

ΠΑΝΕΠΙΣΤΗΜΙΟ ΙΩΑΝΝΙΝΩΝ
ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ Η/Υ &
ΠΛΗΡΟΦΟΡΙΚΗΣ
ΜΥΕ041 - ΔΙΑΧΕΙΡΙΣΗ ΣΥΝΘΕΤΩΝ
ΔΕΔΟΜΕΝΩΝ
ΕΡΓΑΣΙΑ 2

Κίτσιος Κωνσταντίνος 4388

May 5, 2023

Contents

1	Πηγαίος κώδικας	3
2	Μέρος 1	4
3	Μέρος 2	4
4	Μέρος 3	4

1 Πηγαίος κώδικας

Ο πηγαίος κώδικας της εργασίας αποτελείται από τρία αρχεία. Το αρχείο `spatial_index.py` είναι υπεύθυνο για τη δημιουργία του grid και την παραγωγή των αρχείων `grid.dir` & `grid.grd`, γιαυτό και εκτελείται μία φορά (μέρος 1). Το αρχείο `selection_window.py` είναι υπεύθυνο για τα ερωτήματα παραθύρου (μέρος 2) και το αρχείο `refinement_step.py` είναι υπεύθυνο για το refinement step που πραγματοποιείται μετά από τα ερωτήματα παραθύρου (μέρος 3). Πρακτικά ο κώδικας είναι παρόμοιος με αυτόν του `selection_window.py`, απλά για ευκολία και μικρότερα αρχεία μεταφέρθηκε σε ξεχωριστό αρχείο.

Η εκτέλεση του `spatial_index.py` γίνεται με την εντολή: `python3 spatial_index.py`. Για την εκτέλεση των `selection_window.py` & `refinement_step.py` απαιτείται το `queries.txt` να βρίσκεται στον ίδιο κατάλογο με τα αρχεία.

Χρησιμοποιείται η εντολή: `python3 selection_window.py`, `python3 refinement_step.py`

2 Μέρος 1

Στο πρώτο μέρος (αρχείο `spatial_index.py`), αρχικά διαβάζονται τα δεδομένα του dataset, αποθηκεύονται σε ένα λεξικό με κλειδί το id του αντικειμένου και ως τιμή μια πλειάδα με το MBR και το linestring του αντικειμένου. Έπειτα δημιουργείται το grid και τέλος παράγονται τα αρχεία `grid.dir` & `grid.grd` τα οποία θα βοηθήσουν στην ανακατασκευή του grid στο δεύτερο μέρος.

3 Μέρος 2

Στο δεύτερο μέρος (αρχείο `selection_window.py`), χρησιμοποιώντας τα παραγόμενα αρχεία του πρώτου μέρους, ανακατασκευάζεται το αρχικό grid. Παράγονται δύο δομές, ένα λεξικό με κλειδί τις συντεταγμένες του κάθε κελιού και ως τιμή μία λίστα με τα id των object που βρίσκονται στο κελί και ένα λεξικό με κλειδί το id του object και ως τιμή δύο λεξικά, ένα με κλειδί το string 'mbr' και τιμή το mbr του object και ένα με κλειδί το string 'coords' και τιμή το linestring του object. Έπειτα, γίνεται ανάγνωση του αρχείου `queries.txt` και αποθηκεύονται οι συντεταγμένες του παραθύρου. Στο πρώτο στάδιο ελέγχονται τα κελιά τα οποία τέμνει το παράθυρο και τα αποθηκεύει σε μία δομή και έπειτα για κάθε κελί και για κάθε αντικείμενο ελέγχεται εάν το MBR του τέμνει το παράθυρο. Καθώς μερικά αντικείμενα μπορεί να υπάρχουν ως αντίγραφα σε πάνω από ένα κελιά, μπορεί να καταμετρηθούν πάνω από μία φορά το καθένα στα τελικά αποτελέσματα. Για να το αποφύγουμε αυτό, αναθέτουμε ως σημείο αναφοράς του κάθε αντικειμένου (reference point) το κάτω αριστερά σημείο της τομής του MBR του με το παράθυρο του ερωτήματος. Αυτό το υπολογίζουμε βρίσκοντας τις μέγιστες τιμές για το x και το y από τις ελάχιστες των δύο (MBR και παράθυρο). Στην συνέχεια, κάθε ξεχωριστό αντικείμενο θα αναφέρεται ως αποτέλεσμα μόνο εάν το σημείο αναφοράς του βρίσκεται μέσα στο κελί το οποίο εξετάζεται εκείνη την δεδομένη στιγμή.

4 Μέρος 3

Το τρίτο μέρος (αρχείο `refinement_step.py`) δεν έχει κάποια διαφορετική συνάρτηση από αυτές του αρχείο `selection_window.py`, παρά μόνο τις απαραίτητες γραμμές κώδικα για τον έλεγχο του `refinement_step`. Ειδικότερα, μετά την εύρεση των αντικειμένων των οποίων το MBR τέμνεται με το παράθυρο επιλογής, για κάθε αντικείμενο, ελέγχουμε εάν το MBR υπερκαλύπτεται από το παράθυρο στον x & y άξονα (αν υπερκαλύπτεται προστίθεται στο αποτέλεσμα). Αν δεν ισχύει κάποια από αυτές τις περιπτώσεις, έπειτα προχωράμε σε έλεγχο των μεμονωμένων ευθυγράμμων τμημάτων (line segments) του linestring και να εξετάζουμε αν κάποιο τέμνει τις πλευρές του παραθύρου χρησιμοποιώντας για κάθε ευθύγραμμο τμήμα τα σημεία που το ορίζουν στον x και y άξονα, και εάν υπάρχει τομή τότε το αντικείμενο προστίθεται στο αποτέλεσμα και τέλος τυπώνονται τα ζητούμενα στοιχεία.