Introduzione agli algoritmi
Prova di esame del 1/2/2016
E. Fachini - I. Finocchi

1. Si descriva e si analizzi il mergesort.

2. Considerare la seguente funzione:

```
void f(int T [], int inizio, int fine) {
    int n = fine – inizio + 1
    int S[n]
    if (n>0){
         copia(S, T, inizio, fine)
         InsSort(S,n)
         f(T, inizio, inizio+n/2)
         f(T, inizio+n/2+1, fine)
```

Qui la funzione copia(S,T,inizio,fine) copia gli elementi fra inizio e fine dell'array T in S. Si imposti la relazione di ricorrenza che ne descriva la complessità e la si risolva utilizzando il metodo della sostituzione

3.Siano U e V due array contenenti lo stesso numero n di elementi. Progettare un algoritmo che determina se U e V individuano (a meno di ripetizioni) lo stesso insieme di elementi.

Esempio.
Gli array V=[1,2,1,1,2] ed U=[2,2,1,1,2] determinano l'insieme {1,2}, e quindi con questo input l'output dell'algoritmo deve essere true.
Gli array V=[1,1,2,2,3] ed U=[2,3,2,3,2] determinano rispettivamente gli insiemi {1,2,3} e {2,3}, e quindi con questo input l'output dell'algoritmo deve essere false.

L'algoritmo deve avere tempo di esecuzione O(n lg n) nel caso peggiore. Scrivere l'algoritmo ed analizzarne il tempo di esecuzione in modo dettagliato.

Introduzione agli algoritmi
Prova di esame del 1/2/2016
E. Fachini - I. Finocchi

parte II

3. Si dimostri che un max-heap (o un min-heap) può essere costruito a partire da un array qualunque in tempo O(n), dove n è il numero degli elementi dell’array.

4. Sia dato un min-heap H di n interi. Si determini dove può trovarsi il quarto intero più piccolo. Dedurne il tempo necessario per individuarlo.

5. Dato un albero binario T, si descriva un algoritmo Fib(T,k) che restituisce vero se T è un albero di Fibonacci di altezza k. Si analizzi la correttezza e il tempo di esecuzione dell'algoritmo proposto.