

Άσκηση 2 [04.05.2020]**Παράδοση: Τετάρτη 26.05.2020, 23:59**

Η παρούσα άσκηση αφορά στην υλοποίηση ενός αλγόριθμου εύρεσης συντομότερων διαδρομών σε ένα κατευθυνόμενο γράφημα. Σκοπός της άσκησης είναι η εξοικείωση με τη C++, την Boost, την LEDA και την πειραματική αξιολόγηση.

Συγκεκριμένα, το πρόβλημα που θα μελετηθεί αφορά στην εύρεση των **συντομότερων διαδρομών** από μια αρχική κορυφή s προς όλες τις άλλες σε ένα δοθέν συνεκτικό κατευθυνόμενο γράφημα $G = (V, E)$ με συνάρτηση κόστους των πλευρών $c : E \rightarrow \mathbb{Z}$. Για τη λύση του προβλήματος ζητείται να υλοποιήσετε τον αλγόριθμο **Bellman-Ford** που περιγράφεται στην Ενότητα 7 «Συντομότερες Διαδρομές II» (διαφάνειες 9-19) χρησιμοποιώντας την βιβλιοθήκη της **Boost**, αφού αποφασίσετε για την αναπαράσταση του $+\infty$ και του nil , και να τον αξιολογήσετε πειραματικά.

Ο συγκεκριμένος αλγόριθμος εκτελείται σε φάσεις. Σε κάθε φάση διατηρεί ένα σύνολο κορυφών U για τις οποίες ισχύει προσωρινά η τριγωνική ανισότητα:

$$U \supseteq \{u : d(u) < \infty \wedge \exists (u, v) \in E, d(u) + c(u, v) < d(v)\}$$

Το σύνολο U διατηρείται σε φάσεις. Αρχικά $U_0 = \{s\}$ και οι αποστάσεις όλων των κορυφών είναι ίσες με $+\infty$. Σε κάθε φάση $i \geq 0$, κάθε κορυφή $u \in U_i$ χαλαρώνει τις εξερχόμενες ακμές προς κάθε γειτονική της κορυφή v και προσθέτει την v στο U_{i+1} . Ο αλγόριθμος ολοκληρώνει το έργο του σε μια φάση $j \leq n$ όταν $U_j = \emptyset$, ή όταν $j = n$, ακολουθούμενος από μια μεταγενέστερη φάση.

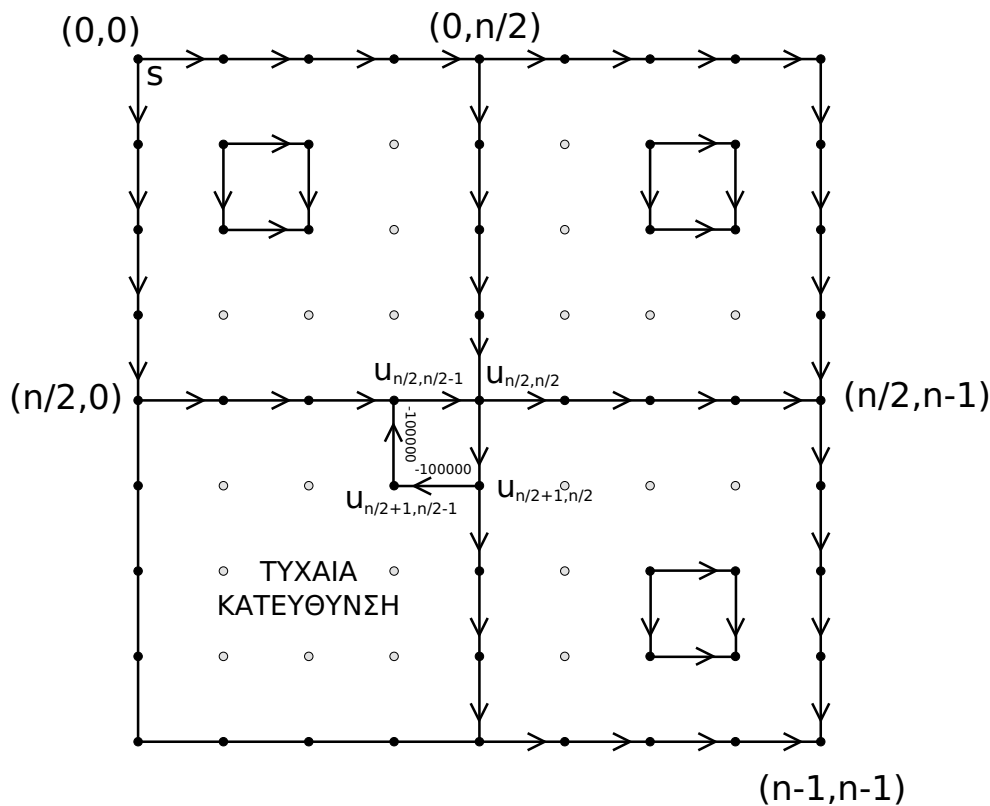
Θα πρέπει να συγκρίνετε τον αλγόριθμό σας με τον ήδη υλοποιημένο στη βιβλιοθήκη LEDA αλγόριθμο `BELLMAN_FORD_B_T` και με τον ήδη υλοποιημένο στη βιβλιοθήκη Boost αλγόριθμο `bellman_ford_shortest_paths()`.

Οι πειραματικές μετρήσεις σας θα εκτελεστούν στις εξής οικογένειες γραφημάτων:

- Τυχαία γραφήματα με μέγεθος κορυφών $n \in \{1000, 4000, 8000\}$ και μέγεθος ακμών $m = 20n \log n$. Ο κόμβος s θα επιλέγεται κάθε φορά τυχαία. Τα κόστη των ακμών θα παίρνουν τυχαίες ακέραιες τιμές στο διάστημα $[-100, 10000]$. Φροντίστε τα παραπάνω γραφήματα να είναι **ασθενώς συνεκτικά** (δηλαδή το υποκείμενο μη-κατευθυνόμενο γράφημα να είναι συνεκτικό) για την ορθή εκτέλεση του αλγόριθμου.
- Γραφήματα τύπου πλέγματος (grid) μεγέθους $n \times n$, όπου $n \in \{100, 200, 300\}$ που κατασκευάζονται ως εξής: Θεωρούμε ότι ο πάνω αριστερά κόμβος έχει συντεταγμένες $(0, 0)$ και ο κάτω δεξιά $(n - 1, n - 1)$.
 - ο Στο τμήμα που ορίζεται από τις συντεταγμένες $(x, y) \in [(0, 0) \times (\frac{n}{2}, n - 1)]$ όλες οι οριζόντιες ακμές έχουν κατεύθυνση από αριστερά προς τα δεξιά, όλες οι κάθετες από πάνω προς τα κάτω, και όλες οι ακμές έχουν τυχαίο κόστος $c \in [0, 10000]$.

- Στο τμήμα που ορίζεται από τις συντεταγμένες $(x, y) \in [(\frac{n}{2}, \frac{n}{2}) \times (n-1, n-1)]$ όλες οι οριζόντιες ακμές έχουν κατεύθυνση από αριστερά προς τα δεξιά, όλες οι κάθετες από πάνω προς τα κάτω, και όλες οι ακμές έχουν τυχαίο κόστος $c \in [0, 10000]$.
- Στο τμήμα που ορίζεται από τις συντεταγμένες $(x, y) \in [(\frac{n}{2}, 0) \times (n-1, \frac{n}{2})]$ όλες οι ακμές παίρνουν τυχαία κατεύθυνση και τυχαίο κόστος $c \in [-100, 10000]$. Επιπλέον οι ακμές $(u_{\frac{n}{2}+1, \frac{n}{2}}, u_{\frac{n}{2}+1, \frac{n}{2}-1})$ και $(u_{\frac{n}{2}+1, \frac{n}{2}-1}, u_{\frac{n}{2}, \frac{n}{2}-1})$ έχουν κόστος -100000 .

Η αρχική κορυφή s θα είναι κορυφή με συντεταγμένες $(0, 0)$. Στο ακόλουθο σχήμα φαίνεται η κατασκευή του γραφήματος.



Για κάθε γράφημα από τα παραπάνω θα πρέπει να εκτελέσετε (τουλάχιστον) πέντε διαφορετικές μετρήσεις και να παρουσιάσετε τον μέσο όρο αυτών, έτσι ώστε να περιοριστεί ο ρόλος των τυχαίων επιλογών.

Στα πειράματα που θα εκτελέσετε πρέπει, εκτός του χρόνου εκτέλεσης, να αναφέρετε εάν έχει βρεθεί αρνητικός κύκλος ή όχι, καθώς και σε ποιο σύνολο από τα V^+ , V^- , V^f ανήκει κάθε κορυφή.

Στην περίπτωση που υπάρχει αρνητικός κύκλος, η σύγκριση των χρόνων εκτέλεσης θα πρέπει να γίνει μεταξύ της υλοποίησής σας και της υλοποίησης της LEDA, χωρίς δηλαδή να συμπεριλάβετε στη σύγκριση και την υλοποίηση της Boost (Γιατί;). Στην περίπτωση που

δεν υπάρχει αρνητικός κύκλος, η σύγκριση πρέπει να γίνει μεταξύ και των τριών υλοποιήσεων. Εξηγήστε τα αποτελέσματα και τυχόν διαφορές που παρατηρείτε στα πειράματα με τα παραπάνω γραφήματα.

Υποβολή Προγραμματιστικής Άσκησης: Η παράδοση της εργασίας θα πραγματοποιηθεί ηλεκτρονικά μέσω της ιστοσελίδας του μαθήματος στο eclass, υποβάλλοντας ένα συμπιεσμένο αρχείο με όνομα hw2_AM_2020, όπου στο AM θα βάλετε τον Αριθμό Μητρώου σας.

Το αρχείο θα περιλαμβάνει όλα τα παραδοτέα της εργασίας:

1. **report.pdf**

Ένα pdf αρχείο που θα περιέχει την αναφορά της εργασίας με τις βασικές αποφάσεις της υλοποίησής σας, τα δεδομένα δοκιμής που χρησιμοποιήσατε και την πειραματική αξιολόγηση σας.

2. **Makefile**

Ένα αρχείο με τις οδηγίες: α) **compile** για τη μεταγλώττιση του πηγαίου κώδικα και β) **run** για την εκτέλεση των μεταγλωτισμένων αρχείων.

3. **src/**

Ένα φάκελο στο οποίο θα υπάρχουν όλα τα αρχεία του πηγαίου κώδικά σας.

4. **README**

Ένα αρχείο που θα περιλαμβάνει τα στοιχεία σας: όνομα, επώνυμο και email.

Παρατήρηση 1: Ο πηγαίος κώδικας που δίνετε για τις υλοποιήσεις και πειραματικές αξιολογήσεις σας πρέπει να είναι σωστά δομημένος, στοιχισμένος και σχολιασμένος. Επίσης ο κώδικάς σας πρέπει να εκτελείται στο σύστημα diogenis.

Παρατήρηση 2: Για περαιτέρω διευκρινήσεις σχετικά με την άσκηση επικοινωνήστε με τον Νίκο Ζαχαράτο (zacharato@ceid.upatras.gr).