

Εθνικό Μετσόβιο Πολυτεχνείο

Σχολή Εφαρμοσμένων Μαθηματικών και Φυσικών Επιστήμων
Τομέας ΜαθηματικώνΔομές Δεδομένων
7 Ιούλιου 2009

- Να απαντηθούν ΟΛΑ (5) τα θέματα.
- Διάρκεια: 2 ½ ώρες.
- Καλή επιτυχία.

Θέμα 1^ο

Έστω 12 στοιχεία με κλειδιά τους αριθμούς:

63, 30, 36, 31, 12, 50, 35, 5, 27, 59, 43, 17

Να τα εισάγετε διαδοχικά, με την ανωτέρω σειρά, σε ένα δυαδικό δένδρο αναζήτησης και, κατόπιν, να διαγράψετε τα στοιχεία με κλειδί 36 και 43. Να περιγραφεί το δυαδικό δένδρο αναζήτησης που προκύπτει μετά την εισαγωγή/διαγραφή κάθε στοιχείου.

Θέμα 2^ο

Ένα διάνυσμα A, η θέσεων καλείται κυκλικό εάν το πρώτο στοιχείο A[0] θεωρείται ότι έπειτα του τελευταίου στοιχείου A[n-1]. Υλοποιήστε τον ΑΤΔ της FIFO χρησιμοποιώντας ως αποθηκευτική δομή ένα κυκλικό διάνυσμα. Όλες οι πράξεις της FIFO να έχουν O(1) κόστος.

Θέμα 3^ο

Έστω ένα δυαδικό δένδρο T του οποίου ο κάθε κόμβος περιέχει ως κλειδί έναν ακέραιο αριθμό.

- Α. Να γραφεί αναδρομική μέθοδος η οποία, δοθείστης της ρίζας ενός δυαδικού δένδρου T παράγει ένα κλώνο του (ακριβές αντίγραφο) T'. Να αναλυθεί η χρονική πολυπλοκότητα.
- Β. Να γραφεί αναδρομικός ψευδό-κώδικας $\text{max}(v)$ για τον υπολογισμό του μέγιστου κλειδιού των κόμβων του υπο-δένδρου το οποίο έχει τον κόμβο v ως ρίζα (συνυπολογίζοντας το κλειδί του κόμβου v). Να αναλυθεί η χρονική πολυπλοκότητα.

Θέμα 4^οΜε δεδομένο ως είσοδο την pre-order σ_{pre} και την post-order σ_{post} διάταξη των κόμβων ενός δένδρου T, να απαντήσετε στα παρακάτω ερωτήματα:

- (a) Είναι το δένδρο T μοναδικό; Δηλαδή, υπάρχει ένα μόνο δένδρο T τέτοιο ώστε $\text{pre-order}(T) = \sigma_{\text{pre}}$ και $\text{post-order}(T) = \sigma_{\text{post}}$;
- (b) Να προτείνετε και να αναλύσετε έναν αλγόριθμο ο οποίος κατασκευάζει ένα δυαδικό δένδρο T, συμβατό με τις διατάξεις σpre και σpost.

Θέμα 5^ο

Να προτείνεται μία υλοποίηση η οποία υποστηρίζει τον αφηρημένο τύπο δεδομένων Min-Max_Priority Queue, δηλ. μία ουρά προτεραιότητας η οποία ταυτόχρονα υποστηρίζει τις λειτουργίες: findMIN, deleteMIN, findMAX, deleteMAX, insert, isEmpty, size. Ζητούμενη είναι $O(\log n)$ πολυπλοκότητα χειρότερης περίπτωσης για κάθε λειτουργία. Να περιγράψετε και να αναλύσετε την πολυπλοκότητα κάθε λειτουργίας.