

PROJECT 2018-19

Ομάδα:

Νικόλαος Τσιουγκράνας (1115201500166)

Αλέξανδρος Νιάρχος (1115201500108)

Περίληψη:

Η εφαρμογή που υλοποιήσαμε ικανοποιεί όλες τις απαιτήσεις των εκφωνήσεων. Επίσης έχει ελεγχθεί με valgrind και δεν υπάρχουν memory leaks. Ακολουθήσαμε όλες τις οδηγίες και υλοποιήσαμε όλες τις δομές που μας ζητήθηκαν στο πλαίσιο της άσκησης. Έχει γίνει Unit Testing με (Gtest). Αρχικά η εφαρμογή παίρνει το input από το harness τρέχει τα ερωτήματα ένα-ένα και κάνει παραλληλοποίηση μέσα στην συνάρτηση RadixHashJoin για κάθε ένα ζεύγος από τα predicates, να αναφέρουμε ότι η σειρά με την οποία τρέχουν τα predicates καθορίζεται από τις δοθέντες εκτιμήτριες και διαλέγονται αυτά με το μικρότερο κόστος. Στην συνέχεια μετά το Join επιστρέφονται ένα-ένα τα αποτελέσματα από τα queries τα οποία ομαδοποιούνται σε batches και επιστρέφονται στο harness μέσα από το output.

Χρόνοι:

Αριθμός threads	Χρόνος (μ.ο. Πέντε εκτελέσεων)
1	1.765 sec
2	1.720 sec
4	1.723 sec
8	1.835 sec
16	1.842 sec

Παρατηρήσεις:

Οι χρόνοι καταγράφηκαν στο σύστημα με επεξεργαστή Ryzen 7 2700x stock clocks(8 cores-16 threads) σε ubuntu 18 64bit σε VMware έχοντας δώσει όλα τα threads του επεξεργαστή στο VM για μεγαλύτερη συνέπεια. Αρχικά παρατηρήσαμε αυτό που ειπώθηκε στο μάθημα για τις εκτιμήτριες, όχι μόνο δεν βελτιώθηκε ο χρόνος αλλά χειροτέρεψε κατά 0.3 δευτερολεπτα άρα περίπου κατά 25%. Σχετικά με τον πολυνηματισμός λόγω του μεγάλου κόστους επανένωσης των αποτελεσμάτων τους μερικές φορές τα περισσότερα threads ήταν πιο αργά από τα λιγότερα ενώ κάποιες φορές το αντίθετο στο ίδιο input με αποτέλεσμα να δημιουργείται ασυνέπεια.