

Η/Υ 432: Δομές Δεδομένων Χειμερινό Εξάμηνο Ακ. Έτους 2005-2006 Παναγιώτα Φατούρου

3^ο Σετ Ασκήσεων

Ημερομηνία Παράδοσης

Θεωρητικό Μέρος: Μέρος 1: Τρίτη, 20 Δεκεμβρίου 2005, ώρα 12:00, Μέρος 2: Τρίτη, 10 Ιανουαρίου 2005, ώρα 12:00.

Προγραμματιστικό Μέρος: Πέμπτη, 22 Δεκεμβρίου 2005, ώρα 23:59

Τρόπος Παράδοσης:

Θεωρητικό Μέρος: Θα παραδώσετε τις θεωρητικές ασκήσεις στους βοηθούς του μαθήματος στο γραφείο Α25.

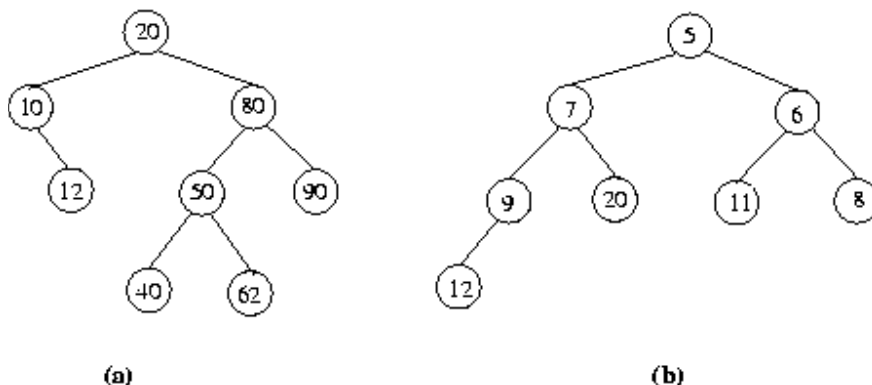
Προγραμματιστικό Μέρος: Θα χρησιμοποιήσετε το πρόγραμμα `turnin`. Η εντολή που θα εκτελέσετε είναι:

`turnin lab3_05@cs432 <όνομα_αρχείου1.c> <όνομα_αρχείου2.c> ... <όνομα_αρχείουk.c>`

Θεωρητικό Μέρος

Μέρος 1

1. α. Εισάγεται το κλειδί 64 στο AVL δένδρο του Σχήματος 1α. Εισάγεται το κλειδί 14 στο δένδρο που προκύπτει. Διαγράψτε το κλειδί 12 από το αρχικό δένδρο του Σχήματος 1α. [5%]



Σχήμα 1

β. Σας δίνεται ένα splay δένδρο τέτοιο ώστε το μονοπάτι από τη ρίζα στο κλειδί 10 περνά από τους κόμβους 30, 20, 25. Δείξτε το αποτέλεσμα μιας λειτουργίας splay γύρω από το 10. [5%]

γ. Παρουσιάστε τα 2-3 δένδρα που προκύπτουν από τις εισαγωγές των κλειδιών 150, 90, 85, 70, 63, 55, 42, 33, 21, 14 σε ένα αρχικά άδειο 2-3 δένδρο. [5%]

δ. Παρουσιάστε τα δένδρα που προκύπτουν από τις διαγραφές των κλειδιών 70, 63, 55 από το δένδρο του ερωτήματος α [5%]

ε. Παρουσιάστε τα δένδρα που προκύπτουν από τις εισαγωγές των κλειδιών 70, 50, 49, 67, 53, 90, 52, 80, 85, 88 σε ένα αρχικά άδειο κόκκινο-μαύρο δένδρο. [5%]

Σημείωση: Θα πρέπει να παρουσιάσετε όλα τα ενδιάμεσα δένδρα που προκύπτουν προκειμένου να γίνουν οι λειτουργίες που περιγράφονται στα ερωτήματα.

2. Σχεδιάστε εκ νέου τις διαδικασίες εισαγωγής και διαγραφής σε ένα 2-3 δένδρο, ώστε να διενεργούνται από πάνω προς τα κάτω (και όχι από κάτω προς τα πάνω όπως παρουσιάστηκαν στο μάθημα). [15%]

Υπόδειξη: Προσπαθήστε, καθώς κατεβαίνετε στο δένδρο, να εξαλείψετε «προβληματικούς» κόμβους. Για την εισαγωγή, τέτοιου είδους κόμβοι είναι 3-κόμβοι επί του μονοπατιού αναζήτησης. Για τη διαγραφή, τέτοιοι κόμβοι είναι οι 2-κόμβοι επί του μονοπατιού αναζήτησης.

3. Αποδείξτε ότι ένα κόκκινο μαύρο δένδρο με n εσωτερικούς κόμβους έχει ύψος το πολύ $2\log(n+1)$. [10%]

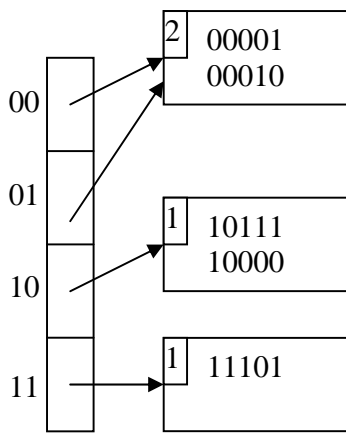
4. Έστω ότι για κάθε κόκκινο μαύρο δένδρο T διατηρούμε μια μεταβλητή bh στην οποία αποθηκεύουμε το μαύρο ύψος του δένδρου. Για παράδειγμα, το δένδρο αναπαρίσταται ως ένα struct με πεδία ένα δείκτη r στη ρίζα του δένδρου και έναν ακέραιο bh στον οποίο αποθηκεύουμε το μαύρο ύψος του δένδρου. [15%]

α. Αποδείξτε ότι το bh μπορεί να υπολογίζεται σωστά κατά την εισαγωγή ή διαγραφή κόμβου στο T χωρίς να απαιτείται επιπλέον μνήμη και χωρίς να επηρεάζεται η χρονική πολυπλοκότητα των αλγορίθμων αυτών (εισαγωγής και διαγραφής). Αποδείξτε επίσης ότι καθώς κατεβαίνουμε από τη ρίζα προς οποιοδήποτε φύλλο, το μαύρο ύψος κάθε κόμβου του μονοπατιού που διασχίζουμε μπορεί να υπολογιστεί σε $O(1)$ χρόνο (έτσι ώστε τα μαύρα ύψη όλων των κόμβων του μονοπατιού να μπορούν να υπολογιστούν σε λογαριθμικό χρόνο).

β. Έστω ένα κόκκινο-μαύρο δένδρο T με μαύρο ύψος bh και έστω n ο αριθμός των κόμβων του. Έστω k ένας ακέραιος τ.ω. $bh > k$. 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).

Μέρος 2

1. α. Εισάγεται το κλειδί 1 στο σωρό του Σχήματος 1β. Εφαρμόστε την `delete_min()` στο σωρό του Σχήματος 1β. [5%]
- β. Εισάγεται το κλειδί με τιμή κατακερματισμού 00100 στον επεκτάσιμο πίνακα κατακερματισμού (extendible hash table) του Σχήματος 2. Διαγράψτε το κλειδί με τιμή κατακερματισμού 11101 από τον πίνακα που προκύπτει. Θεωρείστε ότι κάθε σελίδα-φύλλο του πίνακα χωράει 2 εγγραφές και ότι χρησιμοποιείται μια στρατηγική διαγραφής σύμφωνα με την οποία θα πρέπει το directory (κατάλογος) να διατηρείται όσο το δυνατό μικρότερο. [5%]
- γ. Ξεκινώντας από τα εξής 7 μονοσύνολα $S_1 = \{3\}$, $S_2 = \{5\}$, $S_3 = \{7\}$, $S_4 = \{9\}$, $S_5 = \{1\}$, $S_6 = \{2\}$, $S_7 = \{6\}$, παρουσιάστε τα up-δένδρα που προκύπτουν μετά από την εκτέλεση κάθε μιας από τις ακόλουθες λειτουργίες, δεδομένου ότι χρησιμοποιείται η στρατηγική για μείωση ύψους: $A = \text{Union}(S_1, S_2)$, $B = \text{Union}(A, S_3)$, $C = \text{Union}(S_4, S_5)$, $D = \text{Union}(S_7, S_6)$, $E = \text{Union}(C, D)$, $F = \text{Union}(B, E)$. Διαλέξτε ένα από τα βαθύτερα φύλλα του πάνω δένδρου που προκύπτει και εφαρμόστε μια Find με συμπίεση μονοπατιού στο στοιχείο αυτό. [5%]
- δ. Χρησιμοποιώντας τη συνάρτηση κατακερματισμού της διαίρεσης ως πρωτεύουσα συνάρτηση και τη συνάρτηση $h(K) = (K+3) \% m$ ως δευτερεύουσα συνάρτηση κατακερματισμού να εισάγεται τα κλειδιά 22, 15 και 8 σε έναν πίνακα 7 θέσεων που είναι οργανωμένος με τη μέθοδο του ταξινομημένου κατακερματισμού με ανοικτή διεύθυνση. [5%]



(α)

Σχήμα 2

Σημείωση: Θα πρέπει να δείξετε όλα τα ενδιάμεσα αποτελέσματα που προκύπτουν μετά από κάθε πράξη.

2. Ως σωροί μεγίστου-ελαχίστου χαρακτηρίζονται οι ουρές προτεραιότητας που υποστηρίζουν τις πράξεις DeleteMin, DeleteMax, FindMin, FindMax, και Insert. Προτείνετε μια επέκταση της δομής του σωρού ώστε να υποστηρίζει αποδοτικά όλες τις παραπάνω πράξεις. [15%]

Υπόδειξη: Φροντίστε ώστε η συνθήκη της σωρού να εναλλάσσεται μεταξύ μεγίστου και ελαχίστου καθώς αλλάζουν τα επίπεδα του δένδρου.

Προγραμματιστικό Μέρος

Υλοποιήστε κάποιο από τα παρακάτω προγράμματα.

1. Hash Tables

Κατασκευάστε ένα πρόγραμμα που να υλοποιεί τις διαδικασίες αναζήτησης, εισαγωγής και διαγραφής σε έναν πίνακα κατακερματισμού που είναι οργανωμένος με κάθε μια από τις ακόλουθες μεθόδους:

- των ξεχωριστών αλυσίδων,
- των μικτών αλυσίδων με κελάρι,
- του ταξινομημένου διπλού κατακερματισμού.

Αποφασίστε μόνοι σας ποια συνάρτηση θα χρησιμοποιήσετε ως πρωτεύουσα και ποια ως δευτερεύουσα συνάρτηση κατακερματισμού και ποιο θα είναι το μέγεθος του κελαριού. Οι λειτουργίες που θα πρέπει να υλοποιήσετε είναι οι Insert, LookUp, Delete και PrintHashTable. [E]

2. Splay Δένδρα

Υλοποιήστε ένα Splay δένδρο. Οι λειτουργίες που θα πρέπει να υλοποιήσετε είναι οι Insert, Delete, LookUp και PrintTree. [M-Δ]

3. Κόκκινα-Μαύρα Δένδρα

Υλοποιήστε ένα κοκκινο-μαύρο δένδρο. Οι λειτουργίες που θα πρέπει να υλοποιήσετε είναι οι Insert, LookUp και PrintTree. [Δ]

4. AVL Δένδρα

Υλοποιήστε ένα AVL δένδρο. Οι λειτουργίες που θα πρέπει να υλοποιήσετε είναι οι Insert, LookUp και PrintTree. [Δ]