

ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ ΑΠΘ

Δομές Δεδομένων 2013-2014

Εργασία 2

Περιγραφή

Στο δεύτερο μέρος της εργασίας, που αποτελεί επέκταση του πρώτου, θα ασχοληθούμε με ουρές προτεραιότητας, κατακερματισμό και γραφήματα. Και πάλι θα πρέπει να μπορείτε να διαβάζετε ένα αρχείο **input.txt** που περιέχει τις συνδέσεις μεταξύ κόμβων όπως και στην πρώτη εργασία. Βασική διαφορά είναι ότι τώρα το γράφημα θεωρούμε ότι είναι μη κατευθυνόμενο και ότι σε κάθε ακμή υπάρχει και ένα βάρος που δηλώνει το κόστος της ακμής. Έτσι, το αρχείο input.txt περιέχει **τριάδες ακεραίων αριθμών** της μορφής $x\ y\ w$, όπου x και y τα IDs των κορυφών και w το βάρος της ακμής. Υποθέστε ότι δε θα υπάρχουν πάνω από 50.000 κορυφές στο αρχείο input.txt.

Η οργάνωση του πίνακα δε θα γίνει με ταξινομημένο array αλλά με **πίνακα κατακερματισμού**. Άρα, για να βρούμε τους γείτονες μίας κορυφής, θα πρέπει να εφαρμόσουμε τη συνάρτηση κατακερματισμού και στη συνέχεια να αναζητούμε την κορυφή σύμφωνα με τη λειτουργία του κατακερματισμού ανοικτής διεύθυνσης. Οι γείτονες της κάθε κορυφής θα είναι αποθηκευμένοι σε μία δομή που υποστηρίζει δυναμική δέσμευση μνήμης (π.χ. απλά συνδεδεμένη λίστα ταξινομημένη σε αύξουσα διάταξη ως προς τα IDs, AVL, κλπ). Για καλούς χρόνους αναζήτησης στον πίνακα κατακερματισμού αν ξέρετε ότι θα αποθηκευτούν N κορυφές, χρησιμοποιήστε μέγεθος πίνακα $2*N$, έτσι ώστε ο παράγοντας φόρτωσης να είναι περίπου 0.5.

Οι λειτουργίες που πρέπει να υποστηρίζετε είναι τρεις: 1) η εύρεση του ελάχιστου δένδρου σύμφωνα με τον αλγόριθμο του Prim (εντολή **MST**), 2) η εύρεση του πλήθους των κοινών γειτόνων δύο κορυφών (εντολή **CN**) και 3) εύρεση ελαχίστων διαδρομών με τον αλγόριθμο του Dijkstra (εντολή **SP**). Οι λειτουργίες θα διαβάζονται από το αρχείο commands.txt, όπου θα υπάρχουν οι εξής εντολές:

```
READ_DATA input.txt // ανάγνωση του αρχείου input.txt
INSERT_LINK x y w    // εισαγωγή ακμής μεταξύ κορυφών x και y με βάρος w
DELETE_LINK x y      // διαγραφή ακμής
MST                  // υπολογισμός του ελαχίστου δένδρου
CN x y               // υπολογισμός πλήθους κοινών γειτόνων των κορυφών x και y
SP x                 // υπολογισμός ελαχίστων διαδρομών από την κορυφή x
```

Στο αρχείο output.txt γράφουν μόνο οι εντολές MST, CN και SP. Κάθε φορά που στο αρχείο input.txt συναντούμε μία εντολή MST, στο output.txt γράφουμε το κόστος του ελάχιστου δένδρου καθώς επίσης και το χρόνο εκτέλεσης της εύρεσης του δένδρου (π.χ. σε seconds). Κάθε φορά που συναντούμε εντολή CN γράφουμε το πλήθος των κοινών γειτόνων. Τέλος, όταν συναντούμε την εντολή SP γράφουμε το κόστος της ελάχιστης διαδρομής από την κορυφή x σε όλες τις άλλες. Σημειώνεται ότι οι εντολές που βρίσκονται στο αρχείο commands.txt μπορεί να είναι με τυχαία σειρά ενώ δεν υπάρχει όριο σχετικά με το πλήθος τους. Ο μόνος περιορισμός είναι ότι η πρώτη εντολή του commands.txt είναι η εντολή READ_DATA.

Σημείωση: η απόδοση των αλγορίθμων Prim και Dijkstra είναι σε εξάρτηση με τις βοηθητικές δομές δεδομένων που χρησιμοποιούν. Χρησιμοποιώντας σωρό ελαχίστων θα μειωθεί σημαντικά η πολυπλοκότητα και ο χρόνος επεξεργασίας σε σχέση με τη χρήση ενός απλού array. **Ο χρόνος επεξεργασίας θα έχει αντίκτυπο στο βαθμό της εργασίας.**

Παραδοτέα. Θα πρέπει να στείλετε με e-mail ένα zip, rar, κλπ. αρχείο που θα περιέχει τον πηγαίο κώδικα και μία έκθεση για το πως υλοποιήσατε τα ζητούμενα.

Απαιτήσεις. Η εργασία θα πρέπει να εκπονηθεί μόνο σε C++, σε **ομάδες των δύο ατόμων**. Η εργασία λαμβάνει το **30% του τελικού βαθμού**.

Προθεσμία. Η εργασία θα πρέπει να παραδοθεί μέχρι τις **20 Φεβρουαρίου 2014**.

Καλή επιτυχία