

# INFORMATICA GENERALE

## Esame Scritto

docenti: G. BONGIOVANNI, T. CALAMONERI, I. SALVO  
Sapienza Università di Roma

19 settembre 2018

**Esercizio 1 (10 punti)** Si consideri l'algoritmo di Merge Sort. Di esso:

1. **(2 punti)** Si dia la descrizione a parole, inclusa la funzione di fusione;
2. **(2 punti)** Si scriva lo pseudocodice (inclusa la funzione di fusione);
3. **(1 punto)** Si mostri in dettaglio l'esecuzione dello pseudocodice sul vettore  $[6, 2, 3, 4, 5, 1]$ , specificando come cambia il valore delle variabili introdotte nello pseudocodice;
4. **(1 punto)** Si scriva l'equazione di ricorrenza che ne descrive il costo computazionale, giustificandola;
5. **(2+2 punti)** La si risolva usando il metodo iterativo ed il metodo per sostituzione, dettagliando i passaggi.

**Esercizio 2 (10 punti)** Un *polinomio* in una variabile può essere rappresentato come una lista di *monomi*. Un monomio deve memorizzare semplicemente il coefficiente e il grado. Quindi ad esempio, il monomio  $3x^2$  (supponiamo la variabile sia chiamata  $x$ ) sarà semplicemente rappresentato dalla coppia di numeri 3 e 2. Il monomio 5, semplicemente dalla coppia di numeri 5 e 0.

I polinomi dovrebbero essere sempre rappresentati in una forma canonica, con un unico monomio per ciascun grado e con i gradi ordinati in modo crescente. Definire:

1. **(3 punti)** in linguaggio C i tipi di dato `monomio` (usando opportunamente una `struct`) e `polinomio` (modificando opportunamente il tipo `list`) visto a lezione.

2. (**3 punti**) una funzione di stampa di un polinomio. Ad esempio se la lista di ingresso è  $\langle(1, 0), (2, 1), (1, 2)\rangle$ , la funzione deve stampare la sequenza di caratteri  $x^2 + 2x + 1$  (osservate che la stampa è rovesciata rispetto alla lista).
3. (**4 punti**) una funzione C di prototipo:

```
polinomio sommaPolinomi(polinomio, polinomio)
```

che calcola la somma tra due polinomi.

Il polinomio risultato dovrà essere *allocato durante il calcolo* e i due polinomi di input non dovranno essere modificati.

Anche il risultato dovrà essere esso stesso rappresentato in forma canonica.

**Esercizio 3 (10 punti)** Sia dato un albero binario  $T$  memorizzato tramite record e puntatori. Si progetti un algoritmo che, dato in input il puntatore alla radice di  $T$  ed un intero positivo  $h$ , restituisca TRUE se l'altezza di  $T$  è al più  $k$ , FALSE altrimenti (dove l'altezza è intesa come il massimo numero di archi tra la radice ed una foglia).

Dell'algoritmo proposto:

1. (**2,5 punti**) si dia la descrizione a parole;
2. (**3,5 punti**) si scriva lo pseudocodice;
3. (**2 punti**) si calcoli il costo computazionale, dettagliando comunque il costo delle funzioni studiate eventualmente usate;
4. (**2 punti**) si mostri il dettaglio del funzionamento dell'algoritmo proposto su un albero di esempio con almeno 9 nodi,  $h = 4$  tale che l'algoritmo restituisca FALSE.