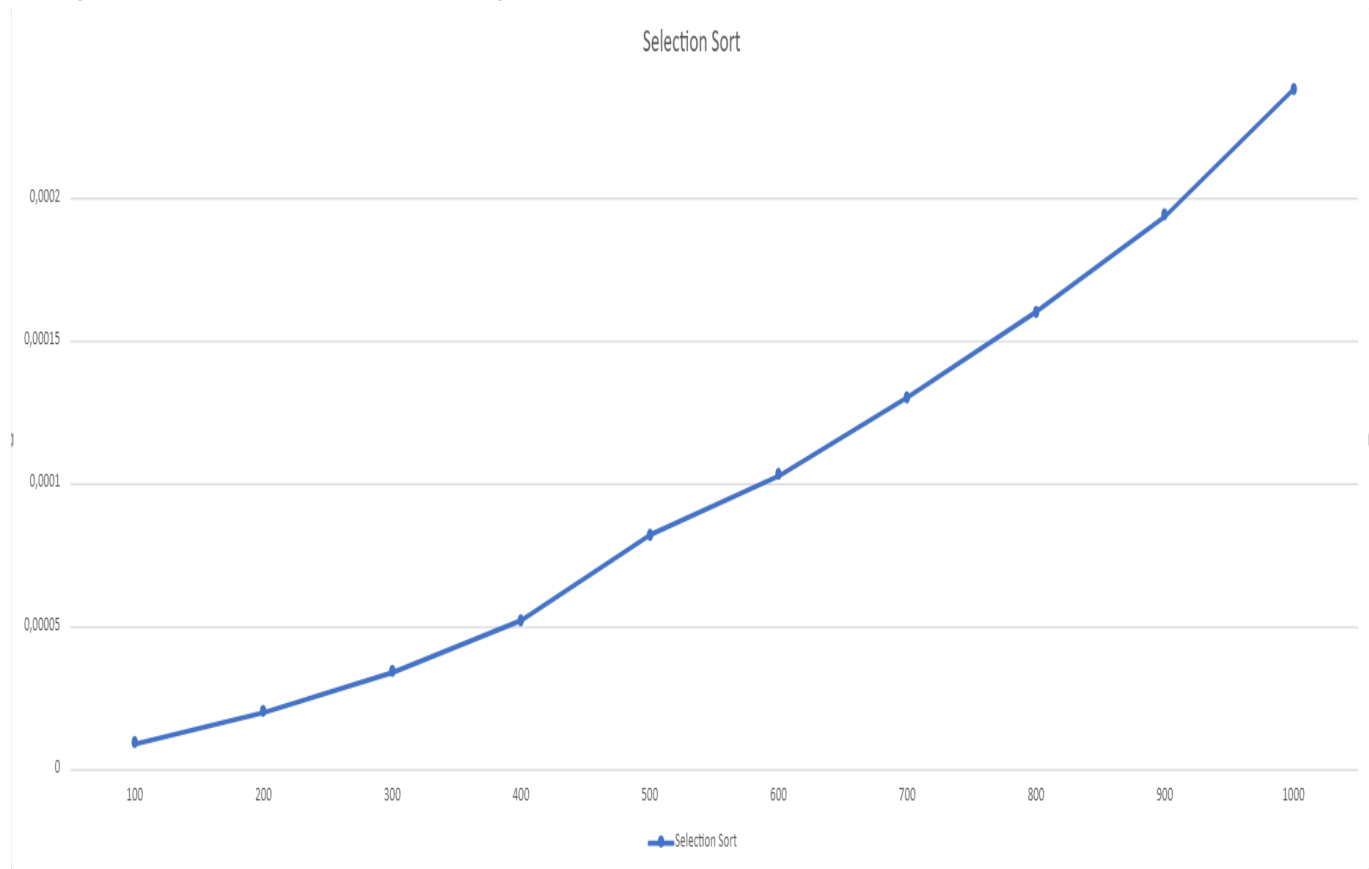


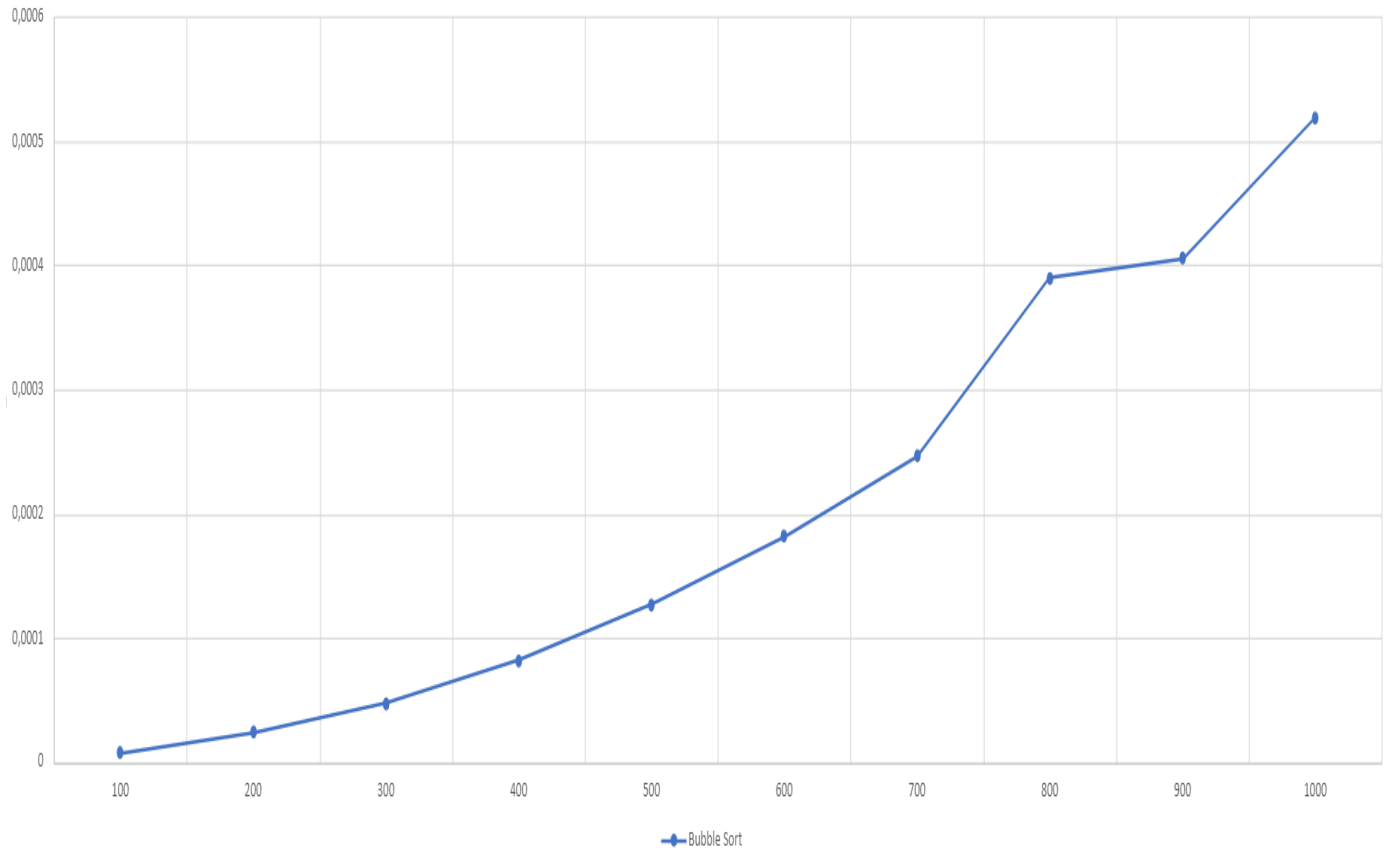
Έχοντας ολοκληρώσει το βήμα 3 παρατηρούμε ότι ,οι χρόνοι εκτελέσεων των αλγορίθμων ταξινόμησης είναι μηδαμινοί και καθώς αυξάνεται το πλήθος των στοιχείων , αυξάνεται και ο χρόνος ταξινόμησης τους.

Selection sort : Παρατηρούμε ότι έως τους 500 ακεραίους οι χρόνοι ταξινόμησης αυξάνονται με αργούς ρυθμούς ενώ στην συνέχεια η αύξηση αυτή γίνεται όλο και πιο έντονη.

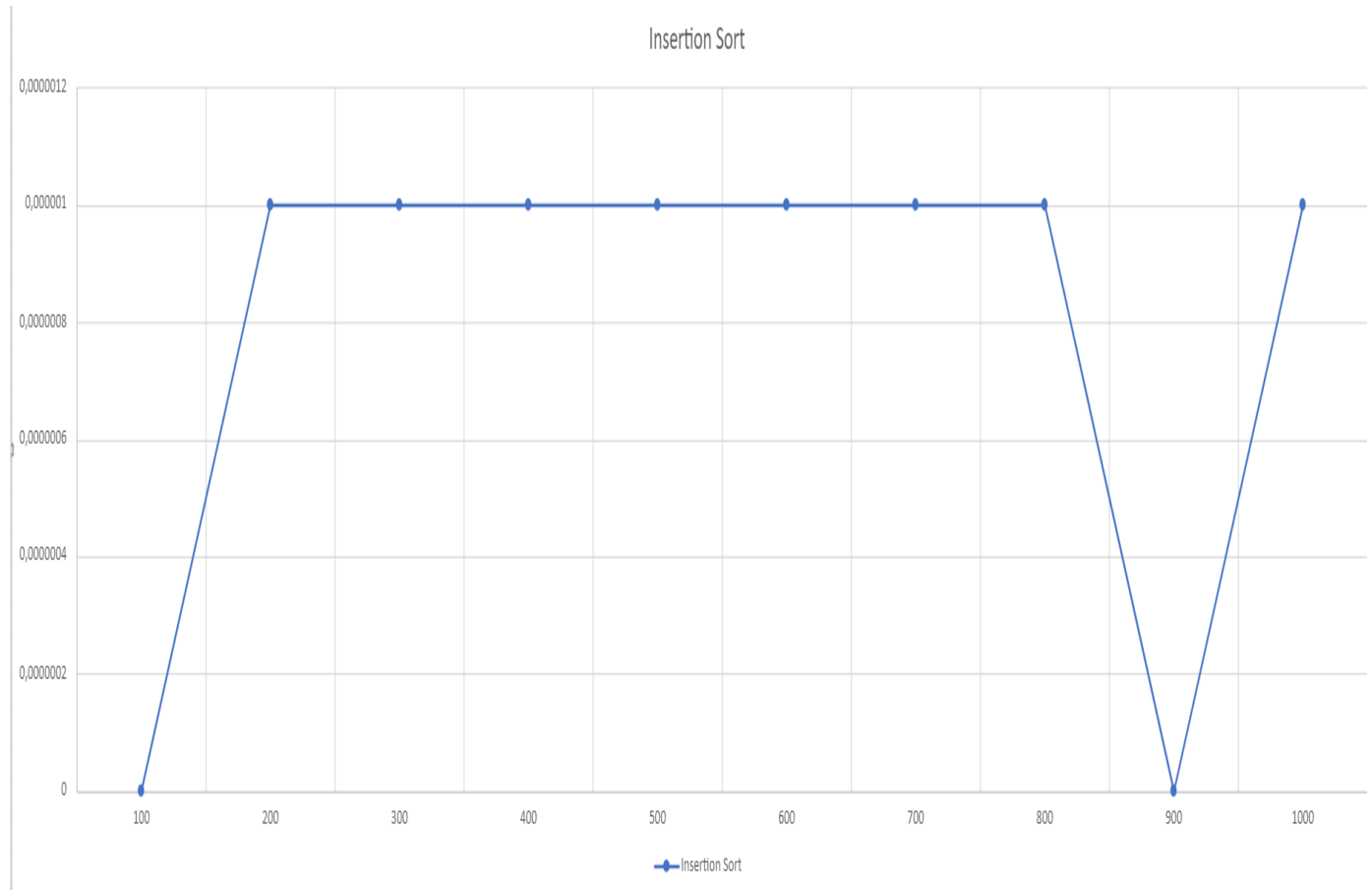


Bubble sort : Εδω φαίνεται ότι μέχρι τους 700 ακέραιους υπάρχει μια σταθερή αύξηση σε αντίθεση με τους χρόνους εκτέλεσης των 800 και 1000 ακέραιων όπου η άνοδος γίνεται με ταχύτατους ρυθμούς . Ενώ απο τους 800 έως 900 οι χρόνοι παραμένουν σχεδόν σταθεροί.

Bubble Sort

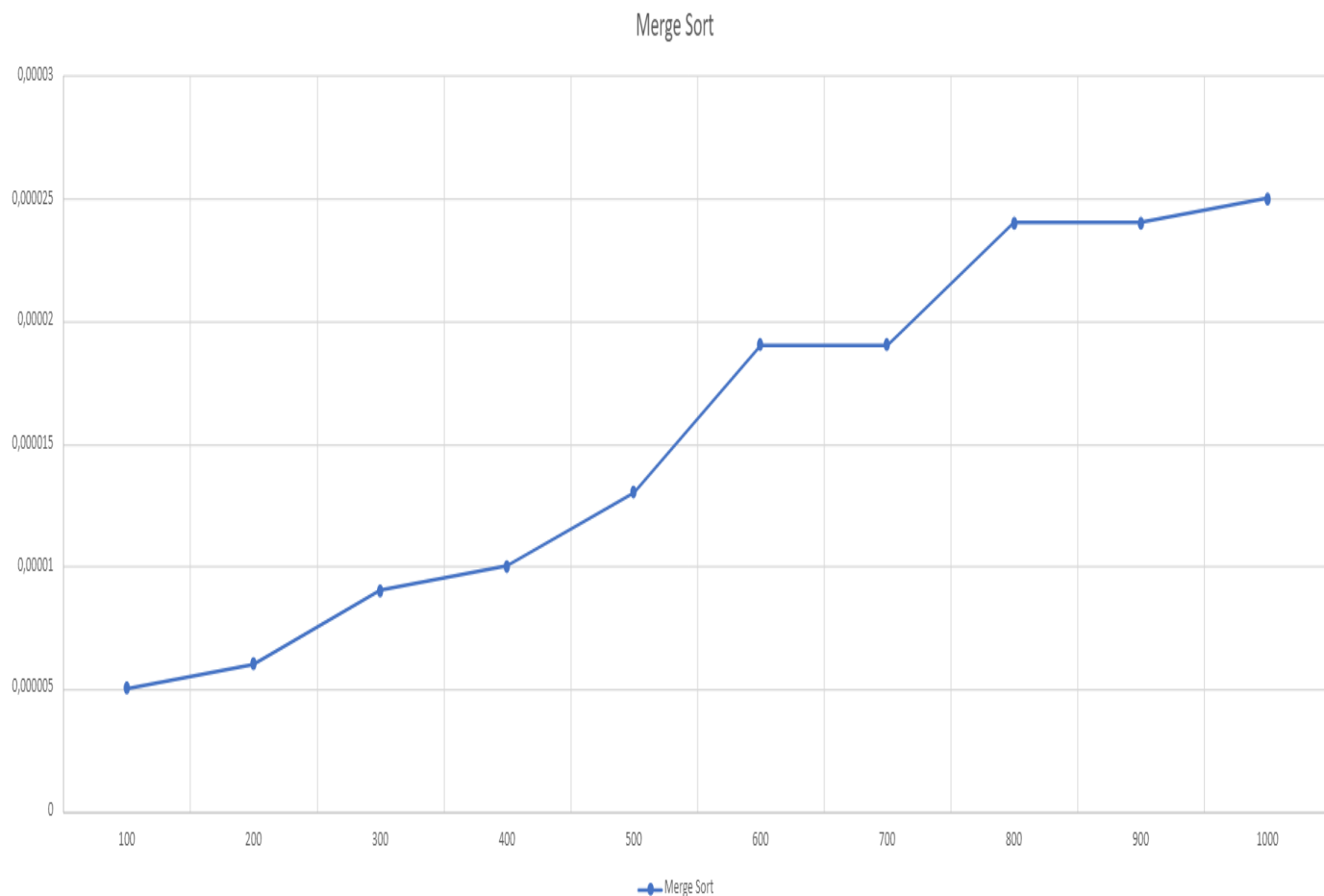


Insertion Sort : Στην Insertion Sort συμπερένουμε ότι οι χρόνοι ταξινόμησης παραμένουν σταθεροί σε όλο το πλήθος των ακεραίων που



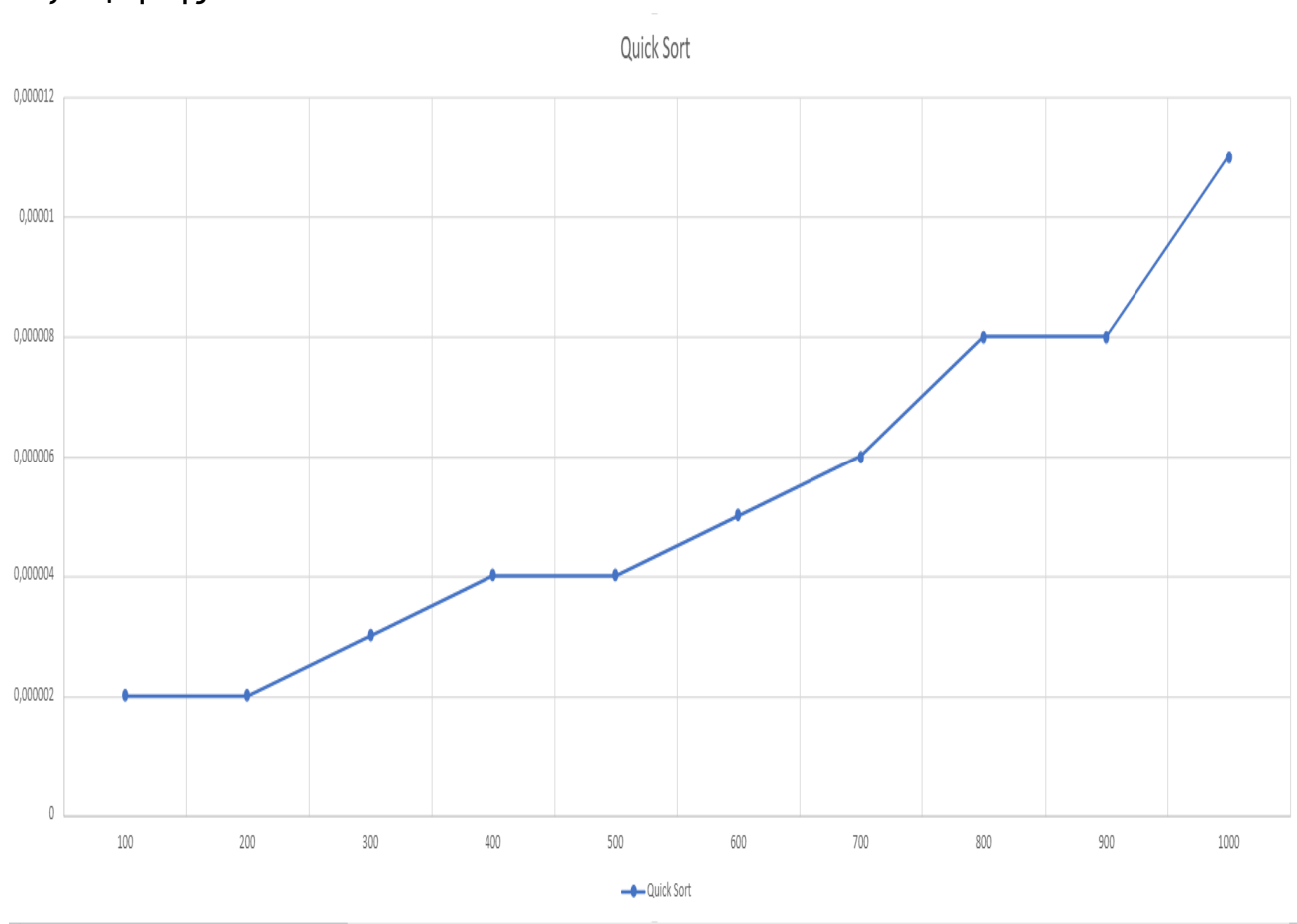
Merge Sort: Στο παρακάτω διάγραμμα παρατηρούμε ότι για μικρό πλήθος ακεραίων η Merge sort είναι γρήγορη ενώ σε μεγαλύτερους αριθμούς

η απόδοση της φθίνειται



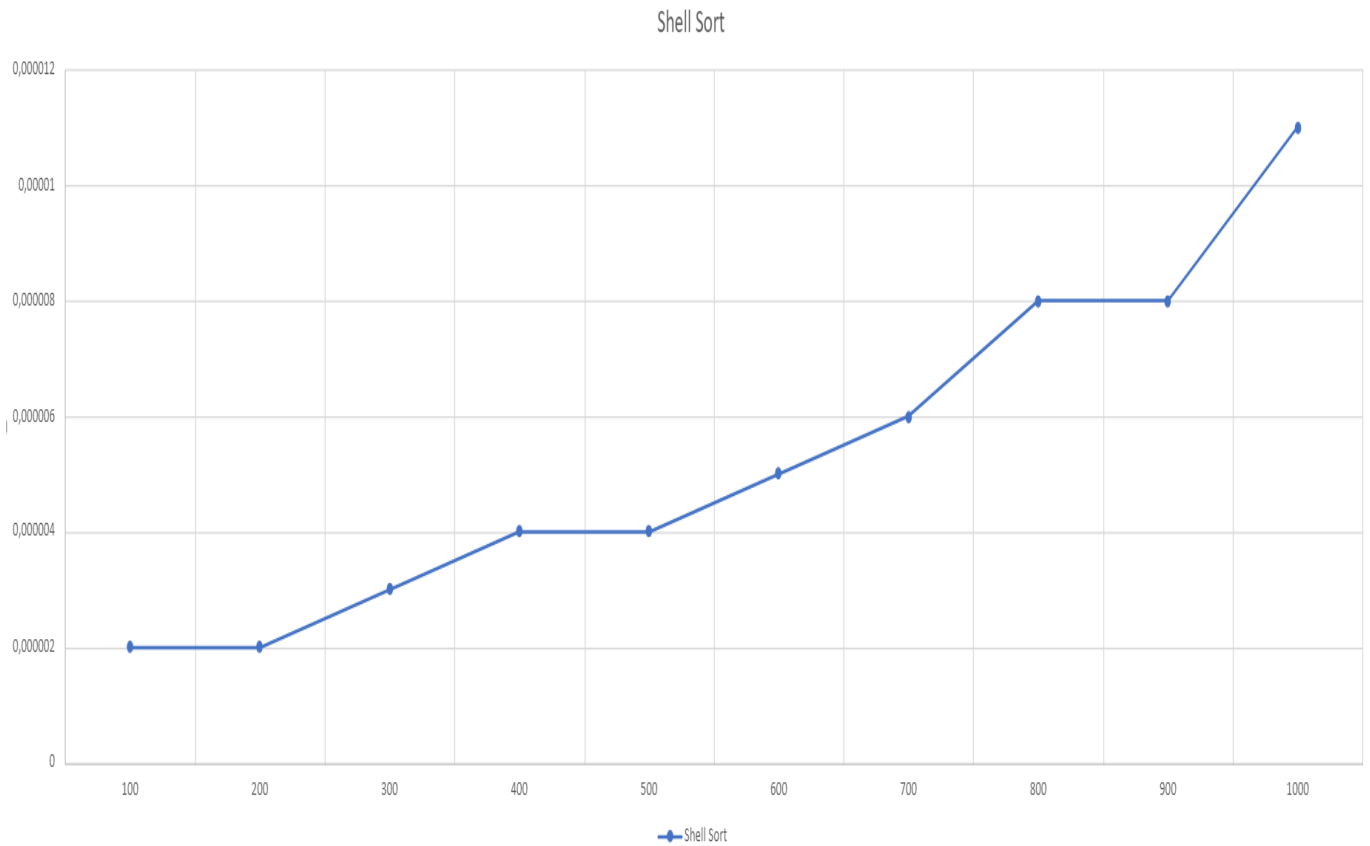
Quick Sort: Το διάγραμμα αυτό δεν διαφέρει σημαντικά με το διάγραμμα της Merge Sort ,δηλαδή η Quick Sort είναι ένας γρήγορος αλγόριθμος

ταξινόμησης.



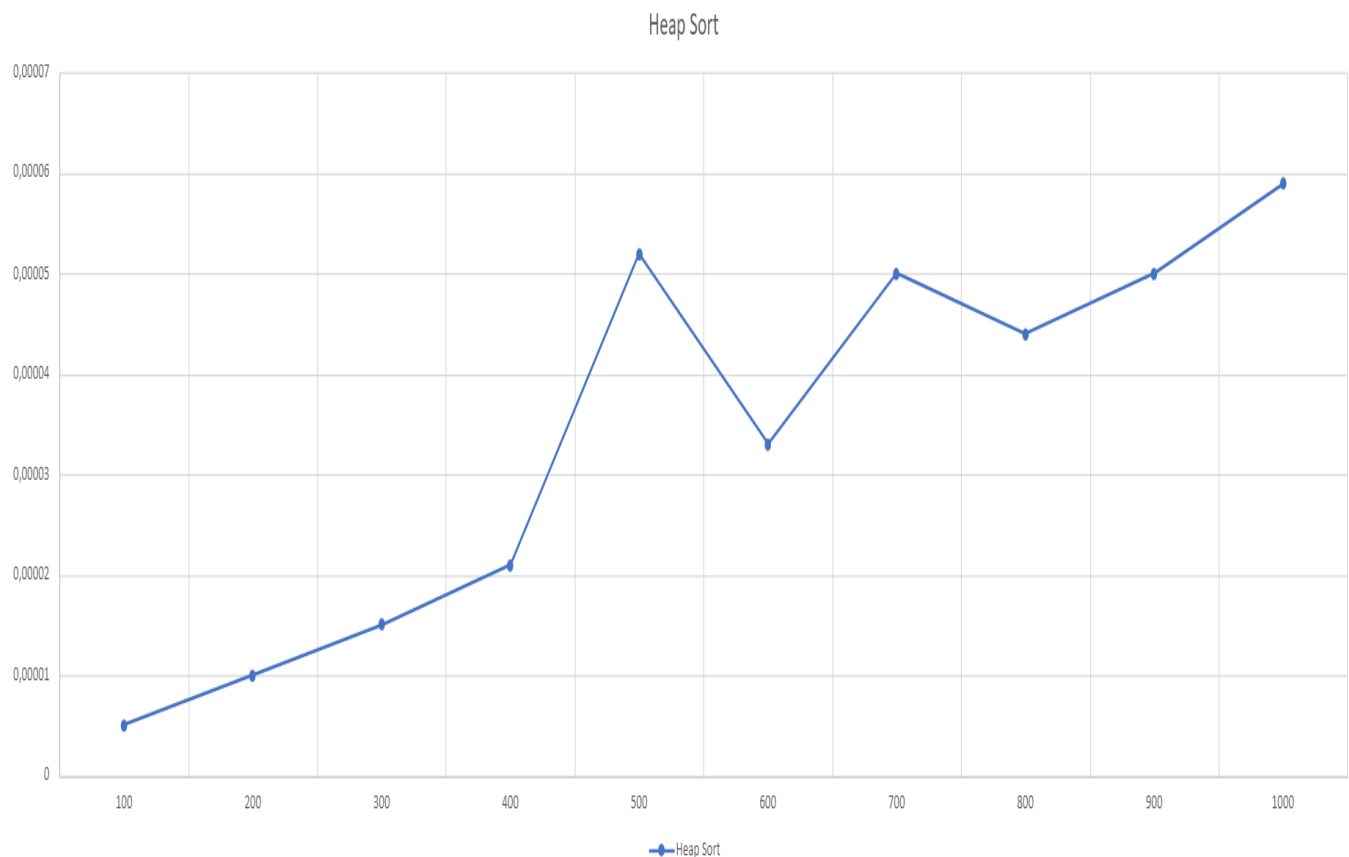
Shell Sort : Η Shell Sort , όπως οι προαναφερθέντες αλγόριθμοι Quick Sort, Merge Sort, αποτελεί μια αποτελεσματική μέθοδο για ταξινόμηση

στοιχείων.



Heap Sort: Τέλος η Heap Sort είναι ένας αλγόριθμος ταξινόμησης όπου ο χρόνος εκτέλεσής της αυξάνεται γραμμικά, εκτός από το πεδίο ακέραιων

500-700 όπου υπάρχουν έντονες αυξομειώσεις.



Ολοκληρώνοντας το βήμα 6, συμπεραίνουμε ότι για 100 εγγραφές, οι χρόνοι ταξινόμησης και των 6 αλγορίθμων διαφέρουν σημαντικά μεταξύ τους. Οι πιο αργοί αλγόριθμοι είναι οι: Selection Sort, Bubble Sort, Insertion Sort, Heap Sort με χρόνους εκτέλεσης: Selection Sort 8 microseconds, Bubble Sort 8 microseconds, Insertion Sort 7 microseconds, Heap Sort 7 microseconds, ενώ οι Merge Sort, Quick Sort, Shell Sort με χρόνους: Merge Sort 3 microseconds, Quick Sort 2 microseconds, Shell Sort 1 microseconds αποτελούν τους ταχύτερους αλγόριθμους ταξινόμησης.

Αφού ολοκληρώσαμε αυτό το πείραμα χρονομέτρησης με διαφορετικά πλήθη ακέραιων και structs, καταλήξαμε σε κάποια συμπεράσματα. Τα συμπεράσματα αυτά τα προσομοιώσαμε σε διαγράμματα τα οποία μας δείχνουν ότι ταυτίζονται με την ύλη που έχουμε διδαχθεί. Όσον αφορά τον χώρο που καταλαμβάνουν οι συναρτήσεις μας, έχοντας υπόψιν μας την ύλη

που έχουμε διδαχθεί καταλήξαμε στο συμπέρασμα ότι οι Quick Sort, Merge Sort και Heap Sort δεσμεύουν περισσότερο χώρο σε σχέση με τις Selection Sort, Shell Sort, Bubble Sort και Insertion Sort διότι η υλοποίησή τους γίνεται με αναδρομικό τρόπο.