Introduzione agli Algoritmi
Prova intermedia - 14 Aprile 2014
Prof. Emanuela Fachini (canale 1) e Prof. Irene Finocchi (canale 2)

Le risposte non motivate non saranno prese in considerazione.

Negli esercizi di progettazione, prima di passare allo pseudocodice descrivete l'idea algoritmica soggiacente. Per tutti gli algoritmi progettati è necessario analizzare tempo di esecuzione e correttezza.

Testo 4

Esercizio 1

Si consideri la seguente funzione:

```
test (intero n)
        if n≤64 then return 1
        k = n*n
        while k ≥ 1 do k=k/2
        return 1 + test(n/4)
```

Scrivere la relazione di ricorrenza che descrive il tempo di esecuzione T(n) della funzione *test* e dimostrare che $T(n)=O(\log^2 n)$.

Esercizio 2

Definiamo il problema del *punto fisso* come segue:

data una sequenza ordinata di n interi distinti, sia positivi che negativi, $a_1 < a_2 < \ldots < a_n$, *determinare se esiste un indice i tale che* $a_i = i$.

Progettare un algoritmo che risolva il problema del punto fisso in tempo O(log n). Analizzare sia la correttezza dell'algoritmo proposto che il tempo di esecuzione nel caso peggiore.

Esercizio 3

Quali delle seguenti affermazioni sono vere? Motivare la risposta con una dimostrazione o con un opportuno esempio.

Introduzione agli Algoritmi
Prova intermedia - 14 Aprile 2014
Prof. Emanuela Fachini (canale 1) e Prof. Irene Finocchi (canale 2)

1. Siano f(n), g(n), t(n) e s(n) quattro funzioni tali che: $f(n)=O(g(n))$ e $t(n)=O(s(n))$. Allora $\max\{ f(n), t(n)\} = O(g(n)+s(n))$

2. Siano f(n), g(n), t(n) e s(n) quattro funzioni tali che: $f(n)=O(g(n))$ e $t(n)=O(s(n))$. Allora $f(n)+g(n)+t(n) = O(s(n))$

3. Sia f(n) il tempo di esecuzione nel caso peggiore di un algoritmo che risolve un problema $\mathscr{P}$ in modo ottimo. Se $\mathscr{P}$ ha una complessità $\Omega(g(n))$, allora deve essere $f(n)= \Theta(g(n))$

4. Sia f(n) il tempo di esecuzione nel caso migliore di un algoritmo che risolve un problema $\mathscr{P}$. Se $\mathscr{P}$ ha una complessità $\Omega(g(n))$, allora può accadere che $f(n)=O(g(n))$ ma $f(n)\neq\Theta(g(n))$.