

Esercitazione 3: Applications of Graph Search Algorithms

Giacomo Paesani

March 27, 2025

Esercizio 1. Fornire un algoritmo in pseudo-codice che dato un grafo diretto e aciclico $G = (V, E)$, restituisce un ordinamento topologico di G . E' possibile implementarlo in modo che il tempo di esecuzione sia $\Theta(|V| + |E|)$? Come dovrebbe essere modificato tale algoritmo per restituire l'elenco di tutti gli ordinamenti topologici di G ? E' possibile fare questa ulteriore modifica mantenendo lo stesso tempo di esecuzione?

Esercizio 2 (I. Salvo). Descrivere in pseudo-codice un algoritmo che, dato un grafo non diretto G , descrivere un algoritmo che ne orienta gli archi in modo da creare un grafo G' diretto e aciclico. Questo algoritmo deve avere tempo di esecuzione $\Theta(n + m)$.

Esercizio 3 (22.2-9, [1]). Fornire un algoritmo in pseudo-codice che, dato un grafo non diretto e connesso $G = (V, E)$, trova una passeggiata in G che attraversa tutti gli archi una e una sola volta in ognuna delle due direzioni in tempo $\mathcal{O}(|V| + |E|)$.

Esercizio 4 (22.4-2, [1]). Fornire un algoritmo in pseudo-codice che dato un grafo diretto e aciclico $G = (V, E)$ e due vertici s e t , restituisce il numero di tutti i cammini da s a t in G .

Esercizio 5. Sia un grafo diretto $G = (V, E)$ ed s e t due vertici di G . G si dice *s-t-connesso* se ogni vertice di G è in almeno un cammino da s a t . Fornire un algoritmo in pseudo-codice che dato un grafo diretto e aciclico $G = (V, E)$ ed s e t due vertici di G , restituisce un sottografo G' di G massimale *s-t-connesso*.

References

- [1] Thomas H Cormen, Charles E Leiserson, Ronald L Rivest, and Clifford Stein. Introduction to algorithms. 2022.