

# Relazione Esercizio 1

giovedì 29 aprile 2021 22:28

## Tabella dei risultati

Aa K	# String field	# Int field	# Float field	Σ Tot
0	19.484	18.969	19.391	57.844
<u>10</u>	16.282	14.812	15	46.094
<u>20</u>	16.937	14.874	15.406	47.217
<u>30</u>	16.359	15.484	15.312	47.155
<u>40</u>	16.578	15.078	15.516	47.172
<u>50</u>	16.687	15.313	15.454	47.454
<u>100</u>	17.109	15.407	15.718	48.234
500	19.063	17.391	17.89	54.344
<u>1000</u>	21.343	19.953	20.391	61.687
<u>5000</u>	52.406	51.719	51.952	156.077
<u>10000</u>	87.844	87.923	88.156	263.923

Dai risultati sopra elencati si può notare di come l'utilizzo del binary-insertion-sort vada a migliorare/peggiore le tempistiche del puro merge ( $k = 0$ ).

Se impieghiamo il binary-insertion-sort per piccoli  $k$ , e dunque per piccoli sottovettori, allora le prestazioni dell'algoritmo ibrido migliorano.

D'altro canto se lo utilizziamo con sottovettori grandi le prestazioni calano vertiginosamente.

Puntualizzo che i dati raccolti potrebbero essere imprecisi a causa dell'impreciso carico del processore ad ogni misurazione, difatti rifacendo i test per lo stesso  $k$  mi sono ritrovato con valori leggermente diversi sebbene in linea con quelli mostrati in tabella.

Il  $k$  ottimale si trova tra 10 e 50, non riuscirei a definirlo con precisione per l'alta variabilità dei tempi.

Per trovare il  $k$  ottimale ho eseguito dei test su  $k$  molto generici (10, 100, 1000 e 10000), andando poi ad approfondire nel dettaglio dove le prestazioni sembravano migliori (20, 30, 40, 50).

Successivamente ho aggiunto dei test per i casi in cui le prestazioni sembravano peggiorare (500, 5000) che, come pensavo, hanno dato anch'essi prestazioni cattive.

Sapendo che con l'aumentare di  $k$  e quindi con il passaggio di sottovettori sempre più grandi al binary-insertion-sort le prestazioni sarebbero ancor di più peggiorate mi è sembrato inutile fornire altre misurazioni oltre  $k = 10000$ .

I risultati ottenuti rispecchiano quelle che erano le mie aspettative. Come immaginavo l'utilizzo dell'algoritmo ibrido avrebbe portato benefici per piccoli sottovettori mentre avrebbe abbassato notevolmente le prestazioni per sottovettori grandi.