

Θεολογή Μαριάνθη Α.Μ. 1115201400047
Λοξός-Ανδρέου Ιάσων Α.Μ. 1115201300085

Εργασία Στατικού Κατακερματισμού και χρήση δευτερεύοντος Ευρετηρίου

Η εργασία υλοποιεί όλα τα ζητούμενα της εκφώνησης και για τον στατικό κατακερματισμό και για το δευτερεύων ευρετήριο.

Τα αρχεία του Στατικού Κατακερματισμού είναι τα HT.c και HT.h στα οποία υλοποιούνται με επιτυχία οι συνρτήσεις που μας ζητούνται και φτιάξαμε επιπλέον τις συναρτήσεις:

- `int hash_function()`: συνάρτηση κατακερματισμού
- `int Insert_in_new_Block()`: εισάγει την εγγραφή σε νέο block το οποίο κάνει allocate, το αρχικοποιεί και το επιστρέφει
- `void delete_from_block()`: διαγράφει μια εγγραφή από ένα συγκεκριμένο block και φτιάχνει τη σειρά των εγγραφών που απομένουν

Τα αρχεία του Δευτερεύοντος Ευρετηρίου είναι τα SHT.c και SHT.h στα οποία επίσης υλοποιούνται με επιτυχία οι συναρτήσεις που μας ζητούνται και οι επιπλέον συναρτήσεις που φτιάξαμε είναι οι:

- `int secondary_hash_function()`: συνάρτηση κατακερματισμού για int, strings
- `int Secondary_Insert_in_new_Block()`: εισάγει την εγγραφή (blockId) σε νέο block το οποίο κάνει allocate ,το αρχικοποιεί και το επιστρέφει

Για τον Στατικό κατακερματισμό πήραμε 3 ήδη block:

- το block 0 που είναι το αναγνωριστικό και κρατάει τις πληροφορίες που μας ζητούνται
- το block 1 που κρατάει τον πίνακα που λειτουργεί σαν ευρετήριο(μεγέθους όσα τα buckets που μας δίνονται)
- τα υπόλοιπα block κρατάνε τα records και στην 1η και 2η θέση δύο integers (block_recs, next) που έχουν τις πληροφορίες για το πόσα records έχει μέσα το block και ποιο είναι το επόμενο block(σε περίπτωση overflow)

Αυτή τη λογική ακολουθούμε και για το δευτερεύων ευρετήριο μόνο που σε αυτό κρατάμε το blockId του πρωτεύοντος αντί για ολόκληρο το record. Υπάρχουν αρκετά σχόλια στον κώδικα για την κατανόησή του.

Υλοποιήσαμε επίσης δική μας main συνάρτηση την main_test.c με την οποία εξετάζουμε τη λειτουργικότητα του προγράμματος όπως μας ζητείται στην εκφώνηση.

Σαν παραμέτρους από το input παίρνουμε το αρχείο με τα records και δύο αριθμούς που καθορίζουν το εύρος των εγγραφών που θα αναζητήσουμε με την GetAllEntries().

Παράδειγμα: ./HT records1K.txt 5 100 που θα εμφανίσει τα records με id 5 μέχρι 100

Στην `main_test.c` έχουμε φτιάξει τη συνάρτηση `InsertRecord()` η οποία παίρνει το αρχείο που έχουμε δώσει στο `input` και με τη χρήση της `getline()` και `strtok()`, φτιάχνουμε τα `records` τα οποία μετά εισάγουμε στο πρωτεύων ευρετήριο και στη συνέχεια αν έχουμε και δευτερεύων τα εισάγουμε και σε αυτό.

Το πρόγραμμα τρέχει κανονικά και χωρίς λάθη και με την `ht_main_test.c` που μας δώθηκε.

Στο τέλος εκτυπώνονται κάποια στατιστικά και για τα δύο αρχεία (πρωτεύοντος και δευτερεύοντος ευρετηρίου) καλώντας τη συνάρτηση `HashStatistics()` την οποία διαμορφώσαμε έτσι ώστε να λειτουργεί και για τα δύο είδη αρχείων.

Το πρόγραμμα υλοποιήθηκε και μεταγλωττίστηκε σε περιβάλλον `linux`, μεταγλωττίζεται με το `Makefile` και χρησιμοποιούμε τη βιβλιοθήκη `BF_64.a`