

Cognome Nome:

N. Matricola:

Corso: A B

Esercizio 1. [6 punti]

Indicare e risolvere la relazione di ricorrenza che descrive la complessità asintotica in tempo della seguente funzione:

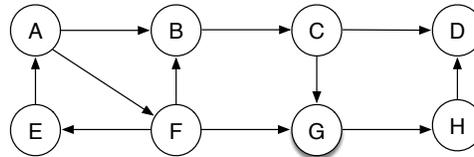
```

PIPP0(n) {
  if (n < 5) return 1;
  a = 0;
  i = 1;
  while (i < n) {
    j = 1;
    while (j < n) {j = 2*j; a++;}
    i++;
  }
  return a + Pippo(n/4) + Pippo(n/4);
}

```

Esercizio 2. [6 punti]

È dato il seguente grafo orientato, rappresentato con liste di adiacenza ordinate alfabeticamente:



1. Indicare l'ordine di visita DFS dei vertici del grafo, partendo dal vertice A .
2. Disegnare la foresta DFS ottenuta con la visita.
3. Indicare la classificazione degli archi indotta dalla visita DFS.

Esercizio 3. [6 punti]

Dato un albero binario T , definiamo per ogni nodo x le misure $s(x)$, $d(x)$, e $c(x)$:

- $s(x)$ rappresenta il numero di nodi che si incontrano partendo da x (incluso) e scendendo nell'albero sempre a sinistra, finché possibile;
- $d(x)$ rappresenta il numero di nodi che si incontrano partendo da x (incluso) e scendendo nell'albero sempre a destra, finché possibile;
- $c(x) = s(x) + d(x)$.

Scrivere lo pseudocodice di un algoritmo ricorsivo che stampi il valore $c(x)$ per ogni nodo x dell'albero T , e valutarne la complessità.

Esercizio 4. [6 punti]

Si indichi la complessità in tempo al caso medio della ricerca con insuccesso all'interno di una tabella hash con n elementi e dimensione m , in cui le collisioni sono gestite con concatenamento, e se ne dimostri la correttezza.

Esercizio 5. [6 punti]

Si consideri il seguente problema (SUBSETSUM): Dato un insieme di n interi positivi $A = \{a_1, \dots, a_n\}$, determinare se esiste un sottoinsieme $A' \subseteq A$ tale che la somma degli elementi in A' abbia valore k . Si fornisca un algoritmo enumerativo che trovi la soluzione, se esiste. Si discuta la complessità dell'algoritmo.