

Cognome e Nome:  
Numero di Matricola:

**Spazio riservato alla correzione**

1	2	3	4	Totale
/25	/25	/25	/25	/100

## 1. Grafi

- a) Scrivere lo pseudocodice dell'algoritmo per ottenere l'ordinamento topologico di un DAG e si analizzi il tempo di esecuzione dell'algoritmo proposto. Analizzare il tempo di esecuzione significa fornire un limite superiore asintotico quanto migliore e` possibile al tempo di esecuzione dell'algoritmo giustificando la risposta.

- b) Dimostrare per induzione che ogni DAG ha un ordinamento topologico. La dimostrazione deve essere chiara e schematica.

- c) **Bonus** Descrivere come implementare l'algoritmo al punto precedente in modo che abbia tempo  $O(n+m)$ , dove  $n$  ed  $m$  sono rispettivamente il numero di nodi e di archi del grafo e **dimostrare** che il tempo di esecuzione dell'algoritmo così implementato è appunto  $O(n+m)$ .

2. Algoritmi greedy

- a) Si descriva in modo chiaro e schematico il problema della minimizzazione dei ritardi.

- b) Fornire un controesempio che dimostri che la strategia greedy **Shortest Processing Time First** non sempre produce la soluzione ottima.

- c) Si dica quale strategia greedy e' in grado di produrre la soluzione ottima.

- d) Si scriva lo pseudocodice dell'algoritmo per la minimizzazione dei ritardi. Si analizzi il tempo di esecuzione dell'algoritmo e si dica in quale punto dell'algoritmo viene effettuata la scelta greedy. Analizzare il tempo di esecuzione significa fornire un limite superiore asintotico quanto migliore e` possibile al tempo di esecuzione dell'algoritmo giustificando la risposta.

3. Programmazione dinamica.

- a) Fornire una formula per il calcolo del valore della soluzione ottima per il problema dei cammini minimi basata sul principio della programmazione dinamica.  
Spiegare in modo chiaro e schematico come si arriva alla formula da voi fornita. Occorre anche dire quale risultato riguardante i grafi privi di cicli negativi occorre usare e dove questo risultato si usa.

- b) Scrivere lo pseudocodice dell'algoritmo di Bellman-Ford e analizzare il tempo di esecuzione dell'algoritmo. Analizzare il tempo di esecuzione significa fornire un limite superiore asintotico quanto migliore e' possibile al tempo di esecuzione dell'algoritmo giustificando la risposta.

4. **Massimo flusso**

- a) Si definiscano le nozioni di **grafo residuo**, **cammino aumentante**, **collo di bottiglia (bottleneck)**.

- b) Si descriva in dettaglio il comportamento dell'algoritmo di Ford\_Fulkerson e degli algoritmi da esso invocati. Non e' sufficiente scrivere lo pseudocodice.

b) Si analizzi il tempo di esecuzione asintotico dell'algoritmo di Ford-Fulkerson nel caso in cui le capacità siano degli interi. Analizzare il tempo di esecuzione significa fornire un limite superiore asintotico quanto migliore è possibile al tempo di esecuzione dell'algoritmo giustificando la risposta.

c) Si consideri la seguente rete di flusso e la funzione di flusso i cui valori sono indicati a sinistra delle capacità degli archi. Si fornisca la funzione flusso con valore massimo applicando una o più iterazioni dell'algoritmo di Ford-Fulkerson. Per ogni iterazione dell'algoritmo disegnare la rete residua corrispondente. **Occorre indicare per ogni arco qual è il flusso inviato sull'arco dalla funzione flusso di valore massimo.**

