

INFORMATICA GENERALE
Secondo Appello Sessione Estiva
Esame Scritto

docenti: GIANCARO BONGIOVANNI, TIZIANA CALAMONERI, IVANO SALVO
Sapienza Università di Roma

19 Luglio 2018

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

```
test (intero  $n$ )  
 $k \leftarrow 0$ ;  
for  $i = 1$  to  $n$  do  
     $k++$ ;  
if  $n < 2$  return  $k$  else return test( $n/2$ ) + test( $n/4$ ).
```

Si ricavi l'equazione di ricorrenza che ne descrive il costo computazionale, spiegando il ragionamento usato (**2 punti**), e:

- a. (**5 punti**) ci si avvalga del metodo iterativo e di opportune maggiorazioni per stimare superiormente ed inferiormente la soluzione dell'equazione;
- d. (**3 punti**) ci si avvalga del metodo di sostituzione per trovare la soluzione esatta dell'equazione.

In tutti i casi, si dettagli il procedimento usato e si giustifichino le proprie risposte.

Esercizio 2 (10 punti) Considerare l'algoritmo di ordinamento *Selection Sort* che ordina un vettore selezionando iterativamente i minimi della porzione di vettore non ancora ordinata.

1. Scrivere una funzione C di prototipo: `void listSelectionSort(list L)` che, ricevendo come parametro di ingresso una lista di interi L la riordina, applicando l'algoritmo *Selection Sort* adattato alle liste, selezionando iterativamente i minimi della coda non ancora ordinata della lista e *scambiando i campi informazione* della lista (i nodi allocati in memoria, quindi, rimangono esattamente quelli originari di L).

2. Scrivere una funzione C di prototipo: `list listSelectionSortFun(list L)` che, ricevendo come parametro di ingresso una lista di interi L la riordina, applicando l'algoritmo *Selection Sort* adattato alle liste, questa volta generando una *nuova lista*. Organizzare il codice in modo da mantenere la complessità quadratica dell'algoritmo.

[Lo svolgimento di un solo punto vale **6** punti, mentre lo svolgimento di entrambi vale **10** punti]

Esercizio 3 (10 punti) Dati due alberi binari T_1 e T_2 memorizzati tramite records e puntatori e dati tramite i puntatori r_1 ed r_2 alle loro radici, si progetti un algoritmo che, presi in input r_1 ed r_2 restituisca 1 se T_1 e T_2 sono uguali (ossia se hanno sia la stessa struttura che le stesse chiavi nella stessa posizione).

Di tale algoritmo:

1. **(3 punti)** Si illustri a parole l'idea entrando nei dettagli;
2. **(3 punti)** Si scriva lo pseudocodice;
3. **(2 punti)** Si calcoli il costo computazionale (utilizzando le equazioni di ricorrenza qualora l'algoritmo sia ricorsivo);
4. **(2 punti)** Si valuti come cambia il tempo di esecuzione nel caso in cui T_1 e T_2 siano memorizzati usando invece la notazione posizionale.