

ΠΕΚΟΣ ΕΥΑΓΓΕΛΟΣ 1115201400157
MENTZELOS ΗΛΙΑΣ 1115201400106
ΕΥΦΡΟΣΥΝΗ ΠΑΝΤΕΛΙΑΔΗ 1115201400301

3η Άσκηση ΥΣΒΔ 2017-2018

Η υλοποίηση της εργασίας έγινε σε γλώσσα C και βρίσκεται στον φάκελο sorted_file_64. Υλοποιήσαμε όλα τα ζητούμενα της εργασίας και είναι πλήρως λειτουργική. Παρακάτω ακολουθεί μια περιγραφή των δομών και των συναρτήσεων και δομών που χρησιμοποιήσαμε για την άσκηση.

ΔΟΜΕΣ

Για την εργασία υλοποιήσαμε μια δομή heap με συναρτήσεις :

```
void update_heap(heap * my_heap, heap_node * node);
```

```
void add_heap(heap * my_heap, heap_node * node);
```

```
int is_empty_heap(heap * my_heap);
```

```
void delete_heap(heap ** my_heap);
```

ΣΥΝΑΡΤΗΣΕΙΣ

- **int Compare(Record* record1, Record* record2, int fieldNo):** Την χρησιμοποιούμε για την σύγκριση των εγγράφων με βάση το fieldNo
- **void Sort_Each_BufferSize_Blocks(int fd_temp , int copied_blocks , int fieldNo, int bufferSize) :** Την χρησιμοποιούμε για το πρώτο βήμα της ταξινόμησης. Ταξινομούμε επιτοπου τα blocks του αρχείου σύμφωνα με το bufferSize. Γιαυτό το βήμα της ταξινόμησης υλοποιήσαμε και κάναμε χρήση της quicksort (**void quicksort(char ** data, int low, int high, int fieldNo)** στην οποία αντίστοιχα χρησιμοποιούμε την **int partition(char ** data, int low, int high, int fieldNo)** η οποία κάνει τους απαραίτητους χωρίσμούς για την quicksort.
- **void Copy_Block(int fd , int fd_temp):** Την χρησιμοποιούμε για να αντιγράψουμε το αρχικό ΜΗ ταξινομημένο αρχείο (που προκύπτει από την SR_ErrorCode SR_InsertEntry(int fileDesc, Record record)) σε ένα προσωρινό έτσι ώστε σε αυτό να αρχίσουμε την ταξινόμηση.
- **char * sort_by_level(int file_id, int level, int fieldNo, int bufferSize):** Εδώ γίνονται τα επόμενα επίπεδα της ταξινόμησης. Δημιουργεί ένα νέο αρχείο όπου τα blocks είναι ταξινομημένα. Πιο συγκεκριμένα προκειται για μια επαναληψη όπου φτιάχνει ένα προσωρινό αρχείο και έχει bufferSize – 1 blocks όπου από τα οποία διάβάζει κάθε φορά το μικρότερο στοιχείο με την χρήση του heap και γράφει στο νέο αρχείο. Έτσι σε κάθε επανάληψη ταξινομούνται (bufferSize – 1) * blocks μέχρι να ταξινομηθούν όλα.

- **void SR_ReadRecord(void * data, Record * record) :** Διαβάζει μια εγγραφή από το block
- **void SR_PrintRecords(Record record) :** Γράφει μια εγγραφή στο block
- **void SR_PrintRecords(Record record) :** Εκτυπώνει μια εγγραφή από το block