

Introduzione agli Algoritmi
Appello esame del 13 gennaio 2016
Prof. Emanuela Fachini (canale 1) e Prof. Irene Finocchi (canale 2)

Parte 1

Le risposte non motivate non saranno prese in considerazione. Negli esercizi di progettazione, prima di passare allo pseudocodice descrivete l'idea algoritmica sottostante. Per tutti gli algoritmi progettati è inoltre necessario analizzare tempo di esecuzione e correttezza.

Esercizio 1

Si descriva l'algoritmo QuickSort e se ne analizzi il tempo di esecuzione nei casi peggiore e migliore.

Esercizio 2

Si consideri la seguente funzione:

```
fun (array A, int i, int f) {  
    n = f-i+1  
    t=n3;  
    while t ≥ 2 do t = t-4;  
    if (n ≤ 1) then return 1;  
    else return fun (A,i,i+(f-i)/4) + fun (A,f-(f-i)/4,f);  
}
```

Qual è il tempo di esecuzione di *fun* in funzione di *n*? Si imposti e si risolva la relazione di ricorrenza.

Esercizio 3

Sia *A* un array di *n* interi positivi tale che la differenza, in modulo, tra due qualsiasi elementi di *A* è sempre $\leq k$, con *k* costante. Si determini quanti elementi distinti al più può contenere l'array, e si delinei un algoritmo che lo ordini con un tempo di esecuzione $O(k \cdot n) = O(n)$.

Si determini poi un algoritmo che fornisca due elementi di *A* aventi differenza minima.

L'algoritmo complessivamente deve operare in tempo $O(k \cdot n) = O(n)$ e può ricevere in ingresso il valore *k*. Si discutano la correttezza e il tempo di esecuzione dell'algoritmo proposto.

Esempio: con $k = 4$ per esempio l'array potrebbe essere

4	4	7	6	7	4	6	4	7	6	4	4	7
---	---	---	---	---	---	---	---	---	---	---	---	---

Introduzione agli Algoritmi
Appello esame del 13 gennaio 2016
Prof. Emanuela Fachini (canale 1) e Prof. Irene Finocchi (canale 2)

Parte 2

Le risposte non motivate non saranno prese in considerazione. Negli esercizi di progettazione, prima di passare allo pseudocodice descrivete l'idea algoritmica sottostante. Per tutti gli algoritmi progettati è inoltre necessario analizzare tempo di esecuzione e correttezza.

Esercizio 4

Si dimostri che un albero AVL con n nodi ha altezza in $O(\log n)$. La parte della dimostrazione relativa agli alberi di Fibonacci deve essere dettagliata in tutti i suoi passaggi.

Esercizio 5

Considerare il seguente algoritmo, che ricerca il minimo in un array A di dimensione n :

```
algoritmo minimo (array A, intero n)
  minIndex = 1
  for i = 2 to n
    if A[minIndex] > A[i] then minIndex = i
  return A[minIndex]
```

Disegnare l'albero di decisione che descrive le possibili esecuzioni dell'algoritmo minimo nel caso $n = 4$. Qual è il numero di confronti eseguiti nel caso peggiore? Qual è un caso peggiore?

Esercizio 6

Sia T un albero binario con n nodi. Si devono determinare due elementi di T la cui somma è minima. Si descrivano algoritmi, il più possibile efficienti, nei seguenti casi:

- T è un albero binario qualunque
- T è un albero binario di ricerca
- T è un albero AVL

In ogni caso si discutano la correttezza e il tempo di esecuzione dell'algoritmo proposto.