

INFORMATICA GENERALE

Sessione Estiva

Esame Scritto

docenti:

GIANCARLO BONGIOVANNI, TIZIANA CALAMONERI, IVANO SALVO
Sapienza Università di Roma
8 luglio 2015

Esercizio 1 (10 punti) Si consideri la seguente funzione in pseudocodice:

```
funzione VectorReloaded (V: vettore; n: intero)
if (n<=1) return
for h = 1 to 3
    VectorReloaded(V, n/9)
for i = 1 to n
    V[i] ← V[i]+V[n-i+1]
return
```

Da essa si ricavi l'equazione di ricorrenza che ne esprime il costo computazione (**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 è necessario dettagliare il procedimento usato e giustificare le proprie risposte.

Esercizio 2 (10 punti) Sia P una lista *non vuota* che contiene ordinatamente i primi k numeri primi ($k > 0$). Scrivere il codice di una funzione C di prototipo:

```
int ennesimoPrimo(lista P, int n);
```

che restituisce l'ennesimo numero primo implementando la seguente strategia: se $n \leq k$, si limita a ritornare il contenuto dell' n -esimo nodo di P . Altrimenti, oltre a ritornare l' n -esimo numero primo, modifica P aggiungendovi in coda altri $n - k$ numeri primi.

Esercizio 3 (10 punti) Un albero si dice *ternario* se ogni nodo ha al più 3 figli: un figlio sinistro, un figlio centrale ed un figlio destro. Nel caso di albero ternari, dati due nodi v ed u sullo stesso livello, diciamo che v è *alla sinistra di* u se v precede u nella visita in pre-ordine.

Si consideri un albero ternario completo di altezza h con n nodi definito con le seguenti proprietà:

(1) per $0 \leq i < h$ ci sono 3^i nodi al livello i ;

(2) tutti i nodi interni al livello $h - 1$ sono alla sinistra delle eventuali foglie che stanno a quel livello e hanno tutti 3 figli tranne, eventualmente quello più a destra (cioè l'albero ternario è completo o quasi completo).

a. (**2 punti**) Dimostrare che $h = \Theta(\log n)$.

b. (**3 punti**) Descrivere un'implementazione efficiente dell'albero ternario completo tramite un vettore (notazione posizionale).

c. (**5 punti**) Si supponga di memorizzare n valori distinti nei nodi dell'albero con la proprietà che un nodo ha chiave minore di quelle dei suoi figli. Si ottiene così uno *heap ternario*. Descrivere a parole e tramite pseudocodice un'implementazione efficiente dell'algoritmo di estrazione del massimo sull'heap ternario e se ne valuti il costo computazionale.