

INFORMATICA GENERALE

Secondo Esonero e Compito Scritto

docenti: TIZIANA CALAMONERI, IVANO SALVO
Sapienza Università di Roma

24 giugno 2014

Coloro che intendono partecipare alla prova di esonero devono svolgere gli esercizi 2 e 3 ed hanno a disposizione un'ora e mezza; coloro che, invece, intendono partecipare all'intero compito scritto devono svolgere tutti e tre gli esercizi in due ore. Il voto di esonero verrà poi normalizzato in 30esimi.

Esercizio 1 (10 punti) Si consideri la seguente funzione `THeight()` che, preso in input un albero rappresentato tramite vettore dei padri T (con nodi numerati da 0 a $n - 1$), calcola l'altezza dell'albero.

```
int THeight(int T[], int n){
    int i, j, h, height=0;

    for (i=0; i<n; i++){
        h=0; j=i;
        while (T[j]!=-1){
            j=T[j];
            h++;
        } //end while
        if (h > height) height=h;
    } // end for
    return height;
}
```

1. **(1 punto)** Si mostri un esempio di funzionamento della funzione `THeight` su un albero con un valore di n almeno pari a 6;

2. **(3 punti)** Si stimi il costo computazionale, nel caso peggiore, in funzione di n , giustificando accuratamente la risposta.
3. **(3,5+2,5 punti)** Si calcoli il costo computazione stretto nel caso in cui:
 - T è un albero binario completo;
 - T è un cammino di n nodi.

Esercizio 2 (10 punti) Una sequenza s_1 è *immersa* in s_2 se gli elementi di s_1 occorrono ordinatamente (non necessariamente in modo consecutivo) in s_2 . Ad esempio, la sequenza di interi $\langle 1, 4, 7 \rangle$ è immersa nella sequenza $\langle 2, 1, 3, 4, 6, 7, 9 \rangle$, ma non nella sequenza $\langle 7, 2, 1, 4, 3 \rangle$ (i valori 1, 4, 7 non sono presenti nel giusto ordine).

- Dare una definizione induttiva della relazione di immersione mediante equazioni ricorsive.
- Scrivere una funzione *C ricorsiva* `int immersaRec(lista L, lista M)` che restituisca 1 se la sequenza di interi rappresentata dalla lista L è immersa nella sequenza rappresentata dalla lista M.
- Scrivere una funzione *C iterativa* `int immersaIt(lista L, lista M)` che restituisca 1 se la sequenza di interi rappresentata dalla lista L è immersa nella sequenza rappresentata dalla lista M.

Esercizio 3 (10 punti) Si progetti un algoritmo che, dato un grafo tramite la sua matrice di adiacenza, restituisca 1 se esistono due nodi con lo stesso grado e 0 altrimenti.

Dell'algoritmo proposto:

1. **(3 punti)** Si dia la descrizione a parole, dettagliando eventuali strutture dati ausiliarie utilizzate;
2. **(4 punti)** Si scriva lo pseudocodice;
3. **(3 punti)** Si calcoli il costo computazionale;
4. **(facoltativo: fino a + 2 punti)** Si discuta come cambia il costo computazionale se il grafo in input è dato tramite le altre memorizzazioni viste a lezione.