

Cognome e Nome:
Numero di Matricola:

Spazio riservato alla correzione

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

1.Grafi

a) Si descriva in modo schematico e in **italiano** l'algoritmo che computa l'ordinamento topologico di un grafo direzionato aciclico.

b) Si analizzi il tempo di esecuzione dell'algoritmo proposto al punto precedente nel caso in cui l'algoritmo **non** faccia uso di particolari strutture dati atte a migliorarne l'efficienza. 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 modifichi l'algoritmo fornito al punto a) in modo da renderlo più efficiente. Occorre descrivere i passi compiuti dall'algoritmo per inizializzare le strutture dati utilizzate e le operazioni effettuate dall'algoritmo sulle strutture dati in ciascuna chiamata ricorsiva.

d) Si analizzi il tempo di esecuzione dell'algoritmo proposto al punto c). Analizzare il tempo di esecuzione significa fornire un limite superiore asintotico quanto migliore è possibile al tempo di esecuzione dell'algoritmo **giustificando la risposta**.

2. Algoritmi greedy

a) Si descriva in modo chiaro e schematico in che cosa consiste un'istanza del problema della minimizzazione dei ritardi (input) e qual è l'obiettivo del problema (output). Definire in modo preciso le quantità che intervengono nella descrizione dell'output del problema. Se dalla risposta a questo punto si evincerà che lo studente non sa in cosa consiste il problema della minimizzazione dei ritardi, i punti successivi dell'esercizio non saranno valutati.

b) Si fornisca un controesempio che dimostra che la strategia shortest processing time first non sempre fornisce la soluzione ottima per il problema della minimizzazione dei ritardi.

c) Si scriva lo pseudocodice di un algoritmo greedy che trova la soluzione ottima per il problema della minimizzazione dei ritardi descrivendo il significato di tutte le variabili che compaiono nel codice. **Nel caso in cui non venga fornita questa descrizione, l'esercizio sarà valutato 0 punti.**

d) Si analizzi il tempo di esecuzione dell'algoritmo fornito al punto c). Analizzare il tempo di esecuzione significa fornire un limite superiore asintotico quanto migliore è possibile al tempo di esecuzione dell'algoritmo **giustificando la risposta**.

e) Si consideri la seguente istanza del problema della minimizzazione dei ritardi sotto forma di insieme di coppie del tipo $[t_i, d_i]$: $\{[9, 5], [4, 8], [2, 10], [3, 9], [2, 6]\}$
Fornire **la soluzione ottima** prodotta dall'algoritmo al punto c) e **il ritardo massimo** ad essa associato.

f) **[Bonus]** Quali osservazioni vengono utilizzate per dimostrare che la soluzione greedy per il problema della minimizzazione dei ritardi è ottima? Spiegare in modo chiaro e conciso perchè queste osservazioni implicano che la soluzione greedy è ottima.

3. Programmazione dinamica.

a) Si descriva in modo chiaro e schematico in che cosa consiste un'istanza del problema Subset Sum (input) e qual è l'obiettivo del problema (output). Definire in modo preciso le quantità che intervengono nella descrizione dell'output del problema. Se dalla risposta a questo punto si evincerà che lo studente non sa in cosa consiste il suddetto problema, i punti successivi dell'esercizio non saranno valutati.

b) Fornire una relazione di ricorrenza per il calcolo del valore della soluzione ottima per il problema Subset Sum. **Spiegare in modo chiaro e schematico come si arriva alla formula da voi fornita.**

c) Scrivere lo pseudocodice dell'algoritmo iterativo che produce il valore della soluzione ottima per Subset Sum e analizzare il tempo di esecuzione dell'algoritmo. Analizzare il tempo di esecuzione significa fornire un limite superiore asintotico quanto migliore è possibile al tempo di esecuzione dell'algoritmo **giustificando la risposta**.

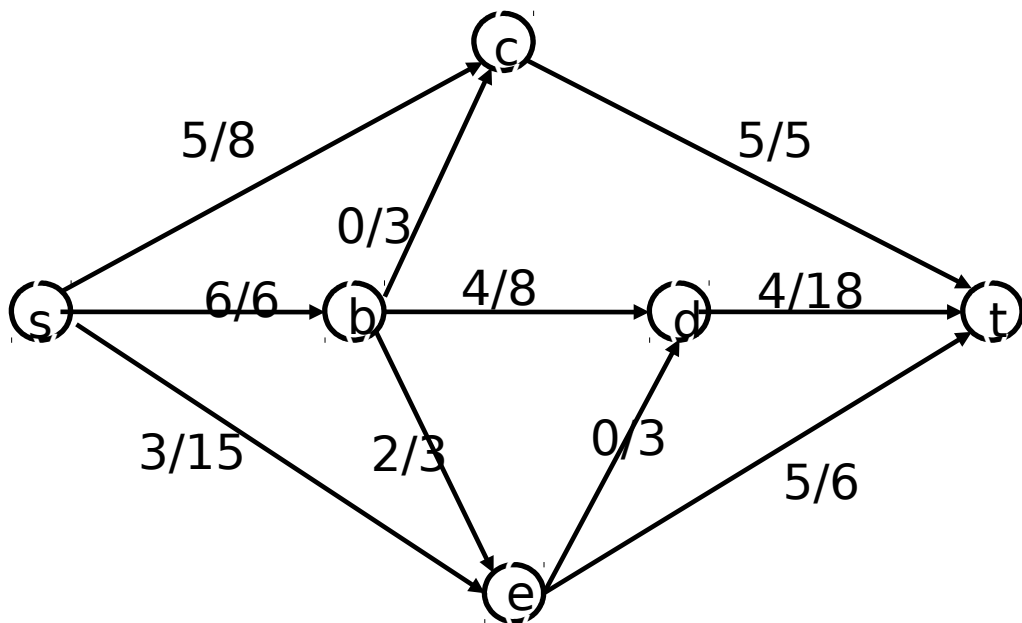
d) Si disegni la matrice prodotta dall'algoritmo di cui al punto precedente per la seguente istanza di Subset Sum: $W=7$, $w_1=3$, $w_2=6$, $w_3=1$, $w_4=2$. Si evidenzino con un cerchietto le entrate della matrice che verrebbero ispezionate dall'algoritmo che produce la soluzione ottima .

4. Massimo flusso

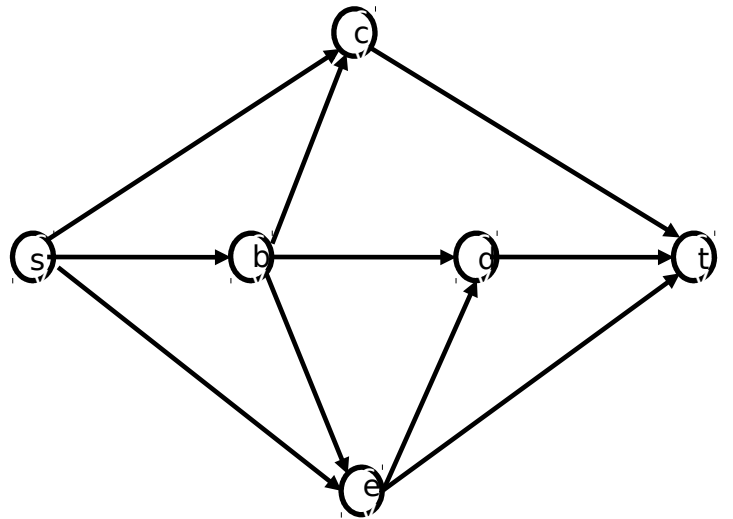
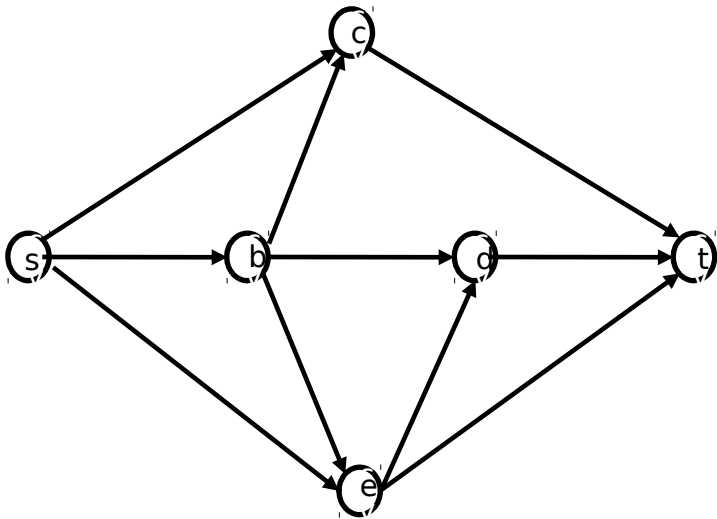
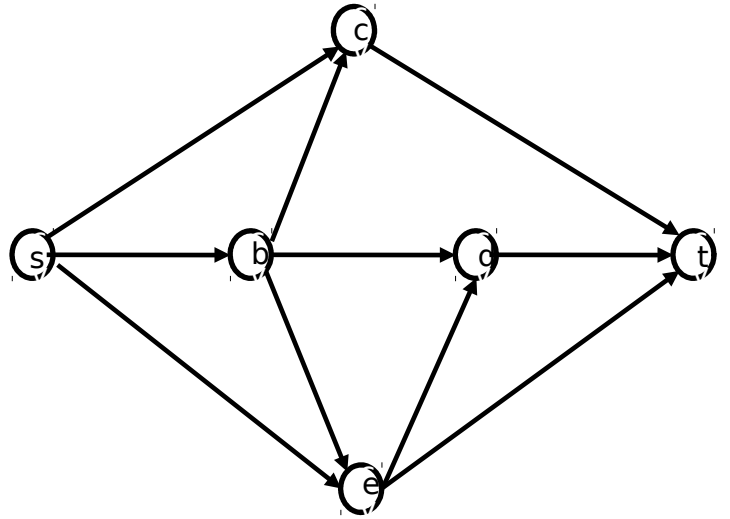
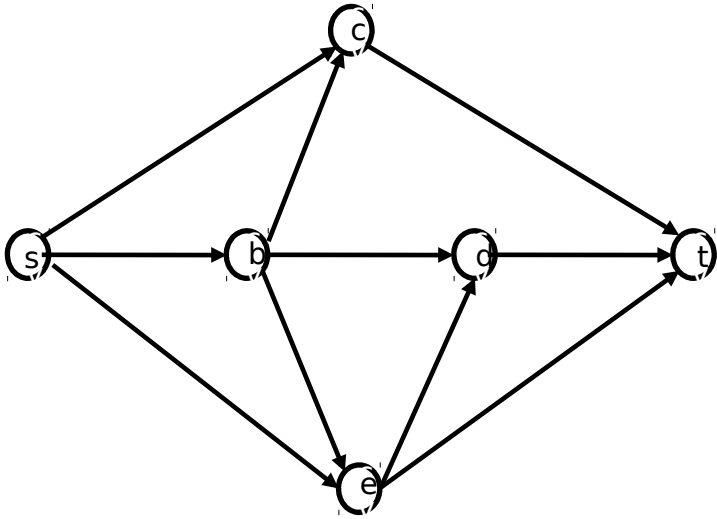
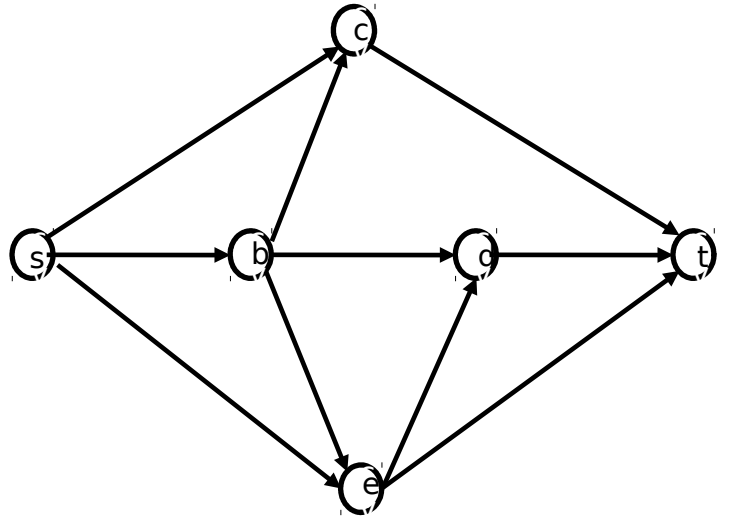
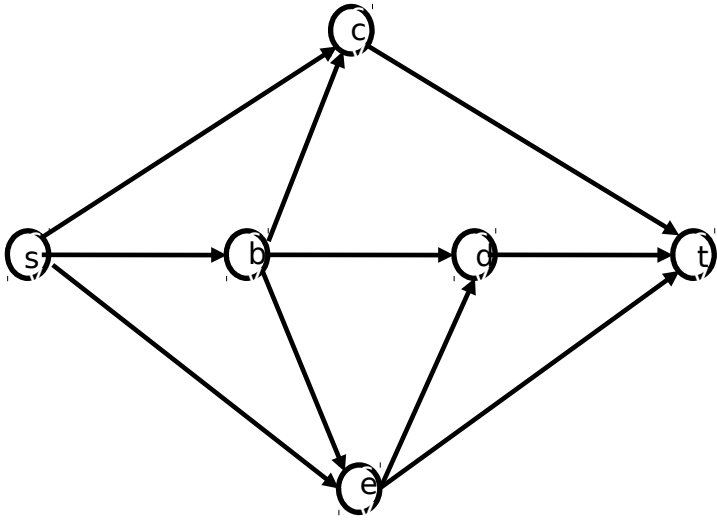
a) Si consideri la seguente rete di flusso e la funzione di flusso i cui valori sono indicati a sinistra delle capacità degli archi.

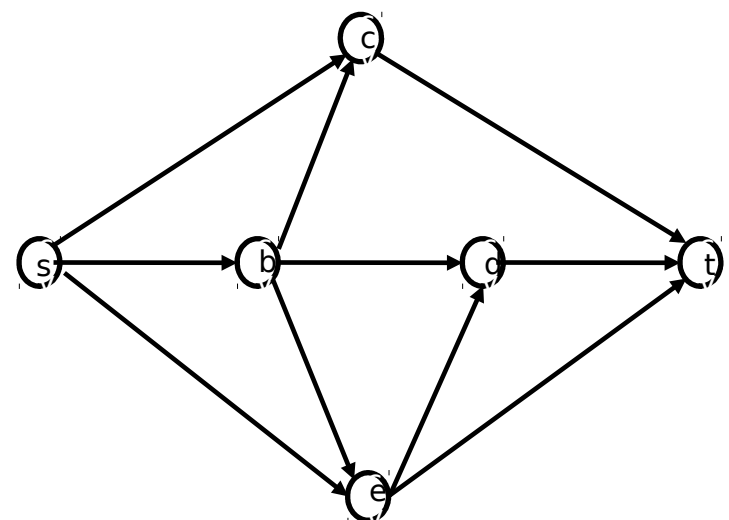
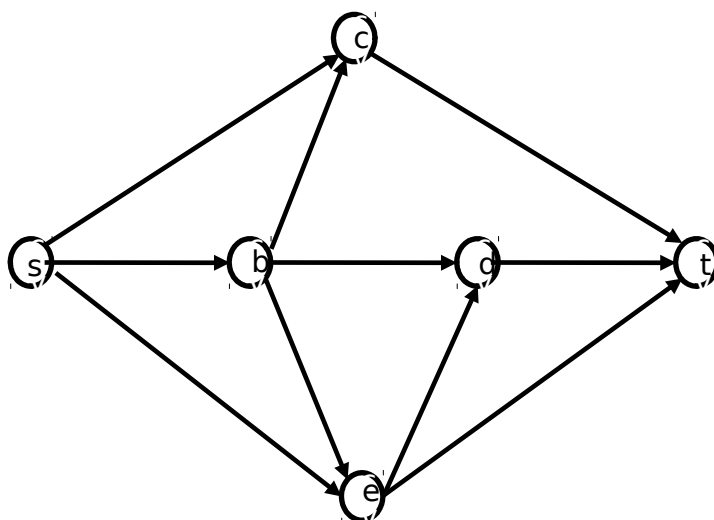
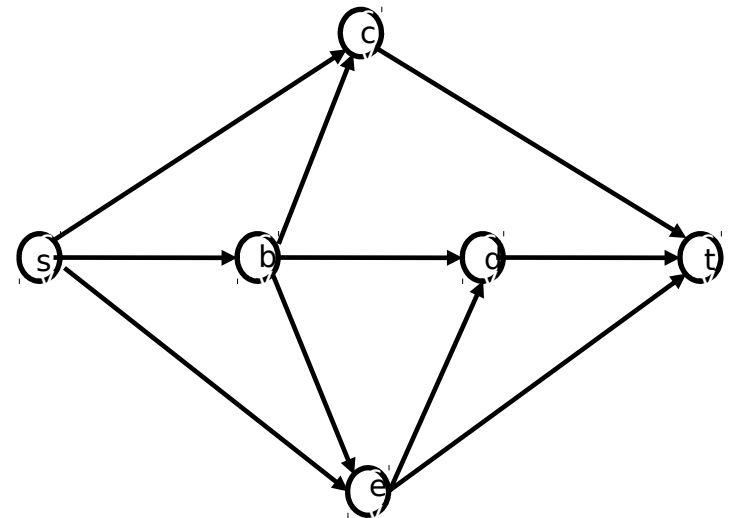
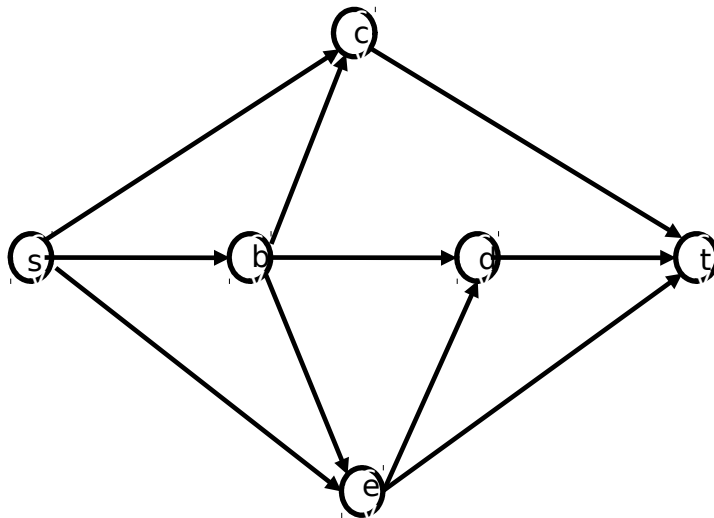
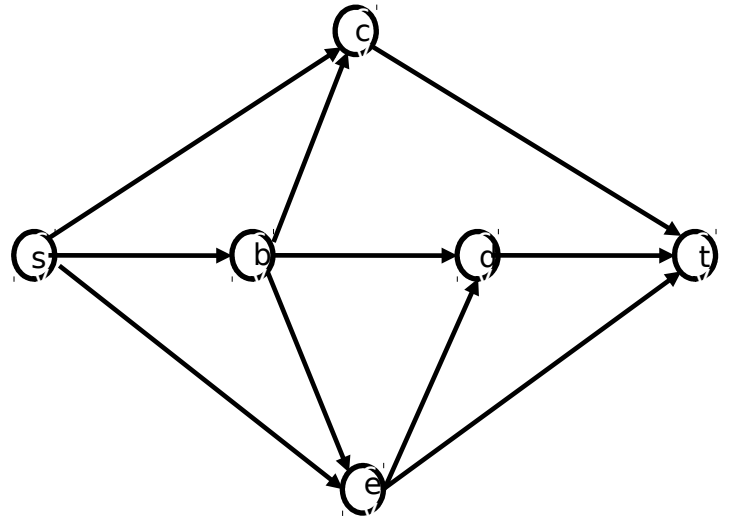
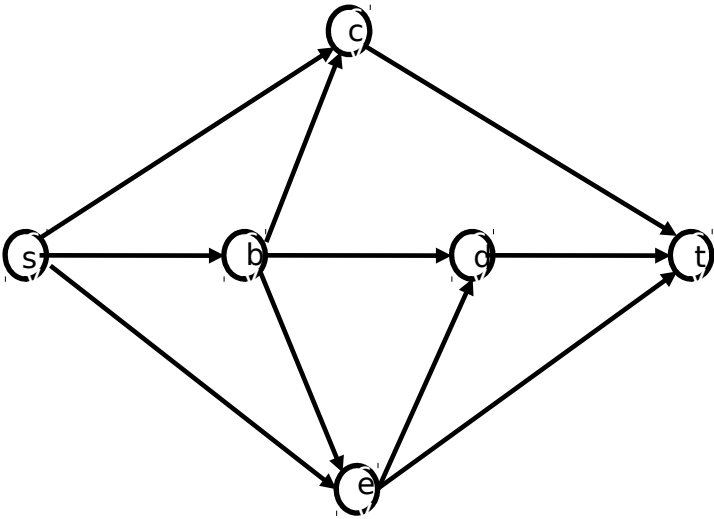
- i. Si disegni la rete residua rispetto alla funzione flusso indicata e si dica se questa funzione ha valore massimo.
- ii. Nel caso in cui la funzione non abbia valore massimo, si fornisca la funzione flusso con valore massimo applicando l'algoritmo di Ford-Fulkerson **a partire dalla funzione di flusso data. Per ogni iterazione dell'algoritmo, occorre disegnare la rete residua all'inizio di quell'iterazione, indicare il cammino aumentante scelto e mostrare il flusso associato ad ogni arco della rete di flusso originaria al termine di quella iterazione**
- iii. Si dica qual è il **valore del massimo flusso** e si fornisca **un taglio di capacità minima**.

N.B.: le risposte che non sono ottenute a partire dalla funzione di flusso data non saranno valutate.



Per vostra comodità, di seguito sono riportate diverse copie della rete di flusso, suddivise a coppie. **A partire dalla funzione di flusso data, usate l'immagine di sinistra di ciascuna coppia per disegnare la rete residua e l'immagine di destra per riportare i valori della funzione flusso assegnati a ciascun arco.** Ovviamente potrebbe essere necessario aggiungere e/o cancellare (con una x) archi nelle immagini di sinistra. Il numero di coppie non è indicativo del numero di iterazioni effettuate dall'algoritmo di Ford-Fulkerson. Procedete dall'alto verso il basso utilizzando solo le coppie di grafi che vi servono per illustrare l'intera esecuzione dell'algoritmo.





b) Si descriva il comportamento dell'algoritmo di Ford-Fulkerson e degli algoritmi da esso invocati specificando l'input e l'output sia dell'algoritmo di Ford-Fulkerson che degli altri algoritmi. Non è ne' necessario ne' sufficiente fornire lo pseudocodice.

c) 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**.

Foglio per la minuta

Foglio per la minuta

Foglio per la minuta

Foglio per la minuta