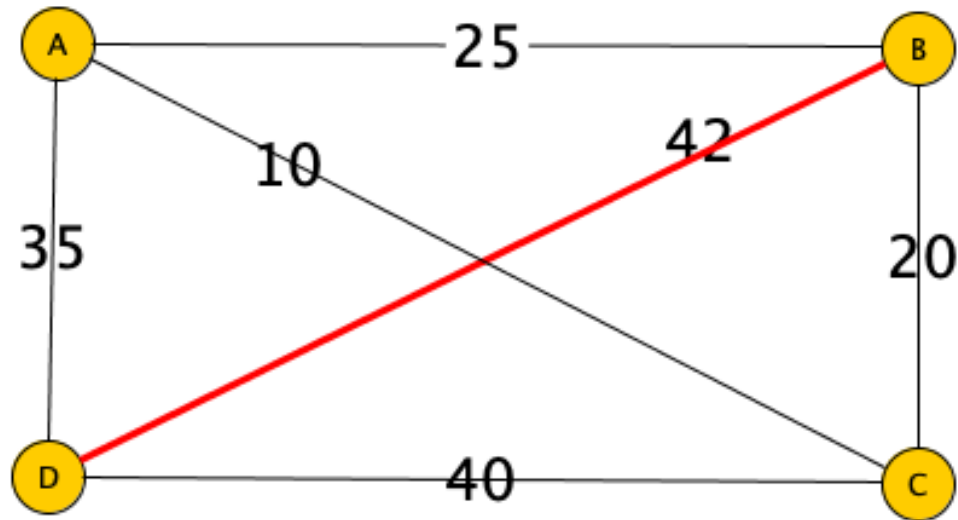
Parti con il grafo G che contiene tutti gli archi del grafo.

Considera gli archi del grafo in sequenza dopo averli ordinati per costo decrescente.

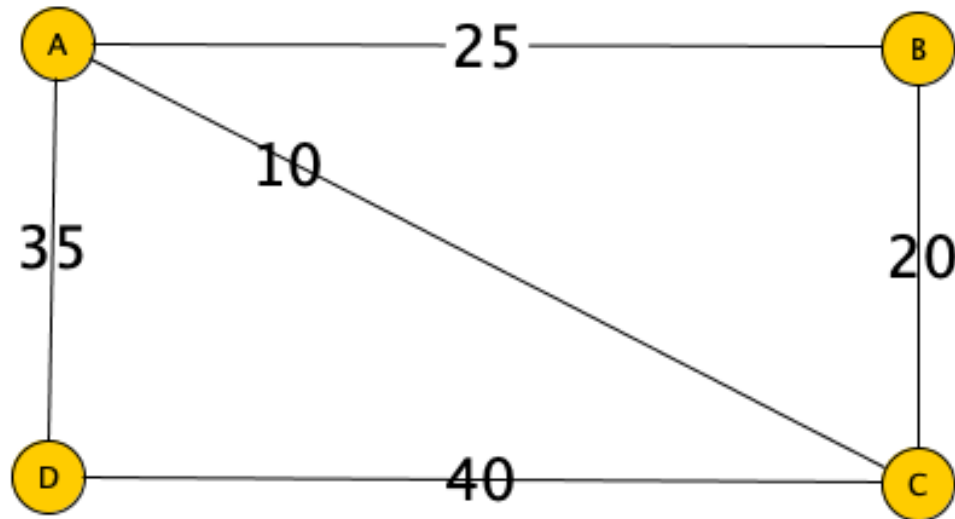
Se l'eliminazione dell'arco sconnette i nodi del grafo allora non eliminarlo altrimenti eliminalo



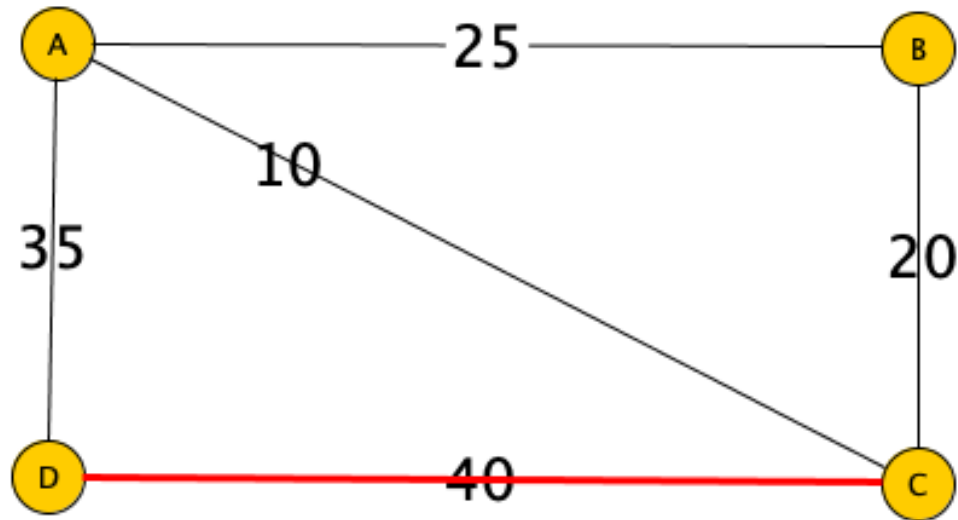
- BD 42
- DC 40
- AD 35
- AB 25
- BC 20
- AC 10



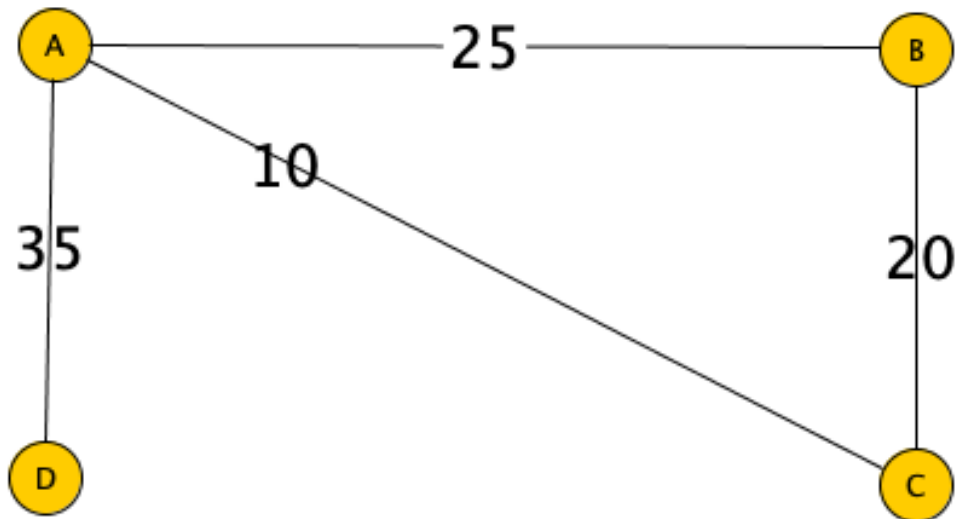
- *BD* 42
- *DC* 40
- *AD* 35
- *AB* 25
- *BC* 20
- *AC* 10



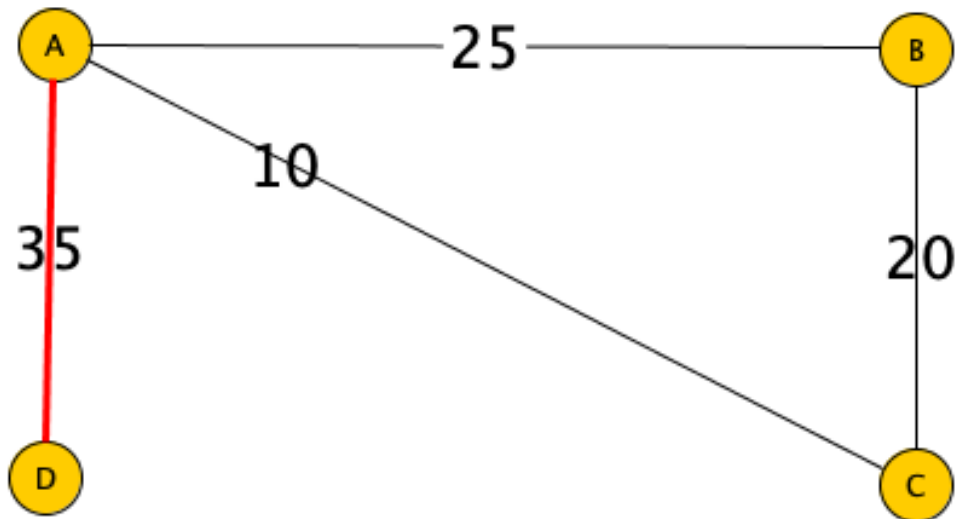
- *BD* 42
- *DC* 40
- *AD* 35
- *AB* 25
- *BC* 20
- *AC* 10



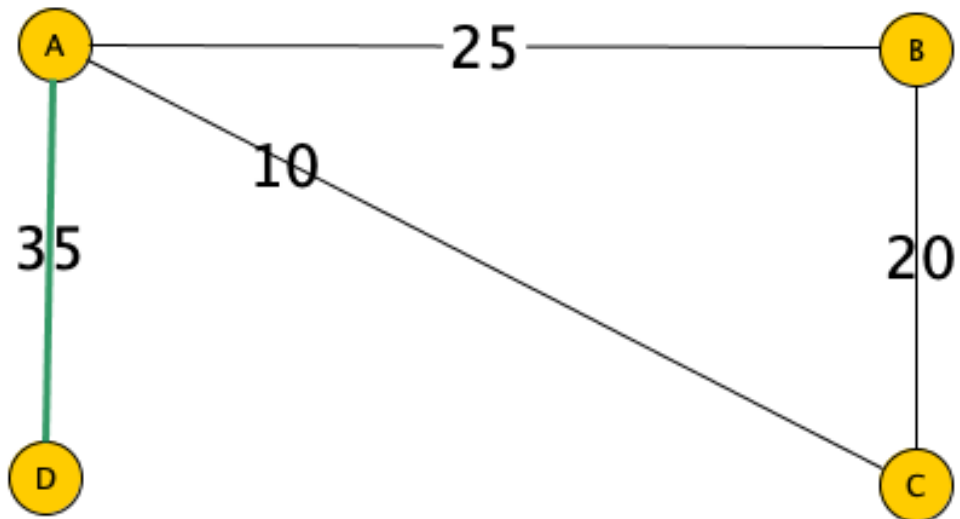
- *BD* 42
- *DC* 40
- *AD* 35
- *AB* 25
- *BC* 20
- *AC* 10



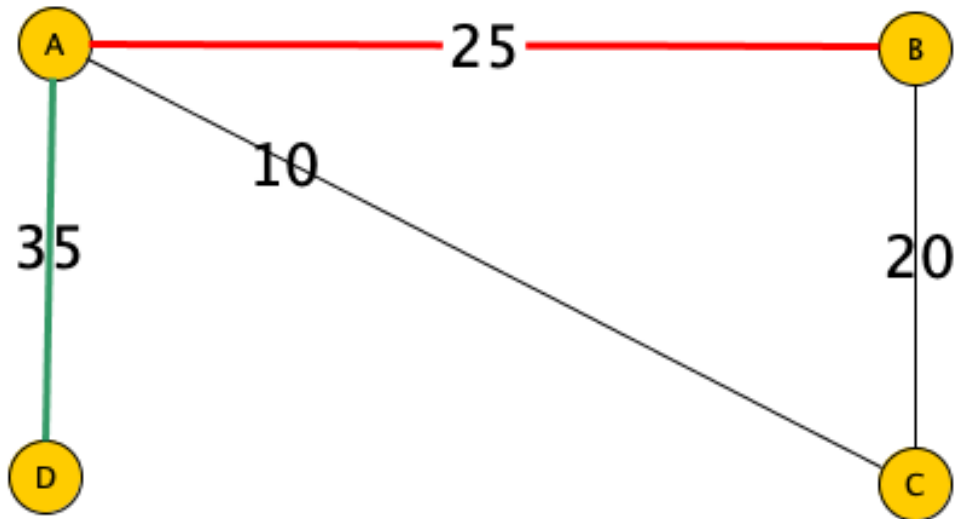
- *BD* 42
- *DC* 40
- *AD* 35
- *AB* 25
- *BC* 20
- *AC* 10



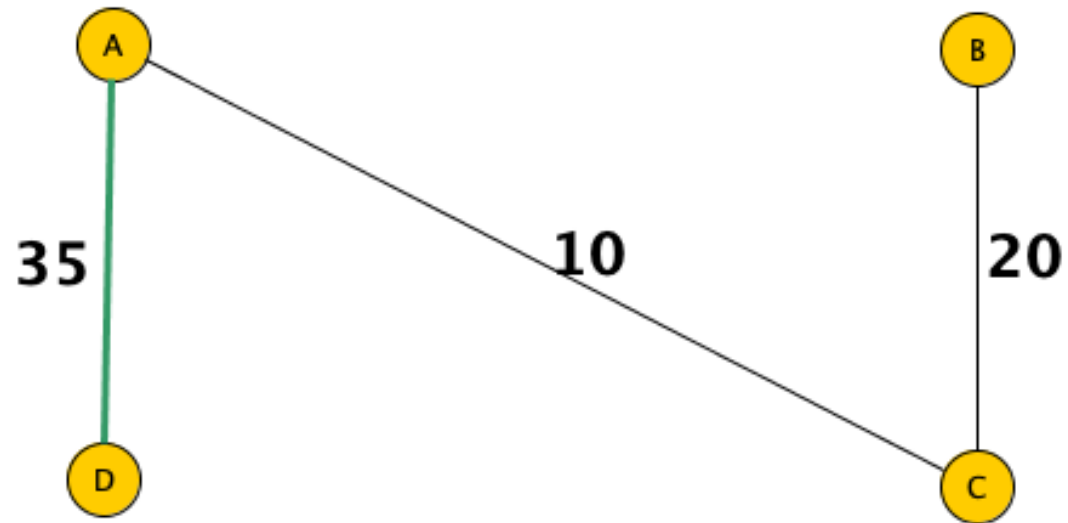
- *BD* 42
- *DC* 40
- *AD* 35
- *AB* 25
- *BC* 20
- *AC* 10



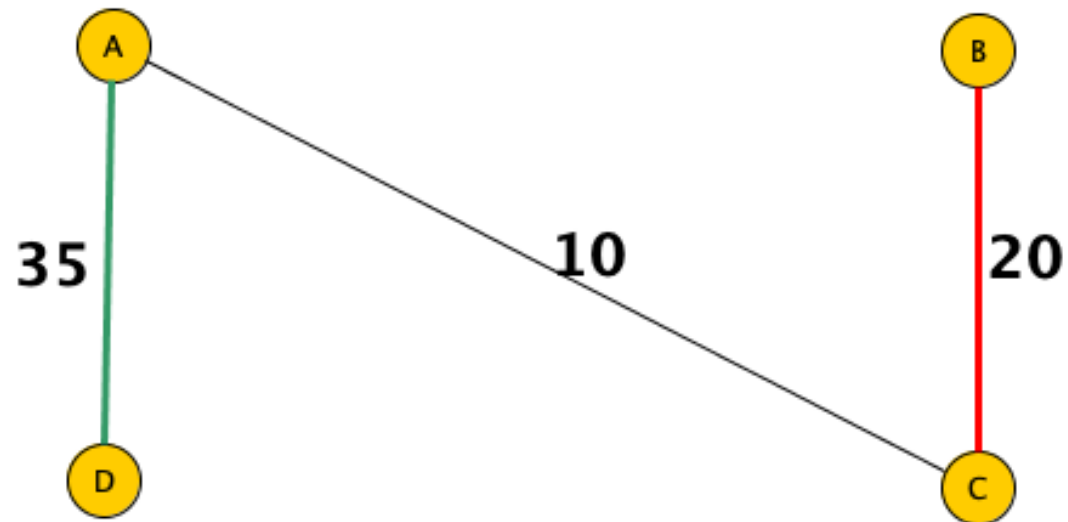
- *BD* 42
- *DC* 40
- *AD* 35
- *AB* 25
- *BC* 20
- *AC* 10



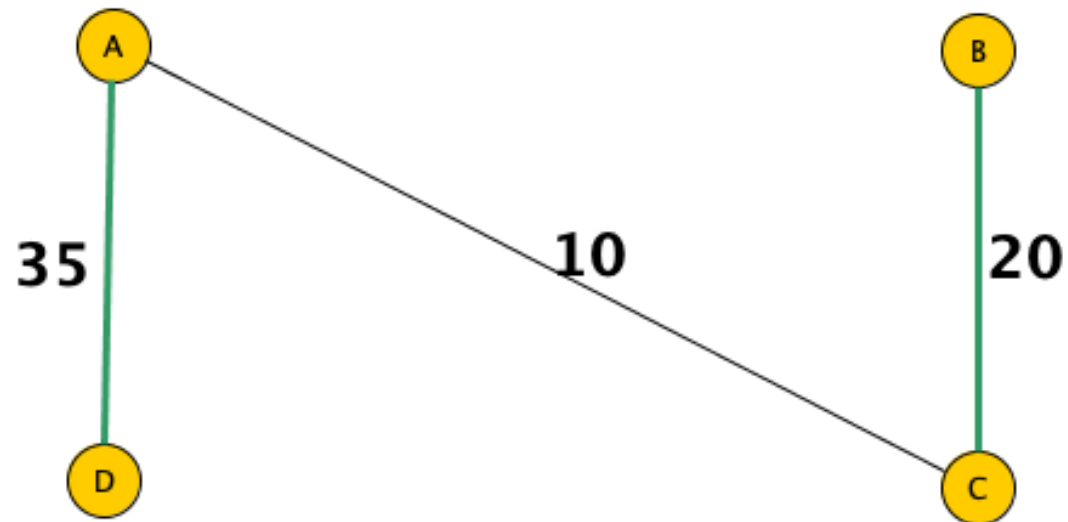
- *BD* 42
- *DC* 40
- *AD* 35
- *AB* 25
- *BC* 20
- *AC* 10



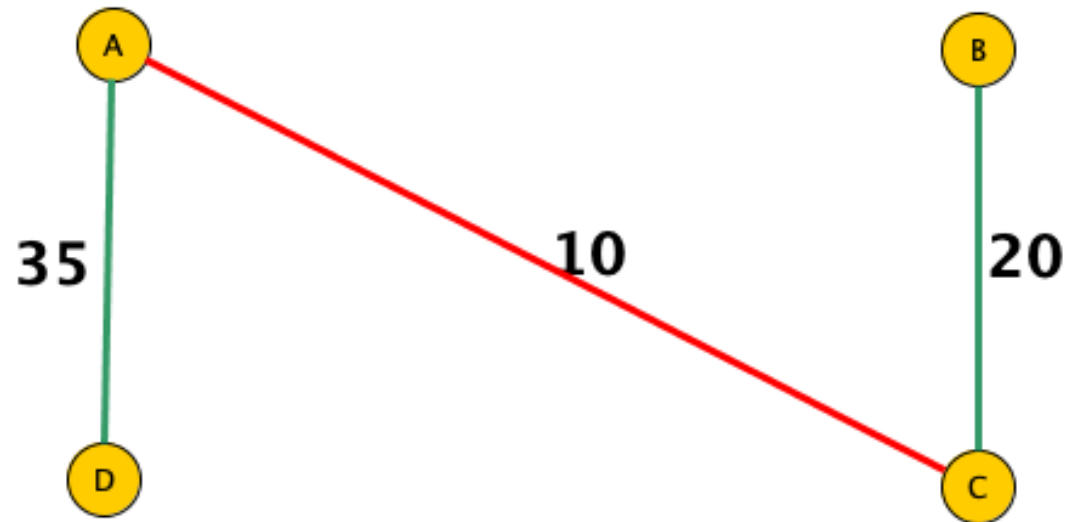
- BD 42
- DC 40
- AD 35
- AB 25
- BC 20
- AC 10



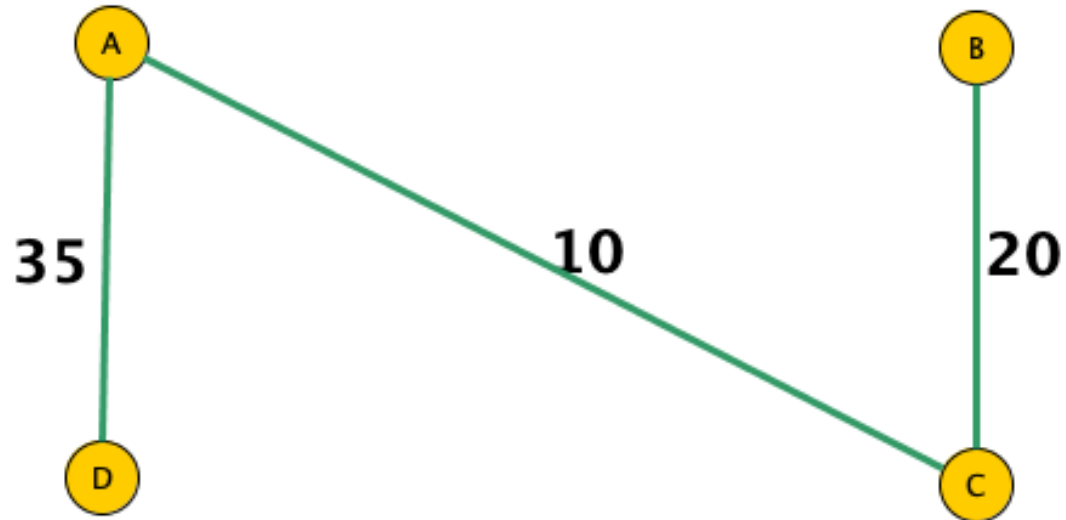
- BD 42
- DC 40
- AD 35
- AB 25
- BC 20
- AC 10



- BD 42
- DC 40
- AD 35
- AB 25
- BC 20
- AC 10

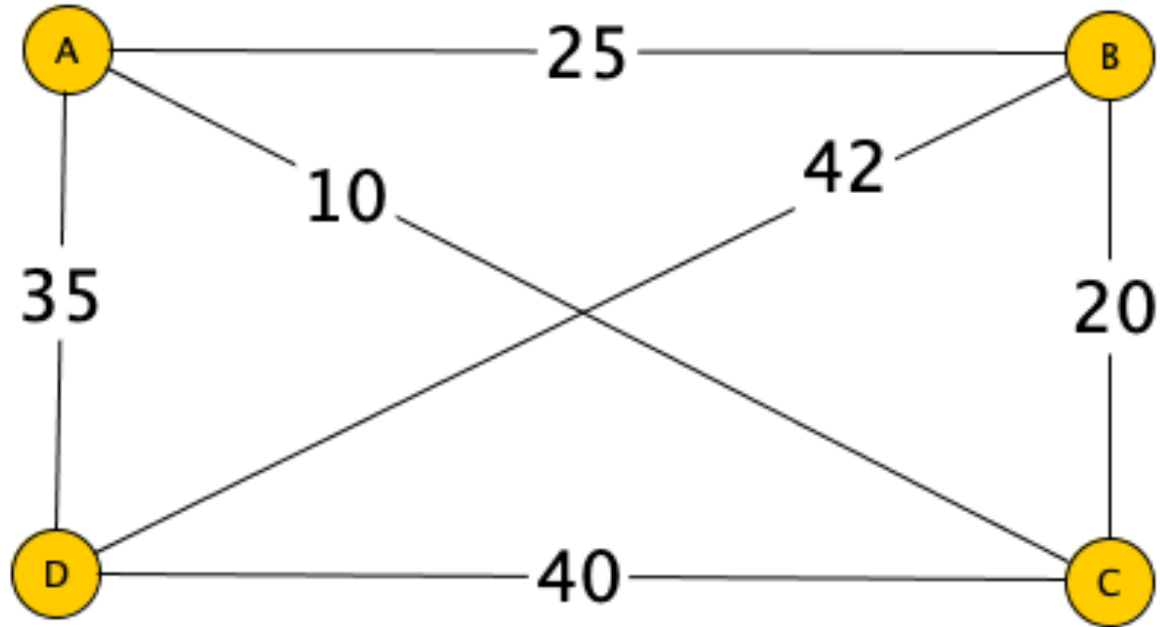


- BD 42
- DC 40
- AD 35
- AB 25
- BC 20
- AC 10



- BD 42
- DC 40
- AD 35
- AB 25
- BC 20
- AC 10

Algoritmo di Prim



All'inizio c'è un unico nodo collegato (il nodo A).
Finché ci sono archi che portano da nodi collegati
a nodi non collegati scegli l'arco di costo minimo
e usalo per collegare il suo estremo non collegato.

