

008AA – ALGORITMICA E LABORATORIO
Primo Compitino 3 Aprile 2013

Cognome Nome:

N. Matricola:

Corso: A B

Esercizio 1. (*2+2 punti*) Si consideri la complessità di QUICKSORT nel caso che sia applicato a un array contenente tutti elementi con lo stesso valore.

- (a) Si imposti l'equazione di ricorrenza associata.
- (b) Si risolva con il metodo iterativo/delle sostituzioni.

Esercizio 2. (*6+2 punti*) Si supponga di avere una funzione DIZIONARIO(S) che dato un array S di n caratteri, restituisca vero o falso a seconda che S contenga o meno una parola della lingua italiana. Si scriva il codice di un algoritmo che conti gli anagrammi di S , ovvero un algoritmo che prenda in input un array di n caratteri (senza spazi) e che restituisca il numero di parole italiane ottenute come anagrammi di S . Si discuta la complessità dell'algoritmo proposto.

Esercizio 3. (*3+6+3 punti*) Dato un array a di n interi **distinti** e **ordinati**, un elfo dispettoso decide di scambiare una coppia di elementi distinti in a .

1. Dato a , **descrivere** un algoritmo di complessità in tempo $O(n)$ per individuare le due posizioni i e j in cui è avvenuto lo scambio. Commentare la soluzione proposta.
2. Dato a e la posizione i di uno degli elementi scambiati, **descrivere** un algoritmo di complessità in tempo $O(\log n)$ per individuare la posizione j dell'altro elemento scambiato, e **dimostrare** che l'algoritmo è ottimo.

Esercizio 4. (*6 punti*) Sia dato l'algoritmo

```
MergeSort3(a, sx, dx)
  if (sx + 1 == dx) {
    if (a[sx] > a[dx]) Scambia(a,sx,dx);
  }
  else if (sx + 1 < dx) {
    cx1 = sx + (dx - sx)/3;
    cx2 = sx + 2*(dx - sx)/3;
    MergeSort3(a, sx, cx1);
    MergeSort3(a, cx1+1, cx2);
    MergeSort3(a, cx2+1, dx);
    Fusione(a, sx, cx1, cx2, dx);
  }
```

Scrivere e risolvere la relazione di ricorrenza $T(n)$ che esprime la complessità in tempo dell'algoritmo, tenendo conto che la procedura **Fusione** è una immediata generalizzazione della procedura usata nel MergeSort classico.