

HY240: Δομές Δεδομένων
Εαρινό Εξάμηνο – Ακαδημαϊκό Έτος 2014-15
Παναγιώτα Φατούρου

3^ο Σετ Ασκήσεων

Ημερομηνία Παράδοσης: Δευτέρα, 27 Απριλίου 2015

Τρόπος Παράδοσης: Οι ασκήσεις μπορούν να παραδοθούν είτε σε ηλεκτρονική μορφή χρησιμοποιώντας το πρόγραμμα submit (δείτε οδηγίες στην ιστοσελίδα του μαθήματος), είτε στους βοηθούς του μαθήματος στο εργαστήριο των μεταπτυχιακών, την Δευτέρα, 27 Απριλίου 2015, ώρα 15:00-16:00. Ασκήσεις που παραδίδονται μετά τις 16:00 της Δευτέρας, 27/5/2015 θεωρούνται εκπρόθεσμες. Εκπρόθεσμες ασκήσεις γίνονται δεκτές σε ηλεκτρονική μορφή και η αποστολή τους πρέπει να γίνει χρησιμοποιώντας το πρόγραμμα submit. Εναλλακτικά, εκπρόθεσμες ασκήσεις μπορούν να παραδοθούν στη διδάσκουσα ή να σταλούν στο hy240b@csd.uoc.gr μέσω ηλεκτρονικού ταχυδρομείου.

Άσκηση 1 [20 Μονάδες]

Παρουσιάστε ψευδοκώδικα για μια συνάρτηση η οποία θα παίρνει, ως όρισμα, έναν δείκτη στη ρίζα ενός δυαδικού δένδρου που υλοποιεί (ένα όχι-απαραίτητα δυαδικό) διατεταγμένο δένδρο και έναν ακέραιο k . Η συνάρτηση θα πρέπει να επιστρέφει TRUE αν το δένδρο είναι γεμάτο k -αδικό δένδρο (δηλαδή αν κάθε κόμβος του διατεταγμένου δένδρου έχει ακριβώς k παιδιά).

Άσκηση 2 [60 μονάδες]

α. Δίνεται μια σταθερά MAX και ένα ταξινομημένο δυαδικό δένδρο (binary search tree) T , στους κόμβους του οποίου αποθηκεύονται κλειδιά που είναι θετικοί ακέραιοι. Ζητείται να παρουσιάσετε ψευδοκώδικα για έναν αλγόριθμο ο οποίος θα λειτουργεί ως εξής. Ο αλγόριθμος θα δέχεται, ως ορίσματα, έναν δείκτη στη ρίζα του T και τη σταθερά MAX και θα δημιουργεί ένα σύνολο από ταξινομημένα δυαδικά δένδρα, τέτοια ώστε, το πρώτο δένδρο T_1 θα περιέχει εκείνα τα κλειδιά του T με τιμές από 0 μέχρι $MAX - 1$, το δεύτερο δένδρο T_2 θα περιέχει εκείνα τα κλειδιά του T με τιμές από MAX μέχρι $2*MAX - 1$ και γενικότερα το i -οστό δένδρο T_i θα περιέχει εκείνα τα κλειδιά του T με τιμές από $(i-1)*MAX$ μέχρι $i*MAX - 1$. Ο αλγόριθμος θα χρησιμοποιεί έναν πίνακα $R[1 \dots \infty]$ (ο οποίος θα είναι global και θα έχει άπειρο μέγεθος). Ο αλγόριθμος θα πρέπει, για κάθε i , $1 \leq i \leq m$, να τοποθετεί στη θέση i του πίνακα έναν δείκτη στη ρίζα του δένδρου T_i και θα πρέπει να επιστρέφει τον αριθμό m των δένδρων που θα προκύψουν από τη διάσπαση του αρχικού δένδρου. Ο αλγόριθμος θα πρέπει να έχει χρονική πολυπλοκότητα $O(h*m)$, όπου h είναι το ύψος του αρχικού δένδρου T .

Υπόδειξη: Προκειμένου να φτιάξετε το δένδρο T_1 , θα πρέπει να εκτελέσετε αναζήτηση για το κλειδί MAX, αποκόβοντας τους δείκτες που βρίσκονται αριστερά και δεξιά του μονοπατιού. Κατόπιν, συγχωνεύστε τα δένδρα του προκύπτοντος δάσους, ώστε να προκύψουν δύο δένδρα, το T_1 με κλειδιά μικρότερα του MAX και το T' με κλειδιά μεγαλύτερα ή ίσα του MAX. Επαναλάβετε την παραπάνω διαδικασία, χρησιμοποιώντας το T' , για να παράγετε τα υπόλοιπα δένδρα.

β. Δίνεται ένας πίνακας $R[1 \dots \infty]$ τέτοιος ώστε, για κάθε j , $1 \leq j \leq k$, το στοιχείο $R[j]$ περιέχει έναν δείκτη σε ένα ταξινομημένο δυαδικό δένδρο (binary search tree) T_j , ενώ για κάθε $j > k$, $R[j] = \text{NULL}$. Ζητείται να παρουσιάσετε ψευδοκώδικα για μια συνάρτηση, η οποία θα δέχεται, ως όρισμα, τον πίνακα R και θα δημιουργεί μια ταξινομημένη λίστα L , η οποία θα περιέχει όλα τα κλειδιά των δένδρων $T_1, \dots, T_k$. Ο αλγόριθμός σας θα πρέπει να έχει χρονική πολυπλοκότητα $O(n_1 + \dots + n_k)$, όπου για κάθε j , $1 \leq j \leq k$, n_j είναι το πλήθος των στοιχείων του δένδρου T_j . Σημειώνεται πως το k δεν είναι γνωστό στον αλγόριθμο που ζητείται να σχεδιάσετε.

Υπόδειξη: Θα πρέπει πρώτα να παρουσιάσετε αλγόριθμο που θα βρίσκει το k . Στη συνέχεια, θα πρέπει να δημιουργήσετε μια αναδρομική διαδικασία η οποία θα σαρώνει τα k δένδρα και θα παράγει το ενιαίο δένδρο. Η διαδικασία αυτή θα παίρνει, ως όρισμα, τον πίνακα R και το k και θα κάνει αναδρομική διάσχιση όλων των δένδρων ταυτόχρονα ώστε να δημιουργήσει τη ζητούμενη λίστα με ένα μόνο πέρασμα από κάθε δένδρο.

Άσκηση 3 [20 μονάδες]

Παρουσιάστε ψευδοκώδικα για τη λειτουργία $\text{Insert}(\text{NODE } *R, \text{int } K)$ η οποία εισάγει ένα στοιχείο με κλειδί K στο δένδρο T του οποίου τη ρίζα δείχνει ο δείκτης R , **έτσι ώστε το T να είναι πλήρες, απλά-συνδεδεμένο, μη-ταξινομημένο δένδρο, το οποίο υλοποιείται με δυναμικό τρόπο** (και όχι με πίνακα, όπως συζητήθηκε στις διαφάνειες).