

ΕΡΓΑΣΙΑ 1

Σκοπός της εργασίας αυτής είναι η κατανόηση της εσωτερικής λειτουργίας των Συστημάτων Βάσεων Δεδομένων όσον αφορά τη διαχείριση σε επίπεδο μπλοκ (block) αλλά και ως προς τη διαχείριση σε επίπεδο εγγραφών. Επίσης, μέσω της εργασίας θα γίνει αντιληπτό το κατά πόσο μπορεί να βελτιώσει την απόδοση ενός Συστήματος Διαχείρισης Βάσεων Δεδομένων (ΣΔΒΔ) η ύπαρξη ευρετηρίων πάνω στις εγγραφές. Υλοποιήσαμε ένα σύνολο συναρτήσεων που διαχειρίζονται αρχεία που δημιουργήθηκαν βάσει στατικού κατακερματισμού (Hash Table) και ένα σύνολο συναρτήσεων που διαχειρίζονται αρχεία που δημιουργήθηκαν βάσει επεκτατού κατακερματισμού (Extendible Hash Table).

Το πρόγραμμα αναπτύχθηκε σε περιβάλλον Linux (Ubuntu), σε γλώσσα C.

Μέρος Α: Συναρτήσεις HT (Hash Table)

Υλοποιήθηκαν οι εξής συναρτήσεις:

1. HT CreateIndex (δημιουργία και αρχικοποίηση αρχείου κατακερματισμού)
2. HT OpenIndex (άνοιγμα αρχείου κατακερματισμού, ανάγνωση πρώτου block σε αυτό και ενημέρωση δομής πληροφοριών)
3. HT CloseIndex (κλείσιμο αρχείου και αποδέσμευση χώρου δομής πληροφορίας)
4. HT InsertEntry (εισαγωγή εγγραφής στο αρχείο κατακερματισμού)
5. HT GetAllEntries (εκτύπωση όλων των εγγραφών ενός αρχείου με συγκεκριμένη τιμή πεδίου-κλειδιού, καθώς και το πλήθος των blocks που διαβάστηκαν)
6. HashStatistics (Εκτύπωση στατιστικών ου αφορούν σε: α) Το πόσα blocks έχει ένα αρχείο, β) Το ελάχιστο, το μέσο και το μέγιστο πλήθος εγγραφών που έχει κάθε bucket ενός αρχείου, γ) Το μέσο αριθμό των blocks που έχει κάθε bucket, και δ) Το πλήθος των buckets που έχουν μπλοκ υπερχείλισης, και πόσα μπλοκ είναι αυτά για κάθε bucket)

Βοηθητικές συναρτήσεις:

1. HT findTypeOfFile (Εύρεση τύπου αρχείου για έλεγχο ορθότητας βάσει fileDesc)
2. init HT bucket (αρχικοποίηση bucket για αρχείο στατικού κατακερματισμού)
3. decimalToBinary (μετατροπή δεκαδικού σε δυαδικό)
4. BinaryToDecimal (μετατροπή δυαδικού σε δεκαδικό)
5. hashFunction (η συνάρτηση κατακερματισμού)
6. WriteToBucket (εγγραφή record στο αντίστοιχο bucket, με έλεγχο overflow)

Για να γίνει compile, βρισκόμενοι στον σωστό φάκελο στο terminal, πληκτρολογούμε την εντολή

\$make t1

οπότε προκύπτει το executable αρχείο execHT.

Για την συνάρτηση κατακερματισμού:

Η hashFunction παίρνει ως ορίσματα δυο ακεραίους, *n* και *numOfBuckets*, και χρησιμοποιεί τον αριθμό 128 για την εύρεση υπολοίπου διαίρεσης, επιστρέφοντας τον δεκαδικό αριθμό που

αντιστοιχεί στα πρώτα bucket bits του αποτελέσματος του hashing, δηλαδή του modulo, σε δυαδικό (char*).

Μέρος B: Συναρτήσεις ET (Extendible Hash Table)

Υλοποιήθηκαν οι εξής συναρτήσεις:

1. EH CreateIndex (δημιουργία και αρχικοποίηση αρχείου κατακερματισμού)
2. EH OpenIndex (άνοιγμα αρχείου κατακερματισμού, ανάγνωση πρώτου block σε αυτό και ενημέρωση δομής πληροφοριών)
3. EH CloseIndex (κλείσιμο αρχείου και αποδέσμευση χώρου δομής πληροφορίας)
4. EH InsertEntry (εισαγωγή εγγραφής στο αρχείο κατακερματισμού)
5. EH GetAllEntries (εκτύπωση όλων των εγγραφών ενός αρχείου με συγκεκριμένη τιμή πεδίου-κλειδιού, καθώς και το πλήθος των blocks που διαβάστηκαν)
6. HashStatistics (Εκτύπωση στατιστικών ου αφορούν σε: α) Το πόσα blocks έχει ένα αρχείο, β) Το ελάχιστο, το μέσο και το μέγιστο πλήθος εγγραφών που έχει κάθε bucket ενός αρχείου)

Βοηθητικές συναρτήσεις:

1. ET findTypeOfFile (Εύρεση τύπου αρχείου για έλεγχο ορθότητας βάσει fileDesc)
2. init ET bucket (αρχικοποίηση bucket για αρχείο στατικού κατακερματισμού)
3. IndexSize (εύρεση μεγέθους βάσει βάθους)
4. decimalToBinary (μετατροπή δεκαδικού σε δυαδικό)
5. BinaryToDecimal (μετατροπή δυαδικού σε δεκαδικό)
6. hashFunction (η συνάρτηση κατακερματισμού)
7. WriteToBucket (εγγραφή record στο αντίστοιχο bucket, με έλεγχο overflow)
8. doubleIndex (διπλασιασμός index βάσει θεωρίας)

Για να γίνει compile, βρισκόμενοι στον σωστό φάκελο στο terminal, πληκτρολογούμε την εντολή

\$make t2

οπότε προκύπτει το executable αρχείο execEH.

Για την συνάρτηση κατακερματισμού:

Η hashFunction παίρνει ως ορίσματα δυο ακεραίους, *n* και *depth*, και χρησιμοποιεί τον αριθμό 128 για την εύρεση υπολοίπου διαίρεσης, επιστρέφοντας τον δεκαδικό αριθμό που αντιστοιχεί στα πρώτα bucket bits του αποτελέσματος του hashing, δηλαδή του modulo, σε δυαδικό (char*).

Γενικές σημειώσεις:

1. Οι συναρτήσεις *getAllEntries* και *HashStatistics*, και για τους δύο τύπους κατακερματισμών δεν είναι πλήρεις.
2. Έγινε χρήση εξίσου των δοθείσων *main*s καθώς και νέων (συμπεριλαμβάνονται στον φάκελο της εργασίας). Το Makefile κάνει χρήση των νέων *main*.
3. Η εκτέλεση του *execHT* εμφανίζει error στην συνάρτηση δημιουργίας *index*.
4. Η εκτέλεση του *execHT* εμφανίζει *segmentation fault*.