



UNIVERSITA DEGLI STUDI DI TORINO

Laboratorio di Algoritmi e Strutture Dati

Relazione e progetto di Francesco Mazzucco

aa2021/2022

Ex1

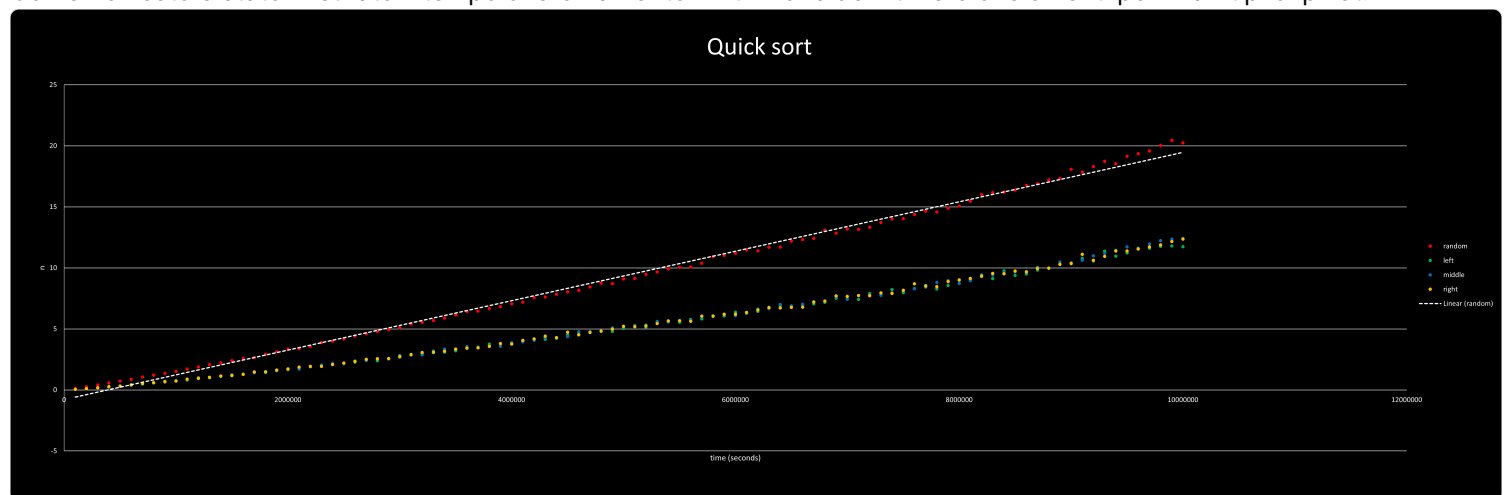
Nel primo esercizio è stato richiesto di implementare una libreria di sort contenente due algoritmi, il quicksort e il binary insertion sort.

Cme requisito aggiuntivo è anche stato chiesto di scrivere un programma che utilizzi la libreria di sort implementata per ordinare, secondo vari metodi di ordinamento, un file CSV dato.

Di seguito nel dettaglio i test effettuati per valutare la performance dei due algoritmi

Quicksort

Come richiesto è stato misurato il tempo di ordinamento in funzione del numero di elementi per i vari tipi di pivot:

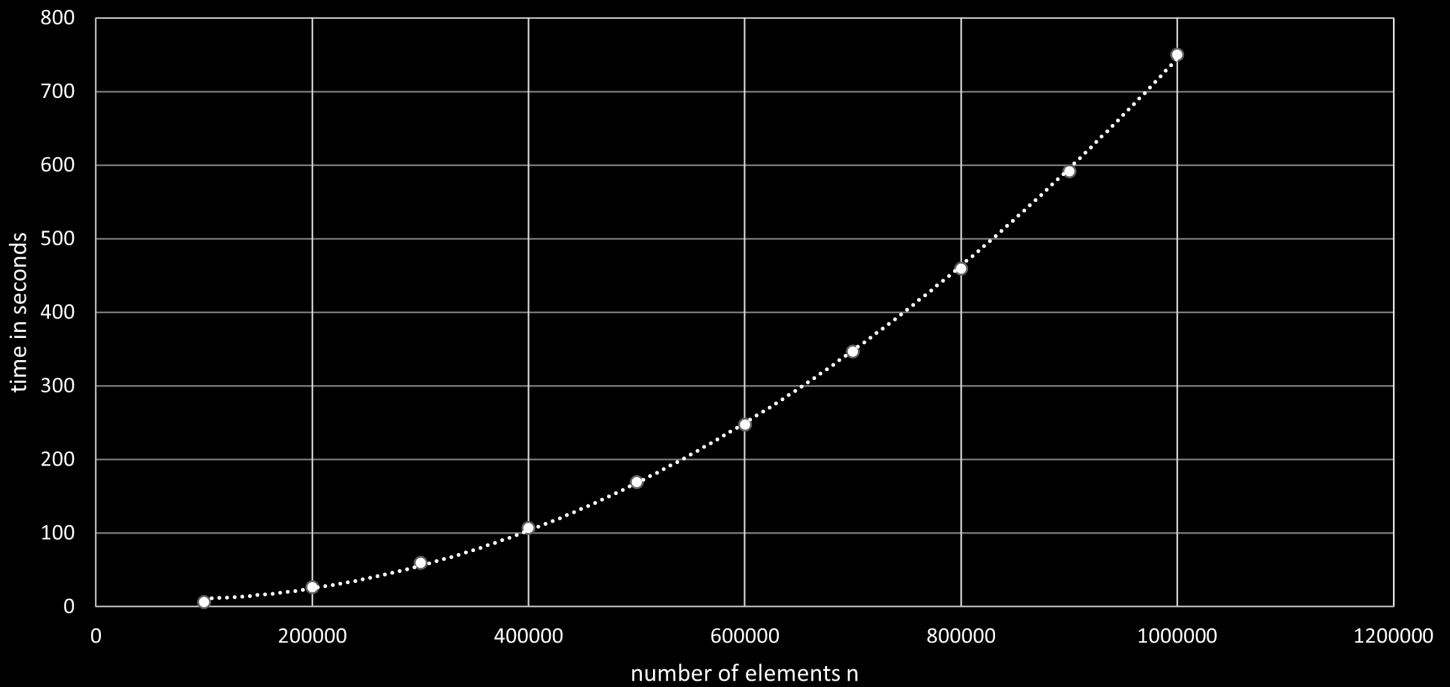


Come si può osservare il pivot random è quello più performante, mentre gli altri 3 non cambiano molto tra di loro

Binary insertion sort

Come è già noto l'insertion sort, indipendentemente dall'algoritmo di ricerca che viene usato ha una funzione di crescita media di n^2 .

Binary insertion sort



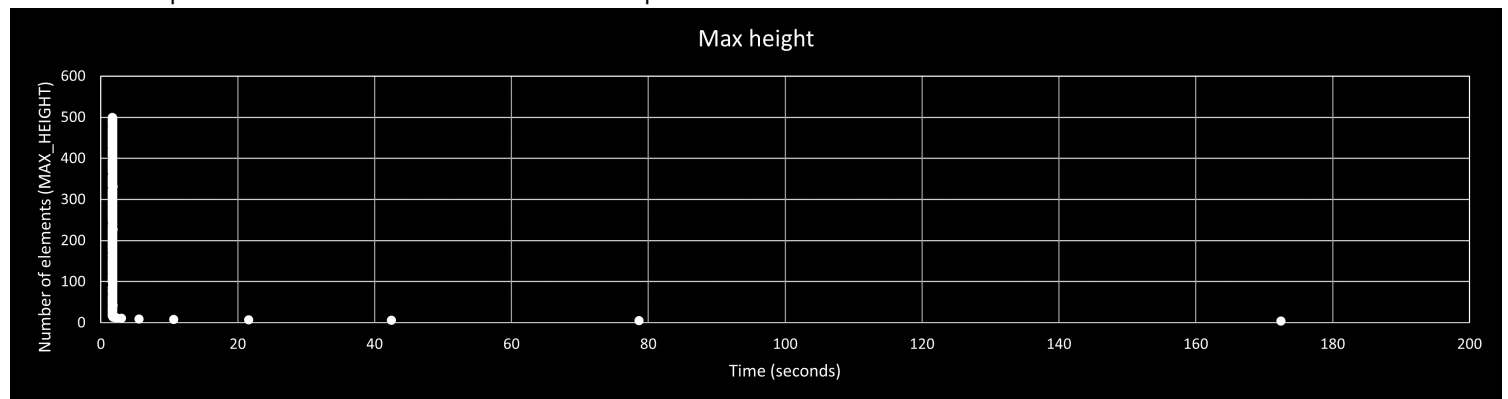
Inoltre non è stato possibile ordinare tutti i 20 milioni di record con questo algoritmo.

Ex2

Nel secondo esercizio è stato richiesto di implementare la struttura dati skip list e di usarla per scrivere un'applicazione che determini le parole sbagliate dentro a un testo.

Per ogni parola del testo si è fatta quindi una ricerca dentro una skip list contenente un dizionario.

E' stato inoltre chiesto di sperimentare l'implementazione della skip list con vari valori di **MAX_HEIGHT** e di osservare la variazione di performance in funzione del variare di questa variabile.



Come possiamo osservare con un numero basso di puntatori ausiliari le performance sono piuttosto scarse ma si raggiunge velocemente un punto, intorno ai 20 elementi, oltre il quale non vi è nessun beneficio ad aumentare il valore di **MAX_HEIGHT**.

MAX_HEIGHT	TIME(s)
5	172.3945
6	78.585918
7	42.411512
8	21.537281
9	10.605983
10	5.553662
11	2.951742
12	2.294606
13	1.934591

