

Υλοποίηση Συστημάτων Βάσεων Δεδομένων

Εργασία 1

Ονοματεπώνυμο:

Γιαμπουόνκα-Κανελλάκος Κάρολος

Χαραλαμπίδης Γεώργιος

Αριθμός Μητρώου:

1115201600030

1115201600193

Η εργασία υλοποιήθηκε σύμφωνα με τα ζητούμενα της εκφώνησης και υποδείξεις από τις συζητήσεις στο eclass.

Το πρωτεύον ευρετήριο(HT) χρησιμοποιεί πάντα το id ως κλειδί.

Το δευτερεύον ευρετήριο(SHT) έχει 3 πιθανά κλειδιά, name, surname ή address τα οποία θέλουν το attrName να είναι "name", "surname" και "address" αντίστοιχα.

Κατά τη δημιουργία του δευτερεύοντος ευρετηρίου, όλες οι εγγραφές που υπάρχουν στο πρωτεύον ευρετήριο συγχρονίζονται με το δευτερεύον.

Όταν διαγράφεται ένα Record τότε αλλάζει το id του σε μια defined τιμή DELETED_RECORD_ID στο HT και το αντίστοιχο κλειδί σε DELETED_RECORD_SHT στο SHT. Ωστόσο δεν έχουμε για διαγραφή ενός Record από το SHT.

Στο SHT γράφουμε στο Block το id του record, το κλειδί του SHT καθώς και το BlockId στο οποίο βρίσκεται το Block στο πρωτεύον ευρετήριο.

Στο πρώτο Block κάθε ευρετηρίου υπάρχει ο πίνακας hash_table, που είναι πίνακας από integers με θέσεις ίσες με τον αριθμό των buckets. Κάθε θέση του πίνακα αρχικοποιείται με -1 δείχνοντας πως δεν έχουν τα buckets Blocks. Μόλις χρειαστεί να κάνουμε insert σε ένα bucket, άμα αυτό δεν έχει Blocks μέσα του, δημιουργούμε ένα Block και δίνουμε στην αντίστοιχη θέση

του πίνακα `hash_table` το `id` του πρώτου Block στο `bucket`(αυτό ππου δημιουργήσαμε δηλαδή).

Ο αριθμός των Records ανα Block γίνεται `define` στο αρχείο `HT.h`.

Ο μέγιστος αριθμός `buckets` που μπορούμε να έχουμε είναι 120, λόγω του στατικά δεσμευμένου πίνακα `hash_table` και το μέγεθος των 512 Bytes στο Block.

Μαζί με τα αρχεία υπάρχει μια `main` συνάρτηση που κάνει 6 πράγματα που ζητούνται στην εκφώνηση.

Εντολή για `compile` σε 64-bit λειτουργικό:

`gcc -o myprog main.c BF_64.a -no-pie`

Εντολή για εκτέλεση:

`./myprog`

Δε χρειάζεται κάποιο όρισμα για την απλότητα της εκτέλεσης.