

INFORMATICA GENERALE

Sessione Autunnale

Esame Scritto

docenti:

GIANCARLO BONGIOVANNI, TIZIANA CALAMONERI, IVANO SALVO
Sapienza Università di Roma
4 Settembre 2015

Esercizio 1 (10 punti) Si consideri la seguente funzione in pseudocodice, che prende come parametri un vettore V di interi e due indici *primo* e *ultimo* che alla prima chiamata valgono 1 ed n rispettivamente:

```
funzione MagicFunction (V: vettore; primo, ultimo: intero)
if (primo >= ultimo) return
for i = primo to ultimo
    for j = primo to ultimo
        for h = primo to ultimo
            V[i] <- V[i] - 2V[j] + V[h]
medio <- [(primo + ultimo)/2]
MagicFunction(V, primo, medio)
MagicFunction(V, medio+1, ultimo)
return
```

Da essa si ricavi l'equazione di ricorrenza che ne esprime il costo computazionale, specificando i contributi delle varie istruzioni (**1 punto**). Inoltre, si risolva l'equazione di ricorrenza trovata utilizzando:

- (**3 punti**) il metodo iterativo;
- (**2 punti**) il metodo dell'albero;
- (**1 punto**) il metodo principale;
- (**3 punti**) il metodo di sostituzione.

In tutti i casi, si dettagli il procedimento usato e si giustifichino le proprie risposte.

Esercizio 2 (10 punti) Il tipo `listaTerne` è un tipo che definisce una lista in cui ciascun nodo contiene tre campi interi. Un *massimo locale* in una matrice di interi è un elemento che è maggiore o uguale a tutti gli elementi adiacenti (in tutte le direzioni). Osservate che gli elementi adiacenti sono 8 per i punti interni della matrice, 5 per gli elementi sui bordi e 3 per quelli agli angoli.

Dare prima la definizione C del tipo `listaTerne` e scrivere poi una funzione C di prototipo:

```
listaTerne massimiLocali(int** M, int m, int n)
```

che, presa in input una matrice di interi M di dimensione $m \times n$, costruisce una lista contenente tutti i massimi locali della matrice. Ogni nodo, oltre al valore del massimo, dovrà registrare gli indici riga e colonna del massimo locale.

Esercizio 3 (10 punti) Siano dati un vettore $A[1..n]$ di n caratteri ed un intero $k \in [0, \dots, n - 1]$. Una *k-rotazione sinistra* di A è il vettore ottenuto:

- considerando il vettore stesso chiuso circolarmente (ossia, la posizione $A[n]$ si considera a sinistra della posizione $A[1]$);
- spostando a sinistra tutti gli elementi di A di k posizioni.

Ad esempio, una 3-rotazione sinistra del vettore ABCDEFGH produce il vettore DEFGHABC.

Si progetti un algoritmo (il più efficiente possibile sia dal punto di vista del tempo che dello spazio) che prende in input A e k , e restituisce il vettore A dopo aver eseguito su di esso una k -rotazione sinistra.

Dell'algoritmo progettato:

- (3 punti)** Si scriva l'idea a parole;
- (4 punti)** Si dia lo pseudocodice;
- (2+1 punti)** Si calcoli il costo computazionale sia in termini di tempo che di spazio.