

Ασκηση 3

Αρχικά δημιουργήσα 6 αρχεία, 1 αρχείο .h και 1 αρχείο .c για κάθε αλγόριθμο, ώστε να μην βαρύνω την main με πολύ κώδικα. Επίσης άλλαξα την τιμή της maxN στην nonrec_mergesort σε 100000.

Main

Μεταβλητές

```
int *numbers,*numbers_copy,i,j,e;  
clock_t start,end;  
double cpu_time_used,insertionInst,insertionAv=0.0f,insertionInstAv[10],quickInst,quickAv=0.0f,quickInstAv[10],mergeInst,mergeAv=0.0f,mergeInstAv[10];
```

Ακέραιες: Ο δείκτης numbers θα δέχεται τους τυχαίους αριθμούς με την σειρά που ήταν αρχικά, χωρίς δηλαδή να ταξινομηθούν. Ο δείκτης numbers_copy θα παίρνει τους τυχαίους αριθμούς από τον numbers και θα τους δίνει στην συνάρτηση για να ταξινομηθούν. Οι μεταβλητές i,j είναι υπεύθυνες για τον δείκτη των πινάκων και για την δημιουργία των 10 στιγμιοτύπων, αντίστοιχα, ενώ η μεταβλητή e είναι υπεύθυνη για την αύξηση της διάστασης, στο realloc των πινάκων και στην προσθήκη των τυχαίων τιμών στους πίνακες.

Clock t: Η μεταβλητή start παίρνει ως τιμή τον χρόνο στην αρχή της χρονομέτρησης και η μεταβλητή end τον χρόνο στον τέλος της χρονομέτρησης.

Double: Η cpu_time_used αποθηκεύει τον χρόνο που χρειάστηκε για την ταξινόμηση του πίνακα μέσα από αυτήν την μαθηματική πράξη `cpu_time_used = ((double)(end-start))/CLOCKS_PER_SEC;`

Οι μεταβλητές insertionInst, quickInst, mergeInst κρατάνε τον συνολικό χρόνο που χρειάστηκε για την ταξινόμηση των 10 στιγμιοτύπων στην διάσταση, γιαυτό και μηδενίζονται κάθε φορά που αλλάζει η διάσταση.

Οι μεταβλητές insertionInstAv, quickInstAv, mergeInstAv κρατάνε των μέσο όρο των 10 στιγμιοτύπων σε κάθε διάσταση.

Οι μεταβλητές insertionAv, quickAv, mergeAv κρατάνε των γενικό μέσο όρο.

Λοιπές αρχικοποιήσεις πριν την βασική δομή

Αρχικά καλώ την srand με όρισμα time(NULL), ώστε να μην έχω κάθε φορά τους ίδιους τυχαίους αριθμούς. Έπειτα κάνω malloc στην numbers και numbers_copy και τσεκάρω αν η κλήση της malloc έγινε με επιτυχία με την if/else if. Τέλος χρησιμοποιώ την printf και την system("PAUSE") ώστε να ορίσει ο χρήστης την στιγμή που θέλει να ξεκινήσει η πειραματική μελέτη αφού το πρόγραμμα κάνει compile.

Βασική Δομή

Η εξωτερική for είναι υπεύθυνη για την αύξηση των διαστάσεων.

```
for(e=1;e<=10;e++){
```

Στην αρχή της εξωτερικής for κάνω realloc στους numbers και number_copy και τσεκάρω αν το realloc έγινε σωστά με την if/else if (το οποίο δεν χρειάζεται να το κάνω στην πρώτη επανάληψη καθώς έκανα malloc για την ίδια τιμή, όμως είναι απαραίτητο για την συνέχεια) και τέλος θέτω τις τιμές των insertionInst, quickInst, mergeInst σε 0.

Στό τέλος της εξωτερικής for υπολογίζω τον συνολικό μέσο όρο.

Η εσωτερική for είναι υπεύθυνη για την δημιουργία 10 πινάκων τις ίδια όμως διάστασης.

```
mergeInst=0;
for(j=0;j<10;j++){
```

Αρχικά στην εσωτερική for χρησιμοποιώ μια ακόμα for για να αποθηκεύσω τις τυχαίες τιμές στον numbers και στον numbers_copy. Στην συνέχεια, στην αρχή κάθε αλγορίθμου θέτω την τιμή της cpu_time_used σε 0, μετά χρονομετρώ την ταξινόμηση, υπολογίζω την τιμή της cpu_time_used και την αποθηκεύω στην insertionInst. Επίσης πριν από την αρχή του QuickSort και του MergeSort χρησιμοποιώ μια for για να αποθηκεύσω εκ νέου τις αποθηκευμένες τυχαίες τιμές του numbers στο numbers_copy, ώστε να μην τρέξει ο αλγόριθμος με ήδη ταξινομημένο πίνακα. Στο τέλος της εσωτερικής for υπολογίζω τους μέσους όρους για κάθε διάσταση.

Τέλος τυπώνω το πινακάκι με τις τιμές.

Υλοποίηση του προγράμματος και πινακάκι με τιμές

```
--Press any key to start--

Press any key to continue . . .
Instance/Algorithm|InsertionSort|QuickSort|MergeSort
10000              |0.051200    |0.001100 |0.001600
20000              |0.199700    |0.002200 |0.003000
30000              |0.433600    |0.003400 |0.004100
40000              |0.771800    |0.004900 |0.005900
50000              |1.203200    |0.006600 |0.007200
60000              |1.731500    |0.007600 |0.008900
70000              |2.363800    |0.009500 |0.010700
80000              |3.111000    |0.010500 |0.011900
90000              |3.849700    |0.011500 |0.013100
100000             |4.750400    |0.013100 |0.015000
Average            |1.846590    |0.007040 |0.008140

-----
Process exited after 187.7 seconds with return value 0
Press any key to continue . . .
```

Υ.Γ.: Όπου γράφω QuickSort εννοώ τον improved_quicksort και όπου γράφω MergeSort εννοώ τον nonrec_mergesort.