Ασκήσεις σε Προσεγγιστικούς/Τυχαιοποιημένους Αλγόριθμους
Καταληκτική Ημερομηνία Παράδοσης: 25/1/2016
Προσθετικός βαθμός: 2 + 4
(με email στο tsichlas@delab.csd.auth.gr)

Παραδοτέο: Αναφορά με λύσεις και παρατηρήσεις + Κώδικας. Οι Α και Β είναι προγραμματιστικές ενώ η Γ και Δ είναι θεωρητικές.

Α) (0.3) Να βρείτε τα πρώτα πέντε δεκαδικά ψηφία του π (3.14159) με την μέθοδο Monte Carlo που είδαμε στη διάλεξη 11. Δείξτε πειραματικά ότι δεν είναι τόσο αποδοτική μέθοδος για να υπολογίζουμε τα ψηφία του π. Αν μπορείτε, βρείτε (ή αποδείξτε μόνοι σας) τις εγγυήσεις μίας τέτοιας μεθόδου.

Β) (0.7) Το συνοδευτικό αρχείο (MinCut.txt) περιέχει την αναπαράσταση με λίστα γειτνίασης ενός απλού μη κατευθυνόμενου γραφήματος. Υπάρχουν 200 κορυφές με ετικέτες από το 1 έως το 200. Η πρώτη στήλη στο αρχείο αντιστοιχεί στην ετικέτα μίας κορυφής ενώ μία γραμμή αντιστοιχεί σε όλες τις κορυφές με τις οποίες συνδέεται ο κόμβος που είναι πρώτος στη γραμμή (1η στήλη). Για παράδειγμα, η 1η γραμμή αναφέρει: «1 37 79 164 155 32 …» το οποίο σημαίνει ότι ο κόμβος 1 συνδέεται με τους κόμβους 37, 79, κοκ.

Θα πρέπει να γράψετε σε C++ τον τυχαιοποιημένο αλγόριθμο εύρεσης καθολικής ελάχιστης αποκοπής και να το χρησιμοποιήσετε στο γράφημα του αρχείου που σας δίνεται. Προσοχή στην υλοποίηση της πράξης συρρίκνωσης μίας ακμής. Αρχικά, θα μπορούσατε να το κάνετε απλά δημιουργώντας ένα καινούργιο γράφημα κάθε φορά που μία ακμή συρρικνώνεται. Αυτό όμως δεν είναι αποδοτικό και θα πρέπει να σκεφτείτε τρόπους να το κάνετε γρήγορα. Όλες οι σχεδιαστικές σας επιλογές θα πρέπει να γραφούν στην αναφορά.

Γ) (0.5) Να δείξετε ότι το ακόλουθο πρόβλημα δεν μπορεί να προσεγγισθεί με λόγο (1+ε) σε πολυωνυμικό χρόνο εκτός και αν P=NP: Δοθέντος ενός γραφήματος G(V,E) με θετικά βάρη στις ακμές και ενός θετικού ακεραίου k, βρείτε ένα υποσύνολο S των κορυφών με πληθάριθμο k, έτσι ώστε το συνολικό βάρος (αθροιστικά) του προκύπτοντος υπογραφήματος του S (που περιλαμβάνει τις ακμές μεταξύ των κόμβων του S και μόνο αυτές) να είναι το ελάχιστο δυνατό.

Δ) (0.5) Έστω το εξής πρόβλημα: μας δίνονται αντικείμενα συγκεκριμένου βάρους το καθένα (δεν έχουν όλα το ίδιο βάρος) και σακιά που μπορούν να σηκώσουν βάρος το πολύ Β (είναι ίδιο για όλα τα σακιά). Ο στόχος μας είναι να τοποθετήσουμε όλα τα αντικείμενα μέσα στα σακιά ώστε κανένα σακί να μην έχει βάρος >Β και με στόχο την ελαχιστοποίηση του πλήθους των σακιών που χρησιμοποιούμε. Έστω ο εξής απλός αλγόριθμος για αυτό το πρόβλημα: Αν ένα αντικείμενο χωρά στο σακί που άνοιξε πιο πρόσφατα τότε το τοποθετούμε εκεί. Διαφορετικά, ανοίγουμε ένα καινούργιο σακί. (i) Δείξτε ότι αυτός ο αλγόριθμος είναι 2-προσεγγιστικός. (ii) Δώστε ένα παράδειγμα χειρότερης περίπτωσης που να δίνει μία 2 προσέγγιση.

<u>(4 μονάδες)</u> (Προαιρετικά – όποιος θέλει – μπορεί να μετατραπεί και σε πτυχιακή ☺) Λύστε το πρόβλημα του Ευκλείδιου TSP (Travelling Salesman Problem) χρησιμοποιώντας τουλάχιστον τέσσερις διαφορετικούς αλγόριθμους (ο ένας θα πρέπει να είναι ακριβής ενώ οι υπόλοιποι προσεγγιστικοί/τυχαιοποιημένοι/natural (ant colony) κτλ.). Μία καλή αρχή για την απαραίτητη βιβλιογραφική αναζήτηση που πρέπει να προηγηθεί για το συγκεκριμένο πρόβλημα θα ήταν να ξεκινήσετε από την Wikipedia. Εδώ μπορείτε να βρείτε ένα μικρό στιγμιότυπο του προβλήματος (συνοδευόμενο αρχείο – tsp.txt). Η πρώτη γραμμή αναφέρεται στο πλήθος των πόλεων ενώ οι υπόλοιπες γραμμές αναφέρονται στις συντεταγμένες τους (*x* και *y* συντεταγμένη).

Έπειτα να λύσετε το πρόβλημα για το αρχείο με τις ελληνικές πόλεις που μπορείτε να βρείτε εδώ. Βεβαίως το να χρησιμοποιήσετε τον ακριβή αλγόριθμο μάλλον είναι αδύνατο. Δώστε τα αποτελέσματά σας τόσο ως προς την ποιότητα της λύσης όσο και ως προς το χρόνο. Περιγράψτε τους αλγορίθμους που επιλέξατε.