

**Χειμερινό Εξάμηνο 2023-2024**  
**Υλοποίηση Συστημάτων Βάσεων Δεδομένων**  
**Διδάσκων Γ. Ιωαννίδης**  
**1η Εργασία**

**Στυλιανός Δημητριάδης 1115201900050**  
**Παναγιώτης Κοντοειδής 1115201900266**

Για την εκτέλεση του προγράμματος χρησιμοποιούνται οι εντολές:

- make hp
- ./build/hp\_main

Για την υλοποίηση των συναρτήσεων και την λειτουργικότητα αυτών συμπληρώθηκαν οι δομές **HP\_info** και **HP\_block\_info** στο αρχείο include/hp\_file.h.

Συγκεκριμένα, η πρώτη δομή περιέχει πληροφορίες σχετικά με το αρχείο όπως, την χωρητικότητα του κάθε μπλοκ σε records, το id του τελευταίου μπλοκ και έναν file descriptor για το αρχείο που θα είναι ανοιχτό. Η δεύτερη δομή αφορά το κάθε μπλοκ και περιέχει τον αριθμό των εγγραφών του μπλοκ την δεδομένη χρονική στιγμή και έναν δείκτη στο επόμενο μπλοκ.

**Σχεδιαστικές επιλογές των ζητούμενων συναρτήσεων.**

### **HP\_CreateFile**

Στην συνάρτηση αυτή δημιουργείται και ανοίγεται ένα αρχείο με blocks ενώ δημιουργείται και αρχικοποιείται ένα καινούργιο μπλοκ το οποίο περιέχει τις βασικές πληροφορίες του heal file αρχείου. Στο block αυτό εισάγονται πληροφορίες στην δομή hp info, όπως ο file descriptor του αρχείου τα records που χωράνε σε κάθε block και το id του τελευταίου block. Όταν γίνουν οι κατάλληλες αρχικοποιήσεις το block γίνεται dirty unpined και καταστρέφεται, ενώ ακόμα κλείνει το αρχείο BF.

### **HP\_OpenFile**

Η συνάρτηση καλεί την BF\_OpenFile για να ανοίξει το αρχείο με τα blocks. Στην συνέχεια, γίνεται init ένα μπλοκ, το αρχικό block και με την malloc δεσμεύεται η κατάλληλη μνήμη ώστε το hp info με την βοήθεια της memcpy να μετακινηθεί στην κύρια

μνήμη και να επιστραφεί απο την συνάρτηση. Τέλος, γίνεται ξεκαρφίτσωμα του block και καταστροφή του και επιστρέφεται το hp info.

## HP\_InsertEntry

Η εισαγωγή μιας εγγραφής απαιτεί τον έλεγχο του αριθμού των μπλοκ που περιέχει το αρχείο. Αν υπάρχει μόνο ένα μπλοκ, το αρχικό που περιέχει τα δεδομένα του αρχείου, τότε δημιουργείται ένα νέο μπλοκ στο οποίο τοποθετείται η εγγραφή εκεί. Διαφορετικά, σε περίπτωση που υπάρχουν και άλλα μπλοκ στο αρχείο, εντοπίζεται το τελευταίο μπλοκ και εξετάζεται αν χωράει η εγγραφή στον διαθέσιμο χώρο του. Αν ναι τότε η εγγραφή εισάγεται στο μπλοκ αλλιώς δημιουργείται ένα καινούργιο μπλοκ όπως και προηγουμένως.

Επίσης, έχουν δημιουργηθεί οι συμπληρωματικές συναρτήσεις:

**get\_metadata:** εντοπίζει και κάνει τον δείκτη να δείξει στην περιοχή που βρίσκονται τα metadata του block

**insert\_record:** ελέγχει αν μπορεί να εισαχθεί νέα εγγραφή στο μπλοκ και αντιγράφει τα δεδομένα με memcpy και αυξάνει τον αριθμο των records in block στα μεταδεδομένα κατα 1. Αν αυτό γίνει με επιτυχία επιστρέφει 0, αλλιώς επιστρέφεται -1.

## HP\_GetAllEntries

Ελέγχονται σειριακά σε κάθε μπλοκ όλες οι εγγραφές του μέχρι να βρεθούν όλες αυτές οι εγγραφές με id ίσο με το value που έχει δοθεί. Παράλληλα επιστρέφεται ο αριθμός των μπλοκ που διαβάστηκαν μέχρι να βρεθούν όλες οι εγγραφές. Σε περίπτωση λάθους επιστρέφεται -1.

## examples / hp\_main.c

Η main συνάρτηση τεστάρει την λειτουργικότητα των παραπάνω συναρτήσεων. Δημιουργείται ένα καινούργιο αρχείο με την HP\_CreateFile και ανοίγει με την HP\_OpenFile. Στο αρχείο αυτό εισάγονται RECORDS\_NUM τυχαίες εγγραφές και επιλέγεται ένα τυχαίο id με την rand για να τυπωθούν όλες οι εγγραφές με το ίδιο id με

την βοήθεια της HP\_GetAllEntries. Τέλος η HP\_GetAllEntries εκτυπώνει και όλα τα blocks που διαβάστηκαν μέχρι να βρεθούν όλες οι εγγραφές με το συγκεκριμένο id.

### **HP\_Print\_All\_Records**

Μιά απλή συνάρτηση που χρειάστηκε για το testing του κώδικα, για κάθε block καλεί την συνάρτηση HP\_Print\_Block\_Records για να εκτυπωθούν όλα τα records του.

### **HP\_Print\_Block\_Records**

Εκτυπώνει όλα τα records που υπάρχουν σε ένα block