Introduzione agli Algoritmi
A.A. 2011/2012
Prova Intermedia del 26 Aprile 2012
II canale - testo 2
Prof.ssa Irene Finocchi

Problema 1

Dimostrare la seguente proprietà:

> *Sia T_n l'albero delle chiamate ricorsive dell'algoritmo con tempo di esecuzione esponenziale per il calcolo dell'n-esimo numero di Fibonacci. Dimostrare che il numero di foglie di T_n è esattamente uguale all'n-esimo numero di Fibonacci F_n*

Problema 2

Un algoritmo di ordinamento è detto "parsimonioso" se nessuna coppia di elementi del vettore che ordina sono confrontati più di una volta. Si esaminino gli algoritmi di ordinamento studiati e per ciascuno di essi si dica se si tratta di un ordinamento parsimonioso, giustificando la risposta:

- insertion sort
- bubblesort
- mergesort
- quicksort

Problema 3

3.1. Si consideri il seguente frammento di codice:

```
count = n
n = A.length
ArraySort(A)
 for  i = 1 to  n  do
      if (binarySearch(A, A[1]+A[i])) then count = count-1
```

Si analizzi asintoticamente il tempo di esecuzione, nel caso peggiore, del frammento di codice sia nell'ipotesi che ArraySort sia il mergeSort che in quella in cui sia il bubbleSort.

3.2 Si consideri il seguente frammento di codice:

```
funzione(n)
if (n == 0) then return 0
return 3*funzione(n/2) + fun(n)
```

Si analizzi asintoticamente il tempo di esecuzione del frammento di codice nell'ipotesi che la funzione `fun` abbia complessità di tempo in $\Theta(n^2)$.

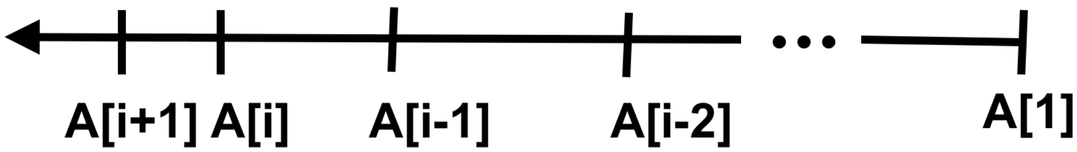
Introduzione agli Algoritmi
A.A. 2011/2012
Prova Intermedia del 26 Aprile 2012
II canale - testo 2
Prof.ssa Irene Finocchi

Problema 4

4.1. Dati un intero k e un array ordinato descrescente A contenente n valori interi tale che, per ogni i≥2, risulti:

$$A[i] - A[i+1] \le A[i-1] - A[i]$$

progettare un algoritmo con tempo di esecuzione O(log n) che trovi, se esiste, una coppia di elementi consecutivi la cui differenza è = k.
Se rappresentiamo i valori dell'array su una retta (la freccia indica che i valori **diminuiscono** verso sinistra) si mette in evidenza la proprietà dell'array e cioè che anche la distanza tra valori successivi decresce:



4.2. Si supponga ora che l'array A, oltre ad essere ordinato decrescente, contenga interi *positivi* e soddisfi la seguente proprietà:

per ogni i ≥ 2 2A[i] ≤ A[i-1]

Progettare un algoritmo con tempo di esecuzione O(log n) che trovi, se esiste, una coppia di elementi consecutivi la cui differenza è =k

[Suggerimento: ricondursi al caso 4.1]