

Relazione MinCut

- **Il codice:**

Per la creazione dei grafi ho utilizzato le funzioni viste nel corso di ASD l'anno scorso.

Quello che fa il codice è: creare il grafo, prendere randomicamente un nodo e fonderlo con il primo della sua lista di adiacenza, fondere le due liste di adiacenza, cancellare i due vecchi nodi e ripetere tali operazioni fino a quando non rimangono solo due nodi nel grafo, a quel punto conta il numero di archi che li collegano i quali saranno un taglio minimo candidato.

Le funzioni in particolare sono:

Il *main()* si occupa di lanciare il processo di MinCut 10^5 volte e di stampare i risultati che comprendono il numero di volte e la probabilità di ogni taglio minimo candidato e quello che è uscito più frequentemente.

L'operazione di ricerca del taglio minimo parte dalla funzione *start()* che si occupa di creare il grafo e chiama la funzione *randomizer()* che va a prendere randomicamente un nodo del grafo da fondere.

Dopo di che viene chiamata la funzione *MCMinCut()* che fa tutto il "lavoro sporco" ovvero, attraverso le funzioni *renameVertex()*, *mergeAdjList()* e *removeVertex()*, crea il nuovo nodo, il quale sarà il nodo pescato randomicamente fuso con il primo nodo della sua lista di adiacenza, va a cambiare tutti i riferimenti nelle liste di adiacenza ai due vecchi nodi che ora sono fusi, fonde le liste di adiacenza dei suddetti nodi e infine li cancella. Queste operazioni vengono eseguite fino a quando non rimangono solo due nodi nel grafo.

A questo punto viene chiamata la funzione *minCut()* che va a contare gli archi che collegano gli ultimi due nodi rimasti.

- **Considerazioni finali:**

Essendo un algoritmo randomizzato di tipo Monte Carlo non è sicuro che ad ogni run ci restituisca effettivamente il taglio minimo ma è sicuro che eseguendo un numero elevato di run il valore che si presenterà più spesso sarà il taglio minimo.

```
Il taglio minimo candidato di valore 1 è uscito 0 volte, con probabilità 0
Il taglio minimo candidato di valore 2 è uscito 0 volte, con probabilità 0
Il taglio minimo candidato di valore 3 è uscito 0 volte, con probabilità 0
Il taglio minimo candidato di valore 4 è uscito 37557 volte, con probabilità 0.37557
Il taglio minimo candidato di valore 5 è uscito 24786 volte, con probabilità 0.24786
Il taglio minimo candidato di valore 6 è uscito 0 volte, con probabilità 0
Il taglio minimo candidato di valore 7 è uscito 2298 volte, con probabilità 0.02298
Il taglio minimo candidato di valore 8 è uscito 24406 volte, con probabilità 0.24406
Il taglio minimo candidato di valore 9 è uscito 8362 volte, con probabilità 0.08362
Il taglio minimo candidato di valore 10 è uscito 2026 volte, con probabilità 0.02026
Il taglio minimo candidato di valore 11 è uscito 565 volte, con probabilità 0.00565

Il taglio minimo è 4 che è uscito 37557 volte con una probabilità di 0.37557
```

Chiaramente, come possiamo vedere, essendo il taglio minimo di 4 è naturale che non esca mai un taglio minimo candidato di valore inferiore.

Ho notato che non esce mai un taglio minimo candidato di valore 6, questo dovrebbe essere dovuto dalla conformazione del grafo.

Sappiamo che MCMinCut restituisce un taglio minimo con probabilità almeno pari a $2/(n^2)$ ovvero circa 0.02 , quindi pur essendo di un ordine di grandezza maggiore il risultato ottenuto è in linea con la stima.