

Elaborato esercizio 1 - Laboratorio Algoritmi e Strutture dati

Taralli Giulio - Toure Ismaila

Introduzione

L'obiettivo dell'esercizio è la ricerca di un numero k ideale per passare dall'ordinamento MergeSort all'ordinamento Binary InsertionSort.

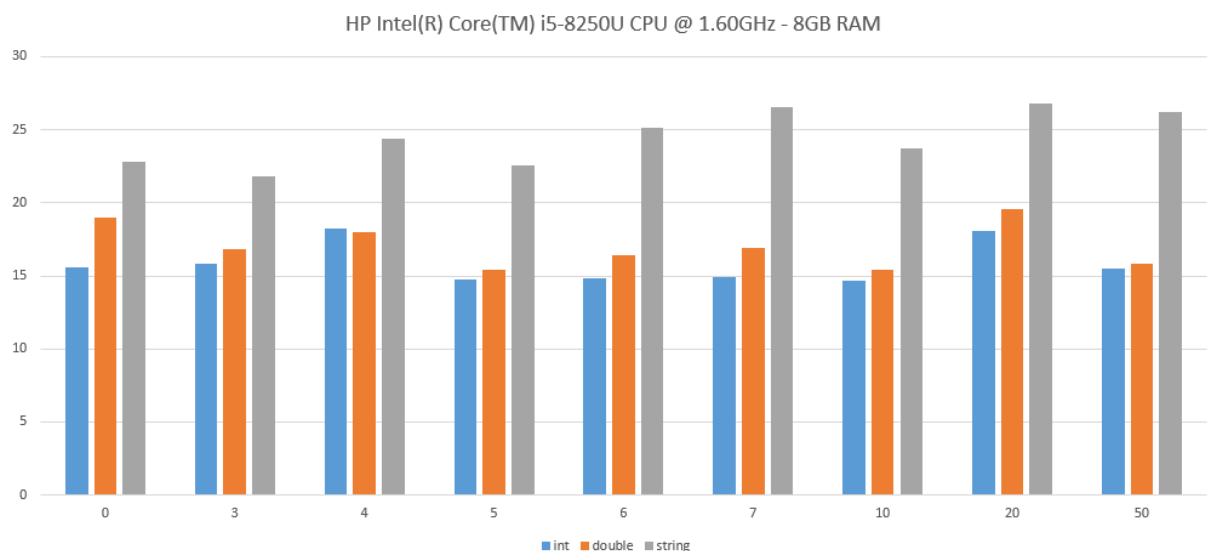
Abbiamo eseguito le seguenti simulazioni con su due diverse macchine linux con differenti processori:

1. HP Intel Core i5 con 8GB di RAM
2. HP AMD A4 con 4GB di RAM

Visualizzazione dei dati

- HP Intel Core i5 con 8GB di RAM

Il seguente grafico riporta le prestazioni dei vari ordinamenti in secondi:



In blu sono rappresentati i tempi di ordinamento degli interi

In arancione sono rappresentati i tempi di ordinamento degli double

In grigio sono rappresentati i tempi di ordinamento delle stringhe

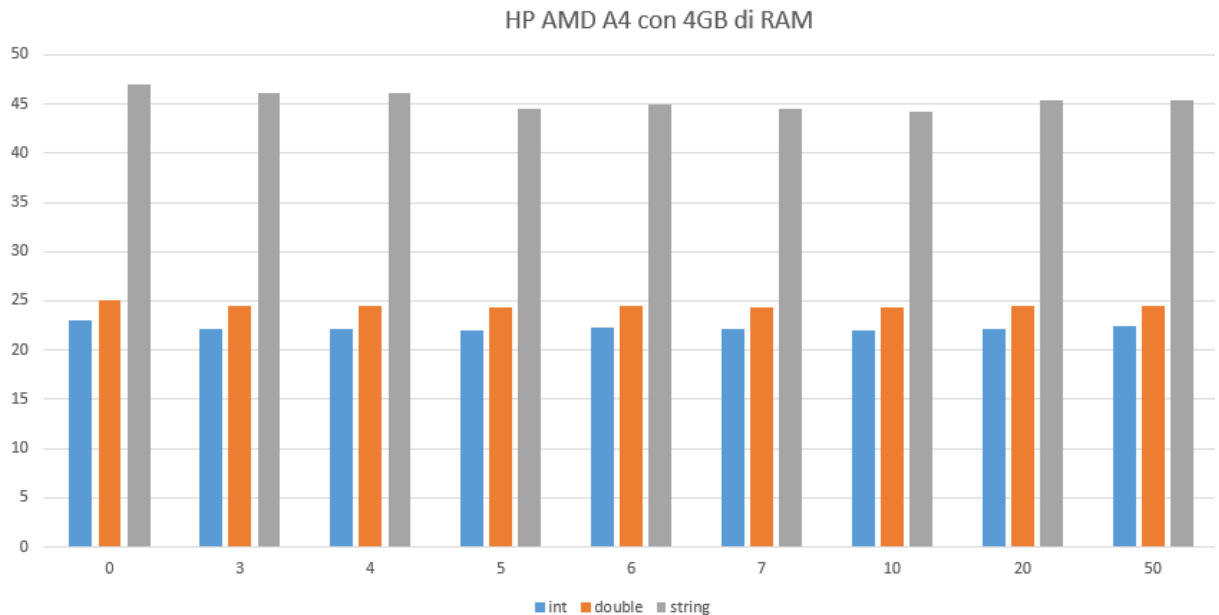
Su questa macchina notiamo che il valore k ottimale risulta essere 5

In seguito riportiamo i valori precisi dei tempi di simulazione:

| k | 0 | 3 | 4 | 5 | 6 | 7 | 10 | 20 | 50 |
|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| int | 15,58 | 15,851 | 18,226 | 14,763 | 14,828 | 14,882 | 14,661 | 18,035 | 15,518 |
| double | 19,025 | 16,864 | 17,998 | 15,389 | 16,454 | 16,899 | 15,411 | 19,603 | 15,799 |
| string | 22,802 | 21,778 | 24,425 | 22,538 | 25,155 | 26,553 | 23,7 | 26,778 | 26,189 |

- HP AMD A4 con 4GB di RAM

Il seguente grafico riporta le prestazioni dei vari ordinamenti in secondi:



In blu sono rappresentati i tempi di ordinamento degli interi

In arancione sono rappresentati i tempi di ordinamento degli double

In grigio sono rappresentati i tempi di ordinamento delle stringhe

Anche su questa macchina notiamo che il valore k ottimale risulta essere 5

In seguito riportiamo i valori precisi dei tempi di simulazione:

| k | 0 | 3 | 4 | 5 | 6 | 7 | 10 | 20 | 50 |
|--------|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| int | 22,97 | 22,205 | 22,212 | 22,023 | 22,357 | 22,134 | 22,073 | 22,135 | 22,431 |
| double | 25,083 | 24,48 | 24,476 | 24,376 | 24,501 | 24,352 | 24,362 | 24,467 | 24,472 |
| string | 46,964 | 46,141 | 46,191 | 44,537 | 44,955 | 44,56 | 44,291 | 45,403 | 45,372 |

Commento finale

Notiamo che ovviamente i valori cambiano a seconda dell'architettura, dal tipo di processore e dalla quantità di RAM disponibile dalla macchina. Nei nostri due casi il processore Intel i5 è molto più veloce rispetto al processore AMD A4.

Prima di iniziare le varie simulazioni abbiamo ritenuto che il valore di k dovrà essere per forza un valore pressoché basso. Questo lo si può dedurre dal fatto che l'algoritmo di BinaryInsertionSort ha una complessità quadratica $O(n^2)$, per cui un valore di k troppo alto allungherebbe il tempo di esecuzione.

Abbiamo notato inoltre che durante le simulazioni che il tempo di esecuzione è influenzato (oltre al k) dallo stato della memoria RAM del PC.

Esempio tempi di esecuzione sulla macchina HP Intel Core i5 con 8GB di RAM:

- Simulazione eseguita appena acceso il pc:

| k | 10 | 20 |
|--------|--------|--------|
| int | 14,661 | 18,035 |
| double | 15,411 | 19,603 |
| string | 23,7 | 26,778 |

- Simulazione eseguita dopo una giornata di uso intensivo:

| k | 10 | 20 |
|--------|--------|--------|
| int | 17,955 | 18,841 |
| double | 18,921 | 19,105 |
| string | 26,627 | 29,669 |