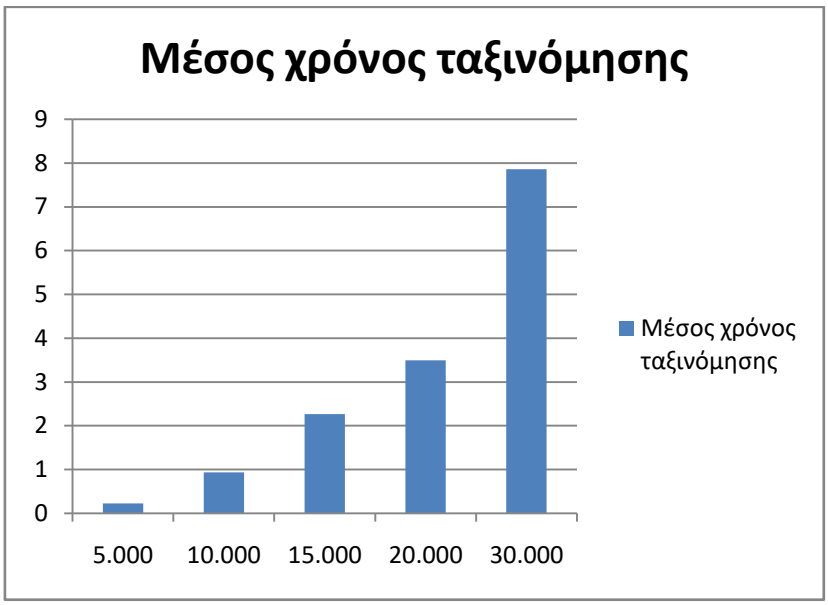
4η Εργαστηριακή Άσκηση Δομών Δεδομένων
Διονύσιος Θεοδόσης
A.M:321/2015066

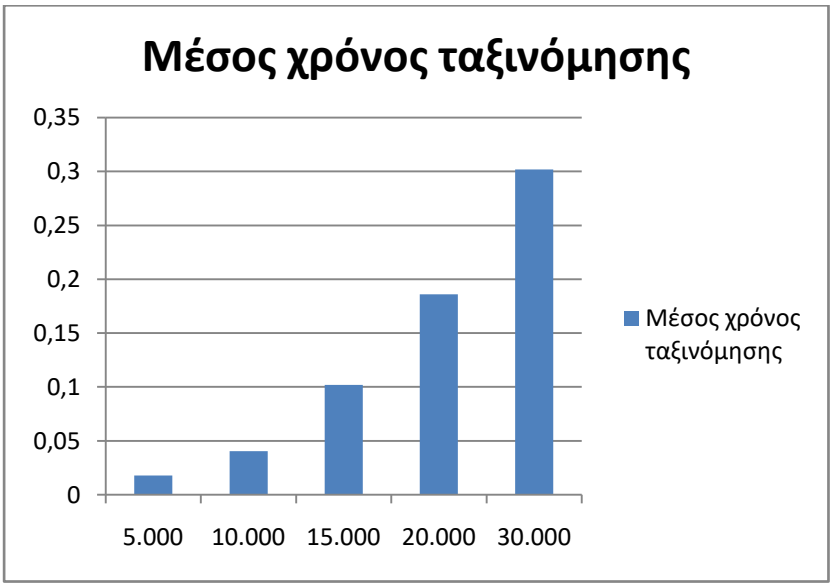
Αλγόριθμος Ταξινόμησης Παρεμβολής (Insertion Sort)

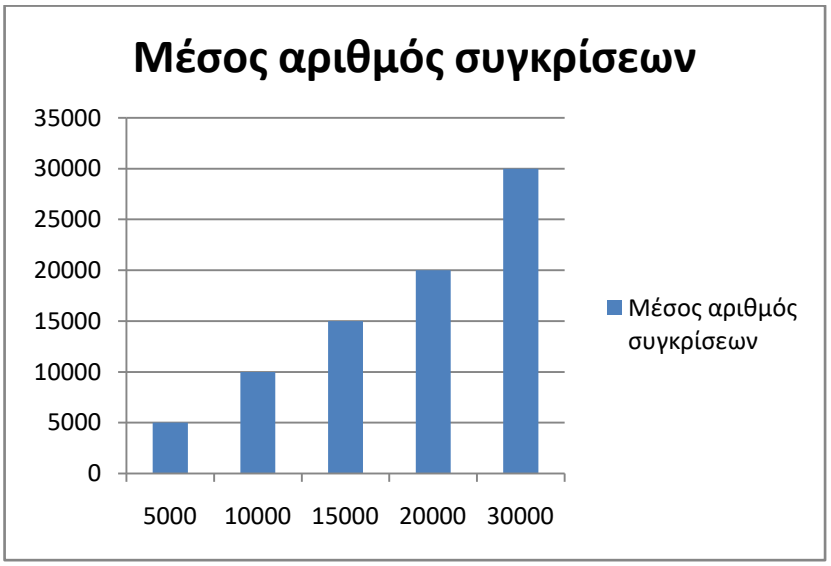
N	Μέσος αριθμός συγκρίσεων	Μέσος χρόνος ταξινόμησης
5.000	6.264.450	0.225967
10.000	25.057.738	0.937600
15.000	56.345.809	2.268100
20.000	100.014.602	3.493300
30.000	224.964.664	7.862233



Αλγόριθμος Ταξινόμησης Σύζευξης (Merge Sort)

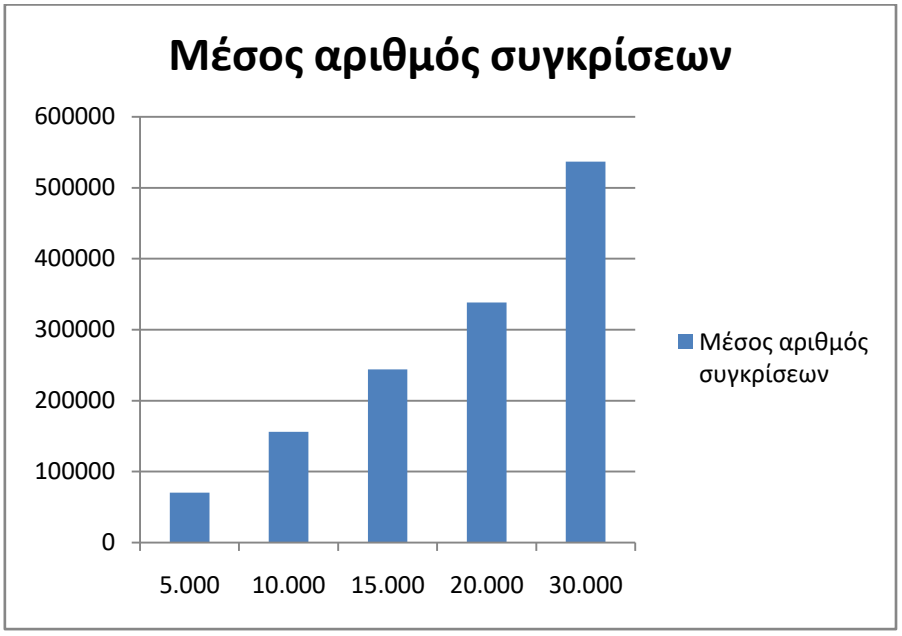
N	Μέσος αριθμός συγκρίσεων	Μέσος χρόνος ταξινόμησης
5.000	4999	0.017800
10.000	9999	0.040508
15.000	15000	0.101984
20.000	19999	0.185899
30.000	29998	0.301963

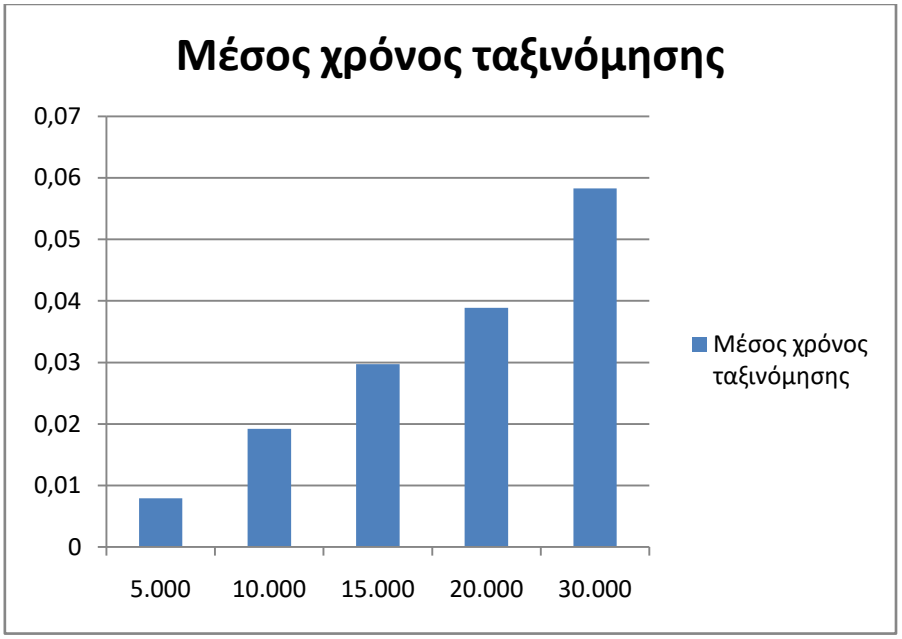




Αλγόριθμος Γρήγορης Ταξινόμησης (Quick Sort)

N	Μέσος αριθμός συγκρίσεων	Μέσος χρόνος ταξινόμησης
5.000	70599	0.007900
10.000	156210	0.019200
15.000	243877	0.029700
20.000	338216	0.038900
30.000	536712	0.058267



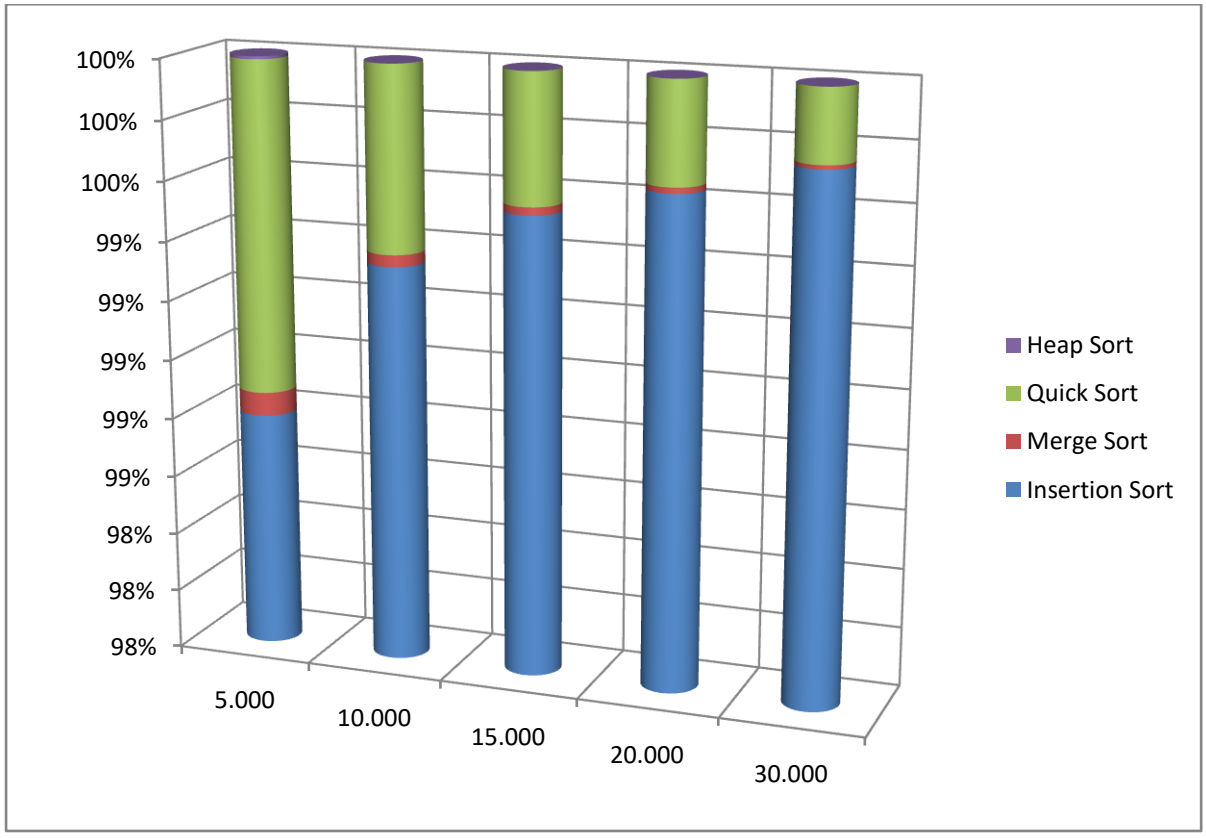


Αλγόριθμος Ταξινόμησης Σωρού (Heap Sort)

N	Μέσος αριθμός συγκρίσεων	Μέσος χρόνος ταξινόμησης
5.000	610	0.001100
10.000	206	0.002133
15.000	207	0.002900
20.000	236	0.003167
30.000	237	0.005400

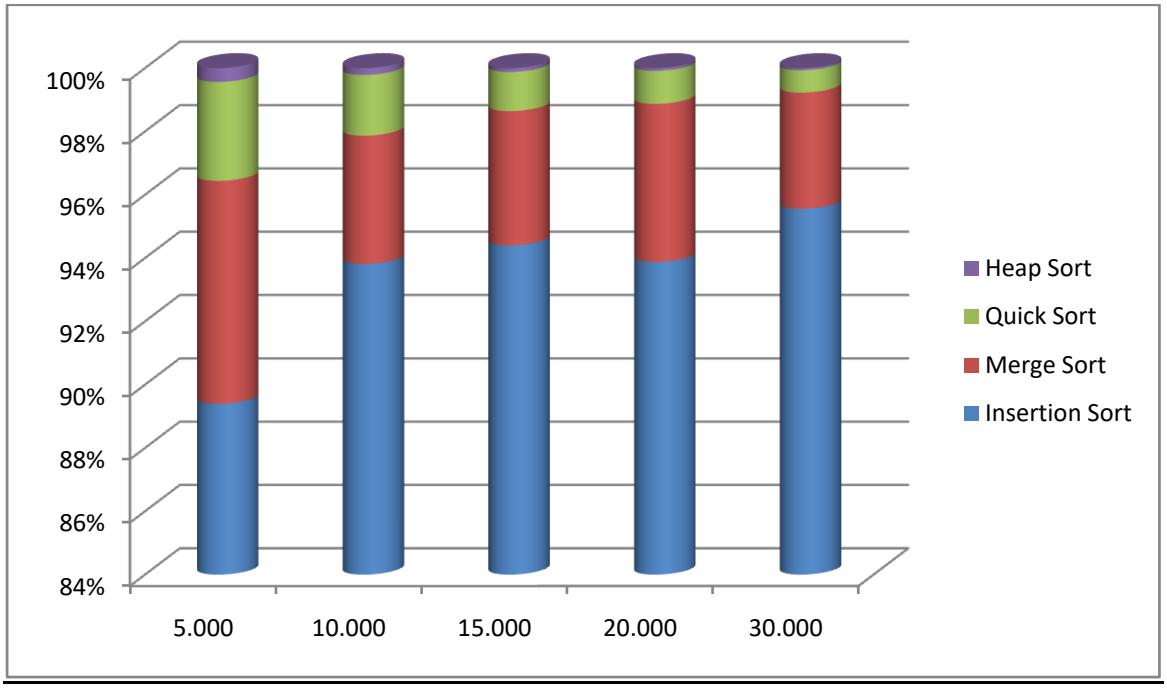
Συνολικά Διαγράμματα

1. Διάγραμμα απεικόνισης του μέσου αριθμού συγκρίσεων για κάθε αλγόριθμο ταξινόμησης.



Συμπέρασμα: Από το διάγραμμα βλέπουμε ότι η heap sort κάνει λιγότερες συγκρίσεις μετά η Merge sort ,μετά η quick sort και τέλος η insertion sort.Επίσης βλέπουμε ότι η insertion sort όσο αυξάνει ο όγκος κάνει περισσότερες συγκρίσεις ενώ οι άλλες δείχνουν ότι σε μεγαλύτερο όγκο είναι ποιο αποδοτικές σχετικά με την insertion sort.

2. Διάγραμμα απεικόνισης του μέσου χρόνου ταξινόμησης για κάθε αλγόριθμο ταξινόμησης.



Συμπέρασμα: Από το διάγραμμα βλέπουμε ότι η heap sort είναι πιο γρήγορη , μετά η quick sort ,μετά η Merge sort και τέλος η insertion sort.Επίσης ότι η merge sort ενώ με τον όγκο δείχνει να μειώνετε ο χρόνος ,όμως δεν έχει σταθερή μείωση. Ενώ στην quick short και στην heap sort βλέπουμε ότι όσο αυξάνετε ο όγκος τόσο ποιο αποδοτικές γίνονται .Τέλος η insertion sort βλέπουμε ότι με την αύξηση του όγκου γίνεται και ολοένα μη αποδοτική.