

Γεώργιος Δημόπουλος

A.M. 2964

2^η σειρά ασκήσεων στο μάθημα: Διαχείριση Σύνθετων Δεδομένων
(κ.Μαμουλής)

Αρχικά για να τρέξετε το πρόγραμμα έχω προσθέσει 2 ορίσματα, ώστε να μου τα δίνει ο χρήστης από την γραμμή εντολών. Το πρώτο όρισμα είναι το όνομα του αρχείου και το δεύτερο όρισμα είναι το block size του R-tree δέντρου. Ένα παράδειγμα για το πως τρέχει: **python ergasia2.py data_rectangles.txt 1024**. Τέλος να αναφέρω ότι το αρχείο με τα ορθογώνια των ερωτήσεων το έχω βάλει σε ένα txt αρχείο, το οποίο το έχω ονομάσει **query_rectangles.txt**. Το καλώ αυτό το αρχείο ώστε να αποτιμηθούν όλες οι ερωτήσεις για όλα τα ορθογώνια του αρχείου.

ΜΕΡΟΣ 1^ο:

Στο πρώτο μέρος ουσιαστικά πρέπει να φτιάξουμε το R-tree δέντρο και να τυπώνουμε διάφορα στατιστικά του. Αρχικά θέλω να υπολογίσω των αριθμών των ορθογωνίων που είναι ίσος με το μέγεθος της λίστας του αρχείου. Έπειτα υπολογίζω πόσα ορθογώνια χωράνε σε κάθε κόμβο από τον τύπο $f = \text{floor}(\text{block_size}/36)$. Ο αριθμός των φύλλων προκύπτει από τον τύπο $L = \text{ceil}(N/f)$. Θέλω τώρα να υπολογίσω πόσα επίπεδα έχει το δέντρο μου και από πόσους κόμβους αποτελείται. Αυτό που κάνω είναι απλά να διαιρώ το προηγούμενο επίπεδο δια το f και να παίρνω το πάνω ακέραιο μέρος μέχρι το αποτέλεσμα μου να γίνει ίσο με ένα, που σημαίνει ότι έφτασα στην ρίζα. Τώρα πλέον μπορώ να δημιουργήσω τα φύλλα σύμφωνα με τον αλγόριθμο sort-tille-recursive (STR). Αυτό που κάνω για να καταλάβω αν γέμισε ένας κόμβος είναι απλά να βρίσκω το mod από τα ομαδοποιημένα ορθογώνια με το πόσα ορθογώνια χωράνε σε έναν κόμβο και αν γίνει ίσο με το $f-1$ ή απλά δεν έχουμε άλλα ορθογώνια, τότε δημιουργείται ένας κόμβος. Για όλα τα υπόλοιπα επίπεδα μέχρι την ρίζα ισχύει ότι ένας κόμβος του πιο κάτω επιπέδου αποτελεί μια εγγραφή στον κόμβο του αμέσως πιο πάνω επιπέδου και λειτουργεί σαν δείκτης. Για αυτό λοιπόν για όλους τους κόμβους του αμέσως πιο κάτω επιπέδου επιλέγω το μικρότερο x_low το μεγαλύτερο x_high το μικρότερο y_low και το μεγαλύτερο y_high από όλες τις εγγραφές του, έτσι ώστε να δημιουργηθεί ένα καινούριο μεγαλύτερο ορθογώνιο το id του οποίου θα δείχνει σε όλα αυτά τα ορθογώνια που περιέχει μέσα. Για την δημιουργία κόμβων χρησιμοποιώ την ίδια λογική με τα φύλλα. Για κάθε επίπεδο επίσης υπολογίζω το μέσο εμβαδόν των ορθογωνίων. Το ύψος του δέντρου είναι όσο ο αριθμός των επιπέδων. Οπότε τυπώνω όλα αυτά που μας ζητάτε στο terminal. Τέλος όπως μας ζητήσατε γράφω σε ένα αρχείο

rtree.txt όλο το r-tree δέντρο μου, όπου η πρώτη γραμμή έχει το id της ρίζας και η δεύτερη γραμμή από πόσα επίπεδα αποτελείται ο κόμβος.

ΜΕΡΟΣ 2°:

Για το δεύτερο μέρος της εργασίας συμβουλευτήκα το παρακάτω link:

<https://www.geeksforgeeks.org/find-two-rectangles-overlap/>

Έτσι λοιπόν δημιούργησα μια συνάρτηση *intersection* η οποία ελέγχει αν ένα ορθογώνιο τέμνει κάποιο άλλο. Συγκεκριμένα ελέγχω αν ένα ορθογώνιο βρίσκεται πάνω, κάτω, δεξιά ή αριστερά από ένα άλλο ορθογώνιο. Αν δεν βρίσκεται σε καμία τέτοια θέση, τότε σημαίνει ότι τα ορθογώνια τέμνονται. Για την πρώτη ερώτηση δημιουργώ την συνάρτηση *intersection_query*. Αυτό που κάνω είναι να βάζω σαν αρχικό κόμβο την ρίζα και να ελέγχω αν κάποιο ορθογώνιο τέμνει κάποιο από τα ορθογώνια της ρίζας. Αν ναι τότε πάω στον κόμβο που μου δείχνει η ρίζα κ.ο.κ. Αν τώρα φτάσω σε φύλλο, ελέγχω αν κάποιο ορθογώνιο τέμνει το συγκεκριμένο ορθογώνιο. Για την δεύτερη ερώτηση δημιουργώ την συνάρτηση *inside(RCT_q, RCT)*, η οποία ελέγχει αν στο ορθογώνιο *RCT_q* περιέχεται το ορθογώνιο *RCT*. Τώρα καλώ την συνάρτηση *inside_query*. Ξεκινάω και πάλι από την ρίζα. Μέχρι να φτάσω σε κάποιο φύλλο ελέγχω αν το ορθογώνιο μου τέμνει κάποια από τα ορθογώνια των κόμβων του δέντρου μου. Αν τώρα φτάσω σε φύλλο ελέγχω ποια από τα ορθογώνια των φύλλων περιέχονται στο ορθογώνιο μου με την συνάρτηση *inside*. Για την τρίτη ερώτηση δημιουργώ την συνάρτηση *containment_query*. Ξεκινάω από την ρίζα και μέχρι να φτάσω σε κάποιο φύλλο ελέγχω αν το ορθογώνιο μου τέμνει κάποια από τα ορθογώνια των κόμβων που επισκέπτομαι. Αν φτάσω σε φύλλο ελέγχω ποια από τα ορθογώνια των φύλλων περιέχουν το ορθογώνιο μου με την συνάρτηση *inside*.