Progettazione di Algoritmi (9 CFU)
26/6/2019

Cognome e Nome:
Numero di Matricola:

Spazio riservato alla correzione

1	2	3	4	5	6	Totale
/18	/8	/20	20	/18	/16	/100

1.

a) Indicare quali delle seguenti affermazioni sono vere e quali sono false.
   1. $\log(n^n) = \Theta((\log n)^n)$
   2. $n^{1/2} = \Omega(n^{1/4})$
   3. $1000\,n + n^3 = O(n)$
   4. $1000n^4 - 100n^2 = \Omega(n^3)$
   5. $1/10\ n + 2^{\log n} = O(n)$

b) Si dimostri che se $f(n)=O(g(n))$ e $g(n)=O(h(n))$ allora $a(f(n))=O(h(n))$, dove a e` una costante. **Occorre utilizzare solo la definizione di O e nessuna altra proprieta`.**

c) Si analizzi il tempo di esecuzione nel caso pessimo del seguente segmento di codice fornendo una stima asintotica **quanto migliore e` possibile** per esso. **Si giustifichi in modo chiaro la risposta.**

```
FOR(i=n; i>0; i=i/3){

  FOR(k=2; k<m; k=k^2) {
    print(k);
  }

  print(i);
}
```

2 Si scriva lo pseudocodice di un algoritmo ricorsivo che prende in input un nodo u di un albero binario ed un elemento x. L'algoritmo effettua una visita preorder dell'albero avente come radice u e restituisce il primo nodo incontrato che contiene x. Se nessun nodo del sottoalbero contiene x allora l'algoritmo deve restituire null.
NB: L'elemento di un generico nodo v è v.dato.

3.

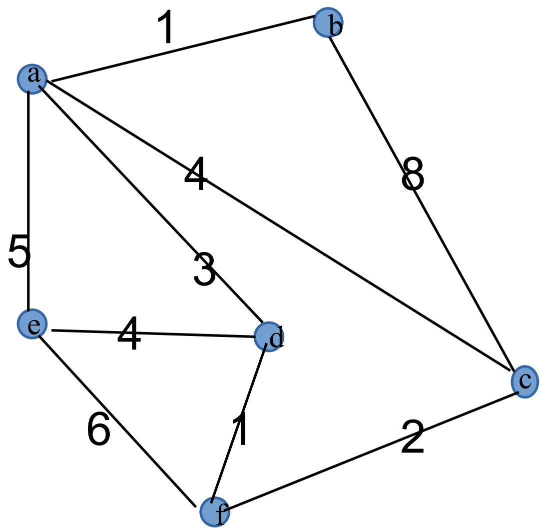
a) Si definisca formalmente il concetto di ordine topologico di un grafo direzionato acliclico. In assenza di questa definizione i punti successivi dell'esercizio non saranno valutati.

b) Scrivere lo **pseudocodice** dell'algoritmo che dato un grafo direzionato aciclico restituisce l'ordinamento topologico del grafo. **L'algoritmo deve avere tempo O(n+m).** Il voto dipendera` da quanto dettagliato sara` lo pseudocodice.

c) Si analizzi la complessita` dell'algoritmo di cui al punto b) nel caso pessimo**.** Si dimostri cioe` che e` O(n+m). **Analizzare il tempo di esecuzione di un algoritmo significa fornire una stima asintotica quanto migliore e` possibile del suo tempo di esecuzione giustificando in modo chiaro la risposta.**

4.

a) Si mostri l'esecuzione dell'algoritmo di Prim sul seguente grafo. Si assuma che l'algoritmo scelga come radice il nodo **a. Occorre mostrare per ogni passo il contenuto della coda a priorita` e l'albero ottenuto fino a quel passo. Disegnare alla fine l'albero generato.**



b) Si spieghi in che cosa consiste un'istanza (input) del problema dell'Interval Scheduling e in cosa consiste una soluzione (output) del problema. **Se dalla risposta a questo punto si evincera` che lo studente non sa in cosa consiste il problema dell'Interval Scheduling, i punti successivi del quesito non saranno valutati.**

c) Si fornisca un controesempio costituito da un'istanza **di almeno 5 attivita`** che dimostri che la strategia Fewest Conflicts non sempre fornisce la soluzione ottima al problema dell'Interval Scheduling. Si spieghi per quale motivo l'istanza da voi proposta rappresenta un controesempio al fatto che Fewest Conflicts produca una soluzione ottima al problema.

d) Si scriva lo pseudocodice dell'algoritmo greedy per il problema dell'Interval Scheduling spiegando **il significato di tutti i parametri e variabili che compaiono nello pseudocodice.** Si analizzi il tempo di esecuzione dell'algoritmo nel caso pessimo. Analizzare il tempo di esecuzione di un algoritmo significa fornire una stima asintotica quanto migliore e` possibile del suo tempo di esecuzione **giustificando in modo chiaro la risposta.**

5.

a) Si fornisca la formula utilizzata dall'algoritmo di programmazione dinamica per i cammini minimi. A tale scopo, occorre fornire:

   i. **una definizione formale di OPT(i,v)**
   ii. **formula di ricorrenza per calcolare OPT(i,v)**
   iii. **una spiegazione chiara di come la suddetta formula di ricorrenza e` ottenuta**

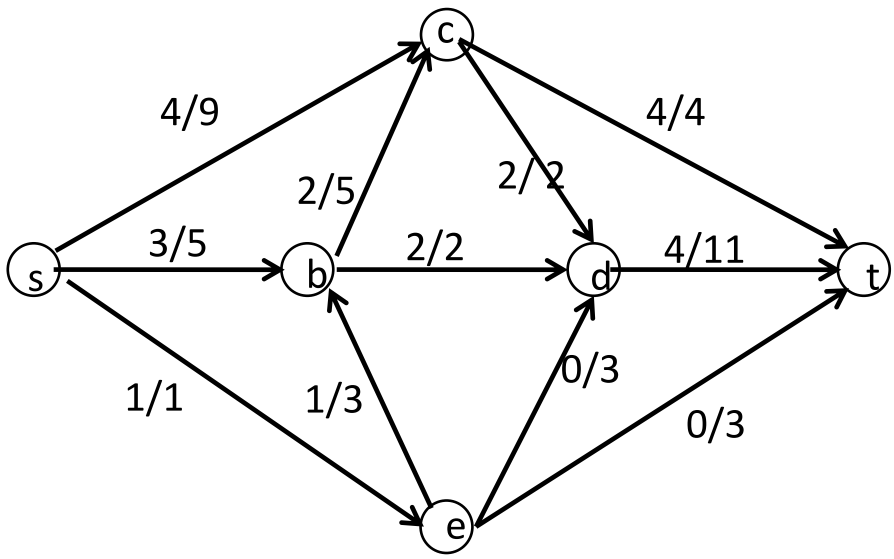
   Se non sara` fornita la definizione di cui al punto i, gli altri punti non saranno valutati.

b) Fornire lo pseudocodice dell'algoritmo di Bellman-Ford.

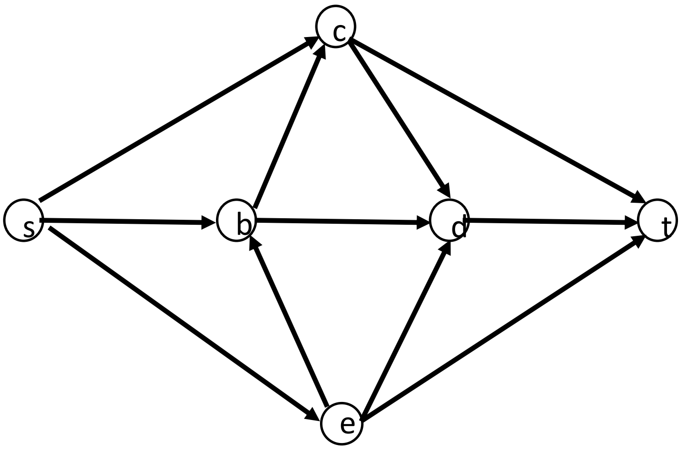
6.
a) Si consideri la seguente rete di flusso e la funzione di flusso i cui valori sono indicati a sinistra delle capacita` degli archi. Si disegni la rete residua rispetto alla funzione flusso indicata e si dica se questa funzione ha valore massimo. Nel caso in cui la funzione non abbia valore massimo, si fornisca **la funzione flusso con valore massimo e il taglio di capacita` minima.** A tal fine si eseguano una o piu` iterazioni dell'algoritmo di Ford-Fulkerson **a partire dalla funzione di flusso data.**
Per ogni iterazione dell'algoritmo, occorre

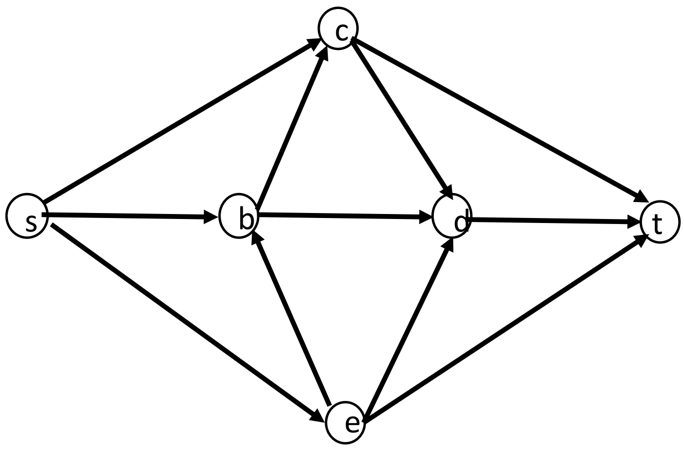
- **disegnare la rete residua all'inizio di quell'iterazione**
- **indicare il cammino aumentante da voi scelto**
- **mostrare il valore associato ad ogni arco del grafo al termine di quella iterazione**

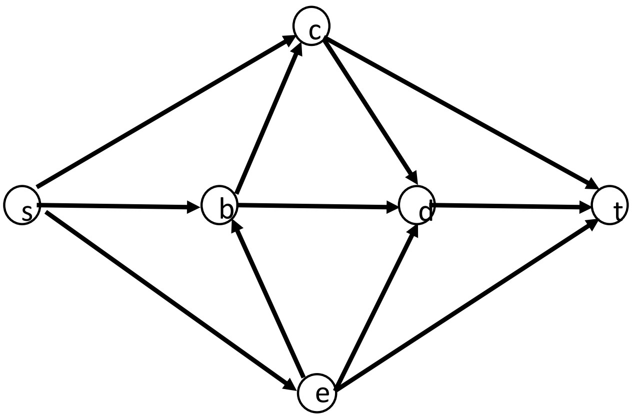
N.B.: le risposte che non sono ottenute a partire dalla funzione di flusso data non saranno valutate.

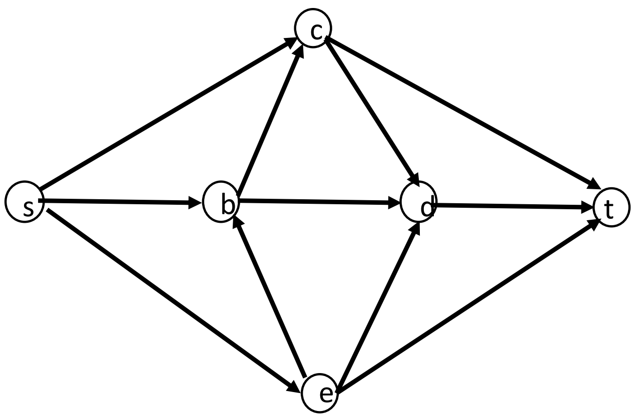


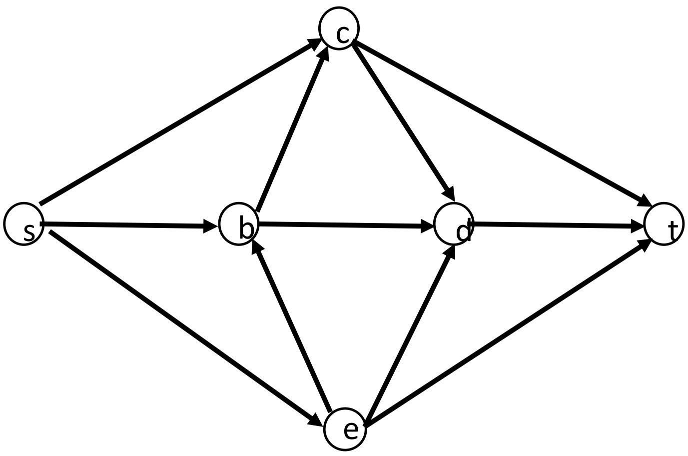
Per vostra comodita`, di seguito sono riportate diverse copie della rete di flusso, suddivise a coppie. **A partire dalla funzione di flusso data, usate l'immagine di sinistra di ciascuna coppia per disegnare la rete residua e l'immagine di destra per riportare i valori della funzione flusso assegnati a ciascun arco.** Ovviamente potrebbe essere necessario aggiungere e/o cancellare (con una x) degli archi nelle immagini di sinistra. Il numero di coppie non e` 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. **N.B.: Se non saranno rispettate queste indicazioni per lo svolgimento dell'esercizio, l'esercizio non sara` valutato.**

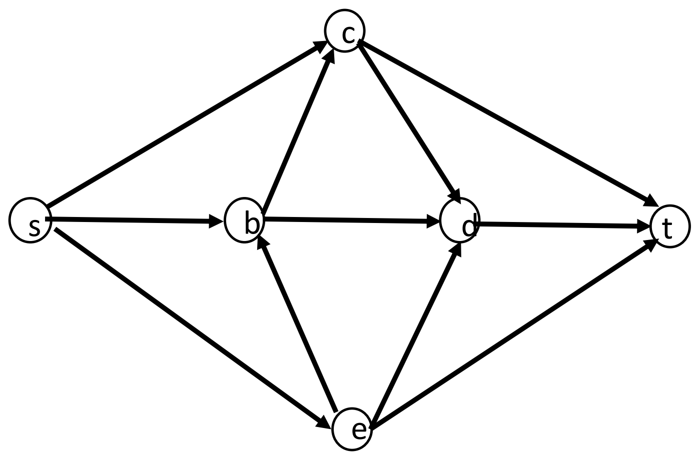

c
s
b
d
t
e


c
s
b
d
t
e


c
s
b
d
t
e


c
s
b
d
t
e


c
s
b
d
t
e


c
s
b
d
t
e

b) Si definisca il concetto di rete residua e cammino aumentante.

c) Sia G una rete di flusso e siano (A, B) un taglio s-t di G ed f una funzione di flusso per G. Si dimostri che se $v(f) = cap(A, B)$ allora f e` un massimo flusso e (A, B) e` un minimo taglio.