Relazione: studio della complessità degli algoritmi di ordinamento

Alberto Pochettino Federico Pitaccolo

* Introduzione

Le colonne {INT, STRING, FLOAT} delle tabelle corrispondono all’asse verticale dei grafici e sono misurate in millisecondi.

Nei grafici i colori corrispondono a un campo di ordinamento: INT, STRING, FLOAT.

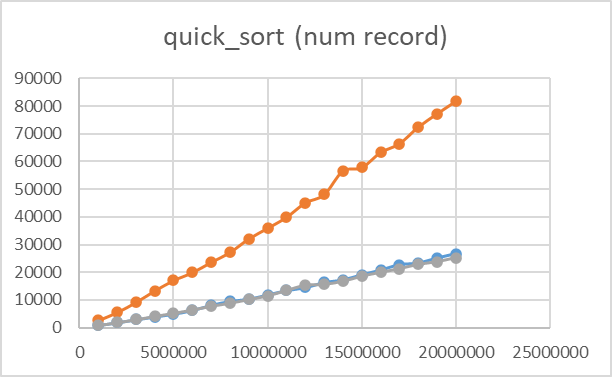
In generale abbiamo constatato che l’ordinamento col quick sort è più efficiente e rapido rispetto all’ordinamento con il binary insertion sort. Specialmente per quanto riguarda il campo int e float.

* Scelta del pivot

Nel caso in cui l’elemento pivot del partition è stato impostato a low l’efficienza del quick sort cala drasticamente, rendendo pertanto tale scelta la peggiore. Tra partition high e partition rand la differenza è minima; tuttavia, a livello probabilistico partition rand è più efficiente, poiché ci sono più chance di trovare già l’elemento medio.

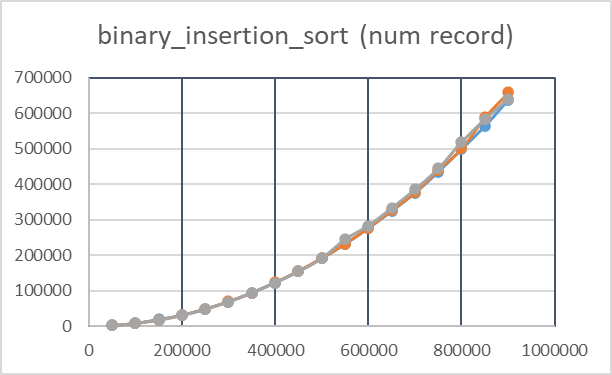
* Tempi di risposta del Quick Sort al variare del numero di elementi da ordinare

con patition\_high



La complessità del quick sort è O( n log n ).

* Tempi di risposta del Binary Insertion Sort al variare del numero di elementi da ordinare



La complessità del binary insertion sort è O( n log n ).