Εθνικό Μετσόβιο Πολυτεχνείο
Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών
Τομέας Τεχνολογίας Πληροφορικής και Υπολογιστών
Αλγόριθμοι και Πολυπλοκότητα
Διδάσκων: Δημήτρης Φωτάκης
3η Σειρά Γραπτών Ασκήσεων - Ημ/νία Παράδοσης 8/1/2016

Άσκηση 1: Γέφυρες και Σημεία Κοπής σε Γραμμικό Χρόνο

Έστω $G(V, E)$ ένα απλό συνεκτικό μη-κατευθυνόμενο γράφημα. Μια ακμή e του G αποτελεί *γέφυρα* αν το υπογράφημα του G που προκύπτει από την αφαίρεση της e είναι μη συνεκτικό (ή ισοδύναμα, αν η e δεν ανήκει σε κάποιο κύκλο του G). Μια κορυφή v του G αποτελεί *σημείο κοπής* αν το υπογράφημα του G που προκύπτει από την αφαίρεση της v και όλων των ακμών που προσπίπτουν σε αυτή είναι μη συνεκτικό.

Να διατυπώσετε αλγόριθμο *γραμμικού χρόνου* που υπολογίζει όλες τις γέφυρες και όλα τα σημεία κοπής ενός απλού συνεκτικού μη κατευθυνόμενου γραφήματος $G(V, E)$ που αναπαρίσταται με λίστα γειτνίασης. Να αιτιολογήσετε προσεκτικά την ορθότητα του αλγορίθμου σας.

Υπόδειξη: Θα σας βοηθήσει να μελετήσετε αυτή την άσκηση παράλληλα με τις αντίστοιχες ασκήσεις **22-2** του CLRS και **3.31** του DPV.

Άσκηση 2: Μια Συνάρτηση Κόστους σε Κατευθυνόμενα Γραφήματα (DPV 3.25)

Θεωρούμε ένα κατευθυνόμενο γράφημα $G(V, E)$ στο οποίο κάθε κορυφή $u \in V$ έχει μία θετική τιμή $p(u) \in \mathbb{N}$. Το κόστος $c(u)$ κάθε κορυφής $u \in V$ είναι η τιμή της φθηνότερης κορυφής που είναι προσπελάσιμη από τη u (συμπεριλαμβανομένης και της ίδιας της u). Να διατυπώσετε αλγόριθμο γραμμικού χρόνου που υπολογίζει το κόστος $c(u)$ για όλες τις κορυφές $u \in V$. Να αιτιολογήσετε την ορθότητα των αλγορίθμων στα (α) και (β).

(α) Να διατυπώσετε αλγόριθμο γραμμικού χρόνου για την περίπτωση που το G είναι ένα Κατευθυνόμενο Ακυκλικό Γράφημα (DAG).

(β) Να γενικεύσετε τον αλγόριθμο του (α) ώστε να εφαρμόζεται σε κάθε κατευθυνόμενο γράφημα G. Μπορείτε να χρησιμοποιήσετε ότι οι ισχυρά συνεκτικές συνιστώσες ενός κατευθυνόμενου γραφήματος G μπορούν να υπολογιστούν σε γραμμικό χρόνο (δείτε πως στις ενότητες DPV 3.4 και CLRS 22.5).

Άσκηση 3: Ανάλυση Ασφάλειας (KT 3.11)

Ένα εταιρικό δίκτυο αποτελείται από n υπολογιστές $C_1, \dots C_n$. Έπειτα από μία προσβολή του δικτύου από ιό, θέλουμε να μελετήσουμε την εξάπλωση του ιού στο δίκτυο. Από την επεξεργασία των log files, έχουμε καταλήξει σε m τριάδες της μορφής (C_i, C_j, t), οι οποίες δίνονται σε αύξουσα χρονική σειρά, που δηλώνουν ότι οι υπολογιστές C_i και C_j επικοινώνησαν μεταξύ τους τη χρονική στιγμή t. Γνωρίζουμε ότι αν δύο υπολογιστές C_i και C_j επικοινώνησαν τη χρονική στιγμή t και (μόνον) ο ένας είχε μολυνθεί από τον ιό κάποια στιγμή $t' \leq t$, τότε μολύνθηκε και ο άλλος τη χρονική στιγμή t. Γνωρίζουμε ότι ο ιός εισήλθε στο δίκτυο από τον υπολογιστή C_1, ο οποίος μολύνθηκε τη χρονική στιγμή $t_1 = 0$. Για κάθε υπολογιστή C_j, θέλουμε να υπολογίσουμε τη χρονική στιγμή t_j κατά την οποία μολύνθηκε. Να διατυπώσετε έναν αλγόριθμο γραμμικού χρόνου για αυτό το πρόβλημα και να αιτιολογήσετε την ορθότητα και την υπολογιστική του πολυπλοκότητα. *Παράδειγμα:* Αν έχουμε $n = 4$ υπολογιστές και $m = 4$ τριάδες, τις $(C_1, C_2, 2)$, $(C_2, C_3, 6)$, $(C_3, C_4, 6)$ και $(C_1, C_4, 10)$, οι χρόνοι μόλυνσης είναι $t_1 = 0$, $t_2 = 2$, $t_3 = 6$ και $t_4 = 6$ (ο C_4 μολύνεται μέσω του C_3 τη χρονική στιγμή 6).

Άσκηση 4: Το Σύνολο των Συνδετικών Δέντρων (ΚΤ 4.27 και ΚΤ 4.28)

Θεωρούμε ένα συνεκτικό μη κατευθυνόμενο γράφημα $G(V, E)$ με n κορυφές και m ακμές.

(α) Έστω T_1 και T_2 δύο διαφορετικά συνδετικά δέντρα του G. Να δείξετε ότι για κάθε ακμή $e \in T_1 \setminus T_2$, υπάρχει ακμή $e' \in T_2 \setminus T_1$, τέτοια ώστε το $(T_1 \setminus \{e\}) \cup \{e'\}$ είναι συνδετικό δέντρο. Να διατυπώσετε αποδοτικό αλγόριθμο που με δεδομένα τα T_1, T_2 και e, υπολογίζει μια τέτοια ακμή e'.

(β) Σχηματίζουμε γράφημα H που κάθε κορυφή του αντιστοιχεί σε ένα διαφορετικό συνδετικό δέντρο του G. Δύο συνδετικά δέντρα T_1 και T_2 του G (κορυφές του H) συνδέονται με ακμή στο H αν διαφέρουν κατά μία μόνο ακμή, δηλ. αν $|T_1 \setminus T_2| = |T_2 \setminus T_1| = 1$. Να δείξετε ότι το H είναι συνεκτικό και ότι η απόσταση (στο H) μεταξύ δύο συνδετικών δέντρων T_1 και T_2 του G είναι ίση με $|T_1 \setminus T_2|$. Να εξηγήσετε πως θα χρησιμοποιήσουμε τον αλγόριθμο του (α) για να υπολογίσουμε ένα συντομότερο μονοπάτι (στο H) μεταξύ των T_1 και T_2.

(γ) Θεωρούμε μια διαμέριση των ακμών του G σε δύο σύνολα E_1 και E_2. Να διατυπώσετε αποδοτικό αλγόριθμο που με είσοδο το $G(V, E)$, τα σύνολα E_1 και E_2, και έναν φυσικό k, $1 \leq k \leq n - 1$, υπολογίζει ένα συνδετικό δέντρο του G με ακριβώς k ακμές στο E_1. Αν δεν υπάρχει τέτοιο δέντρο, ο αλγόριθμός σας θα πρέπει να το διαπιστώνει. *Υπόδειξη:* Μπορεί να σας βοηθήσει το (β).

Άσκηση 5: Διαχωρισμός Γραφήματος

Έστω συνεκτικό μη κατευθυνόμενο γράφημα $G(V, E, w)$, όπου κάθε ακμή e έχει βάρος $w(e) > 0$. Για κάθε ζευγάρι κορυφών $u, v \in V$, συμβολίζουμε με $d(u, v)$ την *απόσταση* μεταξύ των u και v στο G. Δηλαδή, $d(u, v)$ είναι το μήκος του συντομότερου $u - v$ μονοπατιού στο G (τα μήκη των μονοπατιών υπολογίζονται λαμβάνοντας υπόψη τα βάρη w των ακμών).

(α) Έστω T ένα Ελάχιστο Συνδετικό Δέντρο του G. Να δείξετε ότι για κάθε ακμή $e = \{u, v\} \in T$, $d(u, v) = w(e)$ (δηλ. η e αποτελεί ένα συντομότερο $u - v$ μονοπάτι).

(β) Για μια διαμέριση του V σε δύο υποσύνολα S_1 και S_2, ορίζουμε ως απόσταση $d(S_1, S_2)$ την ελάχιστη απόσταση μεταξύ μιας κορυφής του S_1 και μιας κορυφής του S_2, δηλ. $d(S_1, S_2) = \min_{u \in S_1, v \in S_2} d(u, v)$. Να διατυπώσετε έναν αποδοτικό αλγόριθμο που υπολογίζει μια διαμέριση του V σε δύο υποσύνολα S_1 και S_2 που μεγιστοποιούν την απόσταση $d(S_1, S_2)$. Να αιτιολογήσετε αναλυτικά την ορθότητα και την υπολογιστική πολυπλοκότητα του αλγορίθμου σας. *Υπόδειξη:* Δεν είναι απαραίτητο να υπολογίσετε αποστάσεις κορυφών στο G!