

Introduzione agli Algoritmi

Esame Scritto a canali unificati

docenti: T. Calamoneri, A. Monti
Sapienza Università di Roma
21 Marzo 2024

Esercizio 1 (10 punti):

Si consideri la seguente funzione:

```
def Exam( $n$ ):  
    if  $n \leq 1$ : return 5  
     $x = s = 1$   
    while  $3 * x \leq n$ :  
         $x += 1$   
         $s += 2$   
     $s += 2 * \text{Exam}(x) + 2 * \text{Exam}(n - 2 * x)$   
    while  $x > 0$ :  
         $n -= 2$   
         $x -= 1$   
     $s += \text{Exam}(n)$   
    return  $s$ 
```

- Si imposti la relazione di ricorrenza che ne definisce il tempo di esecuzione giustificando dettagliatamente l'equazione ottenuta.
- Si risolva la ricorrenza usando due metodi a scelta, dettagliando i passaggi del calcolo e giustificando ogni affermazione.

Esercizio 2 (10 punti): Sia dato un array A di $n \geq 1$ interi distinti i cui elementi sono ordinati in modo crescente nella parte sinistra ed in modo decrescente nella parte destra. Progettare un algoritmo *iterativo* che restituisca il massimo contenuto nell'array in tempo $O(\log n)$.

Ad esempio, per l'array $A = [8, 10, 20, 80, 100, 200, 400, 500, 180, 30, 2, 1]$ l'algoritmo deve restituire il valore 500.

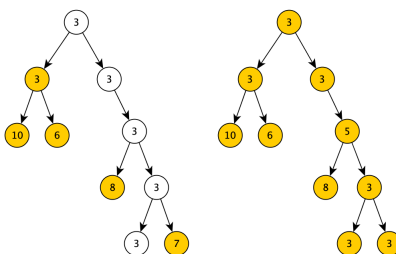
Dell'algoritmo proposto:

- a) si scriva lo pseudocodice opportunamente commentato;
- b) si giustifichi il costo computazionale.

Esercizio 3 (10 punti):

Dato il puntatore r al nodo radice di un albero binario non vuoto con nodi contenenti valori interi, progettare un algoritmo *ricorsivo* che in tempo $\Theta(n)$ determini se l'albero contiene un cammino radice-foglia caratterizzato dalla proprietà di avere nodi tutti dello stesso valore.

Ad esempio, per l'albero in figura a sinistra, l'algoritmo deve rispondere *True* (per via del cammino evidenziato in bianco) mentre per l'albero in figura a destra la risposta deve essere *False*.



L'albero è memorizzato tramite puntatori a record di tre campi: il campo *key* contenente un valore intero ed i campi *left* e *right* con i puntatori al figlio sinistro e al figlio destro, rispettivamente (questi puntatori valgono *None* in mancanza del figlio).

Dell'algoritmo proposto:

- a) si scriva lo pseudocodice opportunamente commentato;
- b) si giustifichi il costo computazionale.

NOTA BENE: nello pseudocodice dell'algoritmo ricorsivo **non** si deve far uso di variabili globali.