

Introduzione agli Algoritmi  
Appello del 13 febbraio 2014  
Proff. Emanuela Fachini ed Irene Finocchi

*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 necessario analizzare tempo di esecuzione e correttezza (canale 1: usare gli invarianti per progettare e verificare il ciclo in caso di algoritmi iterativi).*

### Esercizio 1 (punti 14)

A) Si consideri la seguente funzione:

```
fun (intero k) {
    i=0
    a=1
    while (a ≤ k) {
        for b=1 to a do i++
        a= 2*a
    }
}
```

Qual è il tempo di esecuzione  $F(k)$  di fun?

B) Qual è il tempo di esecuzione  $T(n)$  della seguente funzione ricorsiva? Scrivere la relazione di ricorrenza e risolverla, usando il metodo dell'albero della ricorsione per prevedere l'andamento asintotico della soluzione e dimostrando la verità della previsione per induzione.

```
test(n) {
    if n ≤ 10 then return fun(n)
    for i = 1 to 3 do test(n/6)
}
```

### Esercizio 2 (punti 16)

Sia  $A$  un array di  $n$  valori *interi distinti* presi nell'intervallo  $[0, n]$ .  $A$  è ordinato in ordine crescente. Si progetti e si analizzi un algoritmo che, preso in input  $A$ , restituisca l'intero in  $[0, n]$  mancante in  $A$ . L'algoritmo dovrebbe avere tempo di esecuzione  $O(\log n)$  nel caso peggiore.

Introduzione agli Algoritmi  
Appello del 13 febbraio 2014  
Proff. Emanuela Fachini ed Irene Finocchi

*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 (canale 1: usare gli invarianti per progettare e verificare il ciclo in caso di algoritmi iterativi).*

**Esercizio 3** (punti 14)

Si risponda in modo sintetico e preciso alle seguenti domande:

1. Si dia la definizione di ABR (albero binario di ricerca).
2. Si dica quando un ABR è rosso-nero (canale 1) o AVL (canale 2).
3. Si dia un esempio di ABR che non può essere (??) rosso-nero / AVL.
4. Dimostrare che l'altezza di un albero rosso-nero / AVL contenente  $n$  nodi è  $O(\log n)$ .

**Esercizio 4** (punti 16)

Progettare un algoritmo *TreeInterval* che, dati un albero binario di ricerca  $T$  contenente  $n$  chiavi intere e due chiavi  $min$  e  $max$  (entrambe presenti nell'albero), stampi tutte le chiavi di  $T$  comprese tra  $min$  e  $max$ .

Si supponga che ogni nodo, oltre alla chiave, contenga i puntatori al figlio sinistro, al figlio destro e al padre.

Si discuta la correttezza dell'algoritmo proposto. Si analizzi inoltre il tempo di esecuzione, in funzione del numero  $n$  di nodi dell'albero e del numero  $k$  di chiavi stampate, supponendo che l'albero  $T$  sia bilanciato.