Μηνάς Αδαμάντιος ΑΕΜ: 2373

Μουστάκας Αριστείδης ΑΕΜ: 2380

Διδάσκων: Παπαδόπουλος Απόστολος

Εργασία Εαρινού Εξαμήνου 2014

Το πρόγραμμα που δημιουργήσαμε υλοποιεί τους συνδέσμους μεταξύ σελίδων. Αρχικά φτιάξαμε μία κλάση node η οποία συμβολίζει τον κόμβο ενός δένδρου, με παραμέτρους την αριθμητική τιμή του κόμβου, και 2 δείκτες σε node (left και right) που δείχνουν σε άλλους κόμβους του δένδρου. Έπειτα φτιάξαμε και μία κλάση link η οποία υλοποιεί τον σύνδεσμο μεταξύ σελίδων, με παραμέτρους ίδιους με την node με διαφορά ότι αντί για δείκτες σε node έχει δείκτες σε link(left και right), καθώς και έναν ακόμα δείκτη σε node (filos) που είναι η ρίζα ενός AVL δένδρου οι οποία περιέχει τους «γείτονες» της κάθε σελίδας, δηλαδή τους συνδέσμους που έχει μια σελίδα προς άλλες σελίδες. Τέλος σχετικά με τις κλάσεις υλοποιήσαμε και την κλάση ClassInvertedIndex(η οποία περιέχει έναν δείκτη root που δείχνει σε link και στην ουσία είναι η ρίζα του δέδρου) η οποία αναπαριστά ένα κατάλογο από σελίδες με την βοήθεια ενός AVL δένδρου, όπου ο κάθε κόμβος του δένδρου αυτού είναι ένα link.

- Η κλάση Node:
  Η μόνη συνάρτηση που περιέχει είναι ο constructor ο οποίος δέχεται μια αριθμητική τιμή, την εισάγει στην παράμετρο value και αναθέτει στους δείκτες left και right την τιμή NULL. Destructor δεν υπάρχει καθώς η αποδέσμευση μνήμης υλοποιείται με άλλο τρόπο που θα παρουσιάσουμε παρακάτω.

- Η κλάση Link:
  Όπως και η node, έτσι και η link περιέχει μία μόνο συνάρτηση, τον constructor που επίσης πάιρνει μια αριθμητική τιμή ως όρισμα, την εισάγει στην παράμετρο value, αναθέτει στους δείκτες left και right την τιμή NULL και επιπλέον αναθέτει και την τιμή NULL στον δείκτη filos την τιμή NULL.

- Η κλάση ClassInvertedIndex:
  Η κλάση αυτή περιέχει πολλές συναρτήσεις καθώς υλοποιεί την δομή που χρησιμοποιούμε για την αποθήκευση των σελίδων.

1. Η συνάρτηση insertlink:
   Η συνάρτηση αυτή εισάγει έναν κόμβο link στο δένδρο. Δέχεται σαν ορίσματα την ρίζα του δένδρου(δείκτης root με αναφορά) και μία μεταβλητή

key η οποία είναι η τιμή που θέλουμε να εισάγουμε στο δένδρο. Αρχικά η συνάρτηση ελέγχει αν ο κόμβος που δώθηκε ως παράμετρος είναι NULL, στην οποία περίπτωση δεσμεύει δυναμικά χώρο για έναν νέο κόμβο και έπειτα καλεί στον κόμβο αυτό τον constructor στέλνοντας το key. Στην περίπτωση που ο δωθείς κόμβος δεν είναι NULL, τότε ελέγχει την τιμή του κόμβου με το key. Άν το key είναι μεγαλύτερο, τότε η συνάρτηση καλείται αναδρομικά για τον δεξιό δείκτη του αρχικού κόμβου(root->right). Αντίθετα αν το key είναι μικρότερο τότε η συνάρτηση καλείται αναδρομικά για τον αριστερό δείκτη του αρχικού κόμβου(root->left). Η διαδικασία αυτή συνεχίζεται είτε μέχρι να εισαχθεί ένας νέος κόμβος είτε μέχρι η τιμή key είναι ίση με την τιμή του κόμβου, το οποίο σημαίνει ότι υπάρχει ήδη κόμβος με την τιμή αυτή, και έτσι η συνάρτηση τερματίζεται.

2. Η συνάρτηση insertnode: Εκτελεί ακριβώς την ίδια διαδικασία με την insertlink, με διαφορά ότι δέχεται δείκτη σε node και καλείται για τους δείκτες filos του κάθε link.

3. Η συνάρτηση neos: Η συνάρτηση αυτή καλείται όταν δεχόμαστε την εντολή INSERT_LINK x y από το αρχείο commands.txt. Δέχεται ως ορίσματα την ρίζα(link* root) και 2 αριθμούς, τους x και y. Αρχικά καλεί την insertlink για τον αριθμό x και έπειτα για τον αριθμό y. Μετά καλεί 2 φορές την συνάρτηση insertnode, και τις στέλνει ως ορίσματα τον δείκτη filos του κόμβου που περιέχει το x καθώς και την τιμή y, ενώ την δέυτερη φορά στέλνει τον δείκτη filos του κόμβου με την τιμή y καθώς και την τιμή x. Έτσι εισάγουμε στο δένδρο τους κόμβους x και y, αν δεν υπάρχουν, και καταχωρούμε το y στο δένδρο με τους «γείτονες» του x καθώς και το x στο δένδρο με τους «γείτονες» του y.

4. Η συνάρτηση deletelink: Αυτή δέχεται ως ορίσματα την ρίζα root του δένδρου(δείκτης σε link) καθώς και την τιμή key προς διαγραφή. Αρχικά η συνάρτηση καλείται αναδρομικά μέχρι να βρει τον προς διαγραφή κόμβο,ενώ αν δεν υπάρχει απλά τερματίζεται, και εφόσον τον βρει διακρίνει σε ποια περίπτωση διαγραφής είναι(φύλλο, κόμβος με ένα υπόδενδρο, κόμβος με 2 υπόδενδρα) και ακολουθεί την κατάλληλη διαδικασία, ενώ στο τέλος χρησιμοποιεί την delete για την αποδέσμευση χώρου, και ενημερώνει τον αντίστοιχο δείκτη(left η right) του γονικού κόμβου του διεγραμμένου.

5. Η συνάρτηση deletenode: Κάνει ακριβώς το ίδιο με την deletelink με την μόνη διαφορά ότι δέχεται δείκτη σε node, και χρησιμοποιείται για την διαγραφή των δεικτών (node) filos των link.

6. Η συνάρτηση deletelinkroot: Υλοποιεί την διαγραφή της ρίζας ενός δένδρου, την οποία εναλλάσει με το αριστερότερο φύλλο του δεξιού υποδένδρου, η με το δεξιότερο φύλλο του αριστερού υποδένδρου, και τέλος διαγράφει με την deletelink τον κόμβο του οποίου η τιμή αντιγράφηκε στον κόμβο της ρίζας.

7. Η συνάρτηση deletenoderoot: Ακρίβώς το ίδιο με την deletelinkroot, αλλά για την διαγραφή ρίζας δένδρου με nodes.

8. Η συνάρτηση deleteboth: Καλείται όταν δεχόμαστε την εντολή DELETE_LINK x y από το αρχείο commands.txt. Δέχεται ως ορίσματα την ρίζα root του δένδρου και τις τιμές x και y. Σκοπός της είναι να διαγράψει τον κόμβο με την τιμή x από το δένδρο με τους «γείτονες» του κόμβου με τιμή y και αντίστοιχα να διαγράψει τον κόμβο με τιμή y από το δένδρο με τους «γείτονες» του x(με χρήση της deletenode η deletenoderoot). Στην περίπτωση που μετά τις διαγραφές ένας από τους κόμβους x και y δεν έχει «γείτονες», ο κόμβος διαγράφεται(με χρήση της deletelink η deletelinkroot). Επίσης στην περίπτωση πού ένας απο τους κόμβους x και y δεν υπάρχει ή δεν υπάρχουν οι αντίστοιχοι γείτονες, η συνάρτηση τερματίζεται.

9. Οι συναρτήσεις findlink και findnode: Δέχονται σαν ορίσματα την ρίζα του δένδρου(link η node) και ένα key, και επιστρέφουν τον δείκτη που δείχνει στον κόμβο με τιμή το key, αν υπάρχει, αλλιώς επιστρέφουν NULL.

10. Οι συναρτήσεις leftMostLeaf,rightMostLeaf: Βρίσκουν το αριστερότερο φύλλο και το δεξιότερο φύλλο του δείκτη που δέχονται αντιστοίχως.

11. Οι συναρτήσεις leftRot, rightRot, rightLeftRot, leftRightRot: Δέχονται σαν ορίσματα ένα δείκτη σε κόμβο του δένδρου, και εκτελούν τις αντίστοιχες περιστροφές left, right, rightleft, leftright. Υπάρχουν επίσης και οι συναρτήσεις leftRotnode, rightRotnode, rightLeftRotnode και η leftRightnode, που εκτελούν τις αντίστοιχες διαδικασίες για κόμβους node.

12. Οι συναρτήσεις Height και Heightnode: Δέχονται σαν ορίσματα έναν δείκτη σε κόμβο του δένδρου(link η node αντίστοιχα) και υπολογίζουν αναδρομικά το ύψος του δένδρου με ρίζα τον κόμβο που δώθηκε.

13. Οι συναρτήσεις test και testnode: Οι συναρτήσεις αυτές δέχονται ως ορίσματα έναν δείκτη σε κόμβο. Υπολογίζουν το ύψος του αριστερού και δεξιού υποδένδρου και στην περίπτωση που η διαφορά τους είναι μεγαλύτερη από 1 ή μικρότερη από -1 διακρίνουν την περίπτωση και

πραγματοποιούν την κατάλληλη περιστροφή καλώντας μια από τις συναρτήσεις leftRot, rightRot, rightLeftRot, leftRightRot ή μία από τις συναρτήσεις leftRotnode, rightRotnode, rightLeftRotnode, leftRightnode, για links η nodes αντίστοιχα. Οι συναρτήσεις καλούνται μετά απο εισαγωγή ή διαγραφή. Σχετικά με την εισαγωγή, διατηρούμε μία global μεταβλητή counter η οποία αρχικοποιείται με τιμή 0 μόλις γίνει εισαγωγή ενός κόμβου. Έπειτα καθώς «ξετυλίγεται» η αναδρομική διαδικασία, η μεταβλητή counter αυξάνεται κατά μία μονάδα μέχρις ότου πάρει την τιμή 2. Όταν γίνει αυτό σημαίνει ότι έχουμε φτάσει στον γονικό κόμβο του κόμβου στον οποίο εισήχθηκε στον δείκτη left η right ο καινούριος κόμβος, ο οποίος είναι πιθανόν να χρειάζεται ισορρόπηση. Έπειτα καλείται η test, ή η testnode, με όρισμα τον δείκτη που δείχνει στον κόμβο αυτό. Από την άλλη, η deletelink/deletenode κρατάει έναν global δείκτη parent link η node ανάλογα με την περίπτωση ο οποίος δείχνει στον γονιό του δείκτη που διεγράφει. Έπειτα καλείται η συνάρτηση test ή η testnode για τον δείκτη parent καθώς και για τον δείκτη που δείχνει στον κόμβο που διεγράφει, στην περίπτωση που ο κόμβος αυτός δεν ήταν φύλλο, οι οποίοι είναι οι πιθανοί προς εξισορρόπηση κόμβοι.

14. Η συνάρτηση inorder: Δέχεται σαν ορίσματα έναν δείκτη node και μία μεταβλητή ofstream με αναφορά και τυπώνει στο ρεύμα εξόδου τις τιμές του δένδρου με ρίζα τον δείκτη που δώθηκε, με αύξουσα σειρά.

15. Η συνάρτηση nodecounter: Δέχεται σαν ορίσματα έναν δείκτη node και μετρά το πλήθος των κόμβων, εισάγωντας το στην global μεταβλητή counter.

16. Η συνάρτηση writeindex: Καλείται όταν δωθεί σαν εντολή η WRITE_INDEX output.txt από το αρχείο commands.txt. Δέχεται σαν ορίσματα έναν δείκτη link και μια μεταβλητή ofstream με αναφορά και τυπώνει στο ρεύμα εξόδου, που είναι το αρχείο output.txt την τιμή του κάθε κόμβου το πλήθος των «γειτόνων» του καθώς και τις τιμές του δένδρου των «γειτόνων» του με χρήση της inorder.

Τέλος, η συνάρτηση main αρχικά διαβάζει από το αρχείο commands.txt το όνομα του αρχείου με τις τιμές(input) και εισάγει τα ζευγάρια των αριθμών με τη συνάρτηση neos. Στη συνέχεια δέχεται εντολές INSERT_LINK x y και DELETE_LINK x y, καλώντας τις κατάλληλες συναρτήσεις. Όταν δεχτεί την τελευταία εντολή WRITE_INDEX output.txt καλέι την συνάρτηση writeindex και γράφει στο αρχείο εξόδου τα αποτελέσματα του προγράμματος, το οποίο έπειτα τερματίζεται.