

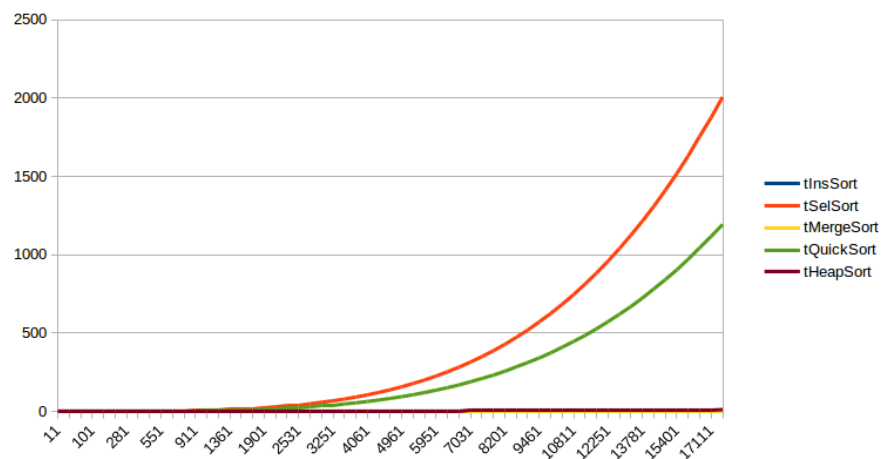
Confronto algoritmi di ordinamento

Usando il linguaggio C, sono stati implementati i seguenti algoritmi di ordinamento:

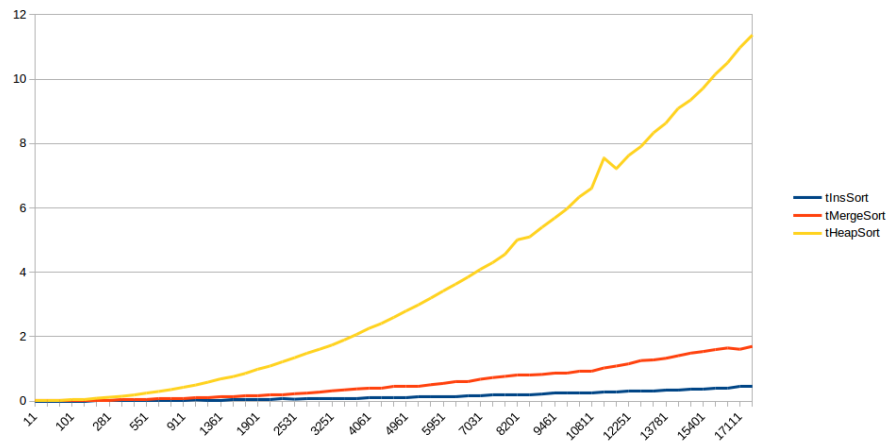
- Selection Sort
- Insertion Sort
- Merge Sort
- Quick Sort (versione Hoare)
- Heap Sort

Tramite un altro programma C, si sono confrontate le performance (dal punto di vista temporale) dei suddetti algoritmi su input di diverse dimensioni; i risultati (calcolati in millisecondi) sono stati salvati in un file *cvs* dal quale si sono poi tratti dei grafici esplicativi.

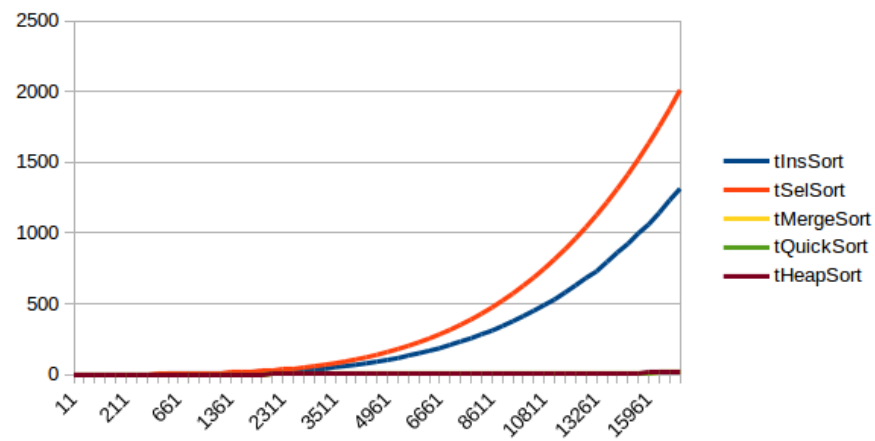
I vettori da ordinare forniti in input sono stati riempiti in maniera sia ordinata sia pseudo-casuale (in quest'ultimo caso utilizzando la libreria standard C).



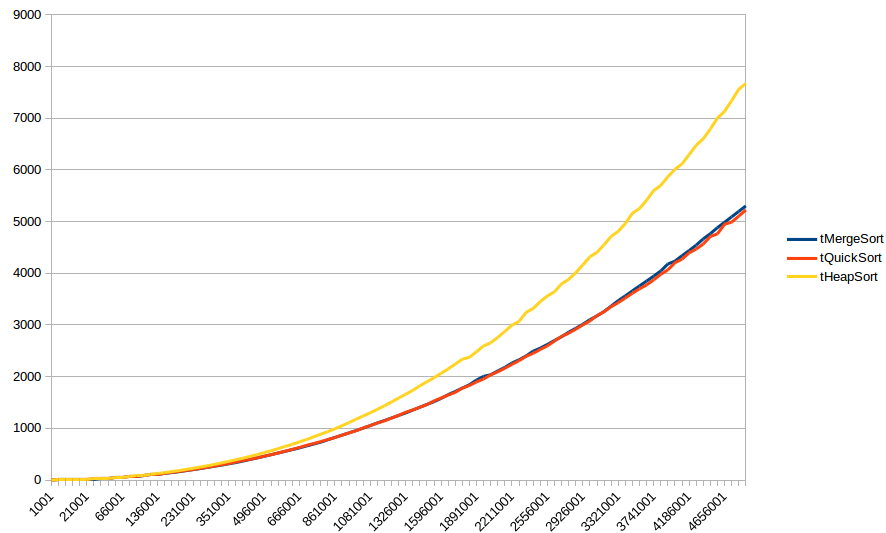
Per prima cosa si sono confrontati tutti gli algoritmi dando loro in pasto degli array di crescente lunghezza già ordinati. Come ci si aspettava l'*heapsort* non ha un caso molto peggiore rispetto al medio in tali condizioni, come invece accade al *quicksort*.



Confrontando però solo i tre algoritmi dalle migliori prestazioni nel caso di array ordinati, ovvero l'*InsertionSort*, il *MergeSort ottimizzato* (cioè che non compie l'operazione di merge ove non necessario) e l'*HeapSort* è immediato notare come i primi due si comportino "molto meglio" del secondo: essi sono infatti lineari in questo miglior caso, mentre l'*HeapSort* rimane pseudo-lineare.



Usando degli array non ordinati si è confermata l'appartenenza della complessità temporale dell'*HeapSort* alla famiglia di quelle di *Mergesort* e *Quicksort*



Confrontando infine questi tre algoritmi piu' performanti l'*HeapSort* risulta distaccarsi notevolmente (in peggio) sia dal *QuickSort* che dal *MergeSort*, che si rivelano invece tra loro molto simili, con un leggero vantaggio del primo rispetto al secondo.

Mentre e' nota la migliore prestazione empirica del *QuickSort* rispetto agli altri algoritmi di ordinamento, il vantaggio del *MergeSort* rispetto all'*HeapSort* era meno previsto.

Tentando di dare una spiegazione, si potrebbe ipotizzare che il *MergeSort* abbia una maggiore frequenza di hit nella cache rispetto a quella dell'*HeapSort*.