Διαχείριση Συνθέτων δεδομένων
Βασίλης Γεωργούλας
ΑΜ 2954
Τριτο σετ ασκήσεων.

Μερος 1

Για το πρώτο ερώτημα αρχικά διαβάζουμε το αρχείο nodes.txt και βάζουμε σε μια λίστα για κάθε γραμμή του αρχείου ένα στοιχειό της μορφής [nId, [x, y], [γείτονες]]. Έπειτα διαβάζουμε το αρχείο edges.txt, και για κάθε άκρο της ακμής προσθέτουμε στη λίστα με τους γείτονες του αντίστοιχου κόμβου τον γείτονα και την απόσταση. Δηλαδή μια πλειάδα με στοιχειά (γείτονας, απόσταση ακμής)

Η υλοποίηση βρίσκεται στο αρχείο builder.py και τρέχει με: python3 builder.py και παράγει το αρχείο out.txt.

Μέρος 2

Διαχείριση ουράς προτεραιότητας.

Η υλοποίηση της ουράς προτεραιότητας έγινε όπως αναφέρεται στα docs της python: https://docs.python.org/3/library/heapq.html. Η ιδέα είναι ότι κάθε στοιχειό της ουράς είναι της μορφής [dist, count, nId]. και επιπλέων έχουμε σε ένα λεξικό το στοιχειό nId: [dist, count, nid]. Κάθε φορά που εισάγουμε ένα κόμβο κοιτάμε το λεξικό αν όντως υπάρχει και αν ναι το αφαιρούμε από το λεξικό και μαρκάρουμε το nId 'removed'. Η άλογη αυτή αυτόματα θα συμβεί και στο στοιχειό της ουράς, λόγο αναφοράς. Έπειτα προσθέτουμε ένα νέο αντικείμενο στην ουρά και στο λεξικό. Κατά το pop βγάζουμε στοιχειά από την ουρά μέχρι να βρούμε κάποιο που δεν έχει removed στο τελευταίο στοιχειό. Αν βρούμε ένα τέτοιο τότε το αφαιρούμε και από το λεξικό πριν το επιστρέψουμε.

Dijkstra και Astar.

Για κάθε ένα από τους 2 αλγόριθμους μετράμε την απόσταση, από τον αρχικό τις διαδρομές προς κάθε άλλο κόμβο καθώς και το ελάχιστο κόστος. Επιπλέον έχουμε ένα μετρητή που τον αυξάνουμε όταν επισκεπτόμαστε ένα κόμβο, δηλαδή τον μαρκάρουμε ως visited. Η διάφορα στον Astar είναι ότι προσθέτουμε στην ουρά προτεραιότητας την ελάχιστη απόσταση + τη ευκλείδεια τρεχοντα-τεληκου. Η παράγωγη του γραφήματος γίνεται διαβάζοντας το αρχείο out.txt.

Η υλοποίηση βρίσκεται στο αρχειο part2.py και τρέχει με python3 part2.py sId tId και τυπώνει την απόσταση, τις συνολικές επισκέψεις, το μονοπάτι και το μήκος του μονοπατιού.

Μέρος 3

Η πρώτη ενέργεια που έγινε είναι η μετατροπή του αλγόριθμου Dijkstra σε Incremental. Κάθε φορά που τον συνεχίζουμε από κάποιον αρχικό κόμβο θα μας επιστρέψει τον επόμενο κοντινότερο κόμβο Επιπλέον για τη λειτουργιά δημιουργούμε δυναμικά k ουρές προτεραιότητες, λεξικά, και λίστες για τα

ελάχιστα μονοπάτια, και αποστάσεις. Επισης για καθε id εισοδου κανουμε την αντιστοιχηση σε ενα index αυξητηκα. Με αυτο το τροπο χειριζομαστε και την περιπτωση που ενας αρχηκος κομβος εμφανιζεται πολλες φορες.

Στην συνεχεία έχει υλοποιηθεί μια παραλλαγή του NRA (reversed NRA). Όσο δεν έχουμε βρει το καλύτερο σημείο παίρνουμε για κάθε κόμβο το επόμενο κοντινότερο του, και ενημερώνουμε τα όρια του. Έχουμε μια λίστα οπού για κάθε κόμβο του γραφήματος κρατάμε τα ώρια ως [lb, ub]. Τα αρχηκοποιουμε ως [minDist, 0] όπου mindist = pow(2, 30). Έπειτα για κάθε ένα επόμενο σημείο ενημερώνουμε τα όρια ως έξης: για το κάτω άπλα κοιτάμε αν αυτό που ήρθε είναι μικρότερο, ενώ για το μεγαλύτερο περνούμε τη μεγαλύτερη απόσταση από κάθε αρχικό προς αυτόν τον κόμβο με την max. Αν το μεγαλύτερο είναι μικρότερο από το minDist τότε είναι σημείο συνάντησης και ελέγχεται με το προηγούμενο καλύτερο με βάση τα άνω όρια. Επίσης κρατάμε και σε ένα λεξικό όλα τα σημεία συνάντησης. Κάθε επόμενο κόμβο που λαμβάνουμε τον αποθηκεύουμε σε μια ουρά προτεραιότητας με βάση το κάτω Όριο.

Τέλος για τη συνθήκη τερματισμού, μετά από μια πλήρη επανάληψη του NRA ελέγχουμε αν το μικρότερο κάτω Όριο απόστασης κάποιου κόμβου από τον οποίο η απόσταση από όλους τους χρήστες δεν είναι γνωστή είναι μεγαλύτερο από το πάνω Όριο του καλύτερου σημείου μέχρι τώρα. Τότε θα είμαστε σίγουροι ότι τα άνω όρια θα είναι πάντα μεγαλύτερα από το τρέχον. Για να παρουμε τον κομβο αυτο κοιταμε στην ουρα προτεραιοτητας με τα σημεια μεχρι να βρουμε καποιο που δεν βρισκεται στο λεξικο με τα καλητερα, δλδ εχει επισκευθει απο καποιους κομβους αλλα οχι απο ολλους. Η παράγωγη του γραφήματος γίνεται διαβάζοντας το αρχείο out.txt.

Η υλοποίηση βρίσκεται στο αρχείο NRA.py και τρέχει με python3 NRA.py id1 id2 id.... , και τυπώνει το id του βέλτιστου σημείου, το άνω Όριο του σημείου, και τις αποστάσεις και μονοπάτια από όλους τους αρχικούς προς το σημείο συνάντησης.