

HY240: Δομές Δεδομένων
Χειμερινό Εξάμηνο – Ακαδημαϊκό Έτος 2008-09
Παναγιώτα Φατούρου

4^ο Σετ Ασκήσεων

Ημερομηνία Παράδοσης: Παρασκευή, 28 Νοεμβρίου 2008

Τρόπος Παράδοσης: Στο εργαστήριο των μεταπτυχιακών στο υπόγειο των λευκών κτιρίων, ώρα 12:00-13:00. Ασκήσεις που παραδίδονται μετά τις 13:00 θεωρούνται εκπρόθεσμες. Οι φοιτητές μπορούν να παραδίδουν τις ασκήσεις τους και νωρίτερα από αυτή την ημερομηνία αλλά δικαιολογίες όπως «ήρθα την τάδε μέρα και δεν ήσασταν στο γραφείο σας» δεν θα γίνουν δεκτές. Όσοι από τους φοιτητές δεν έχουν καταφέρει να παραδώσουν τις ασκήσεις τους πριν τις 28/11 θα πρέπει να τις παραδώσουν στο εργαστήριο των μεταπτυχιακών την προκαθορισμένη μέρα και ώρα. Οι φοιτητές μπορούν να παραδίδουν τις ασκήσεις τους ηλεκτρονικά αλλά σε αυτή την περίπτωση απαιτείται να γίνουν scan οι λύσεις και είτε να χρησιμοποιηθεί το πρόγραμμα submit (σύμφωνα με τις οδηγίες που στάλθηκαν από τους βοηθούς του μαθήματος) ή να σταλούν μέσω ηλεκτρονικού ταχυδρομείου στο λογαριασμό hy240@csd.uoc.gr. Αυτός είναι και ο τρόπος παράδοσης εκπρόθεσμων ασκήσεων.

1. α. Έστω τα στοιχεία με κλειδιά τους εξής αριθμούς: 63, 30, 36, 31, 12, 50, 35, 5, 27, 59, 43, 17. Να εισαχθούν αυτά διαδοχικά, με τη σειρά που εμφανίζονται παραπάνω, σε ένα αρχικώς άδειο δένδρο AVL.

- β. Να πραγματοποιηθεί διαδοχική διαγραφή των στοιχείων με κλειδιά 36 και 43 από το δένδρο που προκύπτει μετά τις εισαγωγές του ερωτήματος α.

Σημείωση: Θα πρέπει να παρουσιαστούν όλα τα δένδρα που προκύπτουν μετά από κάθε εισαγωγή ή διαγραφή (καθώς και τα ενδιάμεσα στάδια των περιστροφών που απαιτούνται προκειμένου να προκύψουν αυτά).

2. α. Να παρουσιαστούν τα 2-3 δένδρα που προκύπτουν από τη διαδοχική εισαγωγή, με τη σειρά που εμφανίζονται στη συνέχεια, των στοιχείων με κλειδιά 150, 90, 85, 70, 63, 55, 42, 33, 21, 14 σε ένα αρχικά άδειο 2-3 δένδρο.

- β. Να παρουσιαστούν τα δένδρα που προκύπτουν από τη διαδοχική διαγραφή των στοιχείων με κλειδιά 70,63,55 από το δένδρο του ερωτήματος α.

Σημείωση: Θα πρέπει να παρουσιαστούν όλα τα δένδρα που προκύπτουν μετά από κάθε εισαγωγή ή διαγραφή (καθώς και η διαδικασία που ακολουθείται προκειμένου να προκύψουν αυτά).

3. α. Να παρουσιαστούν τα δένδρα που προκύπτουν από τη διαδοχική εισαγωγή των στοιχείων με κλειδιά 41,38,31,12,19,8 σε ένα αρχικά άδειο κόκκινο-μαύρο δένδρο.

- β. Να παρουσιαστούν τα δένδρα που προκύπτουν από τη διαδοχική διαγραφή των κλειδιών 8, 12, 19, 31 και 38 (με αυτή τη σειρά) από το δένδρο που προκύπτει μετά από τις εισαγωγές του ερωτήματος α.

Σημείωση: Θα πρέπει να παρουσιαστούν όλα τα δένδρα που προκύπτουν μετά από κάθε εισαγωγή ή διαγραφή (καθώς και η διαδικασία που ακολουθείται προκειμένου να προκύψουν αυτά).

4. Έστω ότι για κάθε κόκκινο μαύρο δένδρο T διατηρούμε μια μεταβλητή bh στην οποία αποθηκεύουμε το μαύρο ύψος του δένδρου. Για παράδειγμα, το δένδρο αναπαρίσταται ως

ένα struct με πεδία ένα δείκτη r στη ρίζα του δένδρου και έναν ακέραιο bh στον οποίο αποθηκεύουμε το μαύρο ύψος του δένδρου. [15%]

- α. Να αποδειχθεί ότι το bh του T μπορεί να υπολογίζεται σωστά από τις διαδικασίες εισαγωγής ή διαγραφής κόμβου στο T , χωρίς να απαιτείται επιπλέον αποθηκευτικός χώρος στους κόμβους του δένδρου και χωρίς να επηρεάζεται η ασυμπτωτική χρονική πολυπλοκότητα των διαδικασιών αυτών. Αποδείξτε επίσης ότι καθώς διατρέχουμε το δένδρο T καθοδικά, από τη ρίζα προς οποιοδήποτε φύλλο, το μαύρο ύψος κάθε κόμβου του μονοπατιού που διασχίζουμε μπορεί να υπολογιστεί σε $O(1)$ χρόνο ανά προσπελαζόμενο κόμβο (έτσι ώστε τα μαύρα ύψη όλων των κόμβων του μονοπατιού να μπορούν να υπολογιστούν σε λογαριθμικό χρόνο), δεδομένου ότι το bh του T είναι γνωστό.
- β. Έστω ένα κόκκινο-μαύρο δένδρο T με μαύρο ύψος bh και έστω n ο αριθμός των κόμβων του. Έστω k ένας ακέραιος τέτοιος ώστε $bh > k$. Περιγράψτε αλγόριθμο με χρονική πολυπλοκότητα $O(\log n)$ που βρίσκει το μαύρο κόμβο με το μεγαλύτερο κλειδί μεταξύ εκείνων με μαύρο ύψος k στο T . Το μαύρο ύψος ενός κόμβου v είναι ο αριθμός των μαύρων κόμβων σε οποιοδήποτε μονοπάτι από τον v προς οποιοδήποτε φύλλο (μη συμπεριλαμβανομένου του v).
5. Έστω ότι τα S και T είναι σύνολα μεγέθους n και m , αντίστοιχα (δηλαδή, $|S| = n$ και $|T| = m$). Διαλέξτε την κατάλληλη δομή δεδομένων για την υλοποίηση των S και T , έτσι ώστε να υπάρχει αλγόριθμος που να υλοποιεί τη λειτουργία $\text{Intersection}(S, T)$ (η οποία επιστρέφει το σύνολο $S \cap T$), σε χρόνο $O((m+n) \log m)$. Περιγράψτε τον αλγόριθμο που σκεφτήκατε και εξηγήστε γιατί είναι σωστός και γιατί επιτυγχάνει τη ζητούμενη πολυπλοκότητα.