**RELAZIONE ESERCIZIO 1**

Il Merge-BinaryInsertion Sort ha come obbiettivo quello di combinare il Merge sort e il BinaryInsertion sort, con lo scopo di verificare quando sia più conveniente usare un algoritmo piuttosto che l'altro.

Per far ciò sono stati svolti vari test facendo assumere a *k* valori sempre più grandi e misurando per ciascuno i tempi di esecuzione dell’algoritmo.

Nei risultati ottenuti sperimentalmente non emerge un valore netto di *k* per il quale il tempo di esecuzione risulti sempre ottimale, piuttosto si delinea un ventaglio di valori.

Ripetendo più volte i test con stessi valori di *k*, si possono infatti verificare differenze non solo assolute ma anche relative tra gli stessi: a parità di input e macchina, il valore, ad esempio, *k=6* può risultare più efficiente del suo successore e, in una esecuzione successiva, verificarsi l’opposto.

I valori compresi tra 2 a 15-20 (circa) contengono sempre le prestazioni migliori, mentre i valori superiori a 100 hanno sempre prestazioni peggiori; i valori compresi tra questi due estremi tendono, al loro crescere, ad allinearsi ai tempi del MergeSort.

I risultati sperimentali si possono dire in linea con le previsioni di partenza: per *n* relativamente piccoli, un algoritmo con crescita asintotica superiore ad un altro può effettivamente risultare più efficiente. Inoltre, superata la soglia critica del 100, le prestazioni effettivamente degradano rispetto al MergeSort: con l’aumentare del peso dato al BinaryInsertionSort nell’algoritmo ibrido e per valori “non piccoli” le prestazioni peggiorano proporzionalmente al peso dato a quest’ultimo.