

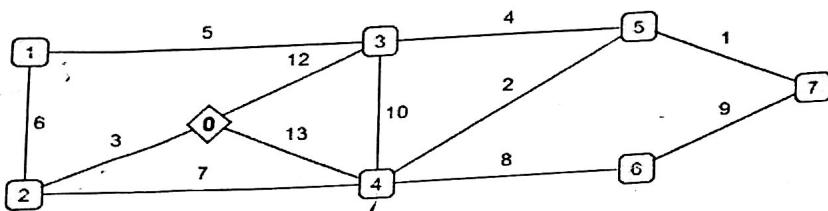
Αλγόριθμοι και Πολυπλοκότητα

15 Φεβρουαρίου 2017

- Διάρκεια: 2 ½ ώρες.
- Καλή επιτυχία.

Θέμα 1°

Χρησιμοποιώντας τον αλγόριθμο του Prim, να υπολογίσετε ένα ελάχιστο διασυνδετικό δένδρο (minimum spanning tree) για το παρακάτω γράφημα. Ως ρίζα του ελάχιστου διασυνδετικού δένδρου να θεωρήσετε την κορυφή «0». Να παρουσιάσετε την δουλειά σας λεπτομερειακά.



Θέμα 2°

Έστω ένα ταξινομημένο διάνυσμα A από ακεραίους το οποίο έχει «κυκλικά περιστραφέν» k θέσεις προς τα δεξιά. Για παράδειγμα, το διάνυσμα $[35, 42, 5, 15, 27, 29]$ είναι ένα ταξινομημένο διάνυσμα που έχει περιστραφεί 2 θέσεις προς τα δεξιά, ενώ το διάνυσμα $[27, 29, 35, 42, 5, 15]$ έχει περιστραφεί κατά 4 θέσεις.

Να σχεδιάσετε έναν $O(\log n)$ αλγόριθμο ο οποίος δεδομένου ενός ταξινομημένου διανύσματος το οποίο έχει κυκλικά περιστραφέν, υπολογίζει το μέγεθος k της περιστροφής. Να επιχειρηματολογήσετε ως προς την ορθότητα του αλγορίθμου σας και την πολυπλοκότητα του.

Θέμα 3°

Έστω διάνυσμα A με n στοιχεία. Ένα στοιχείο του A καλείται πλειοψηφικό εάν εμφανίζεται περισσότερες από $n/2$ φορές. Σχεδιάστε και αναλύστε έναν αλγόριθμο τύπου "διάρει και βασίλευε" ο οποίος θα εντοπίζει το πλειοψηφικό στοιχείο (εάν υπάρχει) σε χρόνο $n \log n$. Δεν επιτρέπεται να μεταχειριστείτε αλγόριθμο ταξινόμησης, καθώς θεωρούμε ότι διαθέσιμη είναι μόνο η πράξη του ελέγχου ισότητας μεταξύ στοιχείων του διανύσματος. Να επιχειρηματολογήσετε ως προς την ορθότητα του αλγορίθμου σας.

Θέμα 4°

Μία γλώσσα επεξεργασίας αλφαριθμητικών (strings) περιλαμβάνει την λειτουργία της «διάσπασης» ενός αλφαριθμητικού σε 2 νέα τμήματα. Επειδή η λειτουργία εμπεριέχει την αντιγραφή του αρχικού αλφαριθμητικού, η διάσπαση ενός αλφαριθμητικού n χαρακτήρων σε δύο νέα αλφαριθμητικά κοστίζει n μονάδες χρόνου. Έστω ότι ένας προγραμματιστής θέλει να σπάσει ένα αλφαριθμητικό σε πολλά τμήματα. Η σειρά με την οποία γίνονται τα σπασίματα μπορεί να επηρεάσει το συνολικό απαιτούμενο χρόνο. Για παράδειγμα, έστω ότι θέλουμε να σπάσουμε ένα αλφαριθμητικό 20 χαρακτήρων μετά τους χαρακτήρες στις θέσεις 3, 8 και 10, δημιουργώντας έτσι 4 νέα αλφαριθμητικά. Εάν η διάσπαση γίνει με την από «αριστερά προς τα δεξιά» σειρά των θέσεων διάσπασης, τότε θα χρειαστούν 20 μονάδες χρόνου για την πρώτη διάσπαση, 17 μονάδες χρόνου για την δεύτερη και 12 μονάδες χρόνου για την τρίτη, δηλαδή συνολικά 49 μονάδες χρόνου. Εάν η διάσπαση γίνει με την από «δεξιά προς τα αριστερά» σειρά των θέσεων διάσπασης, μονάδες χρόνου. Εάν η διάσπαση γίνει με την από «δεξιά προς τα αριστερά» σειρά των θέσεων διάσπασης, τότε θα χρειαστούν 20 μονάδες χρόνου για την πρώτη διάσπαση, 10 μονάδες χρόνου για την δεύτερη και 8 μονάδες χρόνου για την τρίτη, δηλαδή συνολικά 38 μονάδες χρόνου.

Σχεδιάστε έναν αλγόριθμο δυναμικού προγραμματισμού ο οποίος, με δεδομένη μία λίστα από τις επιθυμητές θέσεις διάσπασης ενός αλφαριθμητικού n χαρακτήρων, υπολογίζει την βέλτιστη σειρά διάσπασης σε $O(n^2)$ μονάδες χρόνου. (Σχέδιο)

Θέμα 5°

Να δειχθεί ότι το πρόβλημα του υπολογισμού ενός βέλτιστου Huffman κώδικα για ένα αλφάριθμο n χαρακτήρων, $n > 2$, απαιτεί $\Omega(n \log n)$ χρόνο.