

Μέλος εργασίας:

Νταλές Αθανάσιος (3160117)

Τα αρχεία του φακέλου είναι:

- ST.java: ο ΑΤΔ για πίνακα συμβόλων
- BookInfo.java: η κλάση για την αναπαράσταση πληροφοριών ενός βιβλίου
- List.java: η συνδεδεμένη λίστα
- Sort.java: το αρχείο ταξινόμησης της συνδεδεμένης λίστας (παρμένο έτοιμο από την 2η εργασία)
- SystemMenu.java: το μενού διαχείρισης

Υλοποίηση εργασίας:

Καταρχάς για την υλοποίηση της συνδεδεμένης λίστας έγιναν τα ακόλουθα: πέρα από τα γνωστά (εισαγωγή, διαγραφή, ταξινόμηση) υπάρχει και μία μέθοδος (checking(List list)) η οποία ελέγχει η λίστα να είναι δομημένη έτσι όπως αναφέρεται στην εκφώνηση. Πιο συγκεκριμένα εάν έχουν εισαχθεί 2 βιβλία με το ίδιο isbn, τότε τα αντίτυπα του δεύτερου προστίθενται σε αυτά του πρώτου και το δεύτερο βιβλίο διαγράφεται από τη λίστα.

Όσον αφορά τον ΑΤΔ τύπο ST: στο αρχείο ST.java έχει υλοποιηθεί η κλάση TreeNode η οποία αντιστοιχεί σε καθένα κόμβο του δέντρου.

Εισαγωγή:

Η μέθοδος εισαγωγής αποθήκης έχει υλοποιηθεί όπως η ενδεδειγμένη σε τυχαιοποιημένα δέντρα. Με πιθανότητα $1/N + 1$ εισάγεται κόμβος στη ρίζα μέσω της μεθόδου insertT η οποία εισάγει ρίζα μέσω των κατάλληλων περιστροφών (μέθοδοι rotR και rotL), ενώ αναδρομικά εισάγεται κόμβος σε διαφορετική περίπτωση. Μετά από κάθε επιτυχημένη εισαγωγή ενημερώνεται το πεδίο N.

Για την εισαγωγή βιβλίου σε μία αποθήκη, η μέθοδος found εντοπίζει άμα υπάρχει η αποθήκη με το δοσμένο id. Αν δεν υπάρχει τυπώνει μήνυμα λάθους και τερματίζει, ενώ σε διαφορετική περίπτωση, είτε εισάγει το βιβλίο, εάν δεν υπάρχει, είτε προσθέτει τα δοσμένα αντίτυπα στο ήδη υπάρχον βιβλίο με το δοσμένο isbn.

Διαγραφή:

Η μέθοδος removeWarehouse(int nodeid) ψάχνει μέσω της found εάν υπάρχει αποθήκη με το δοσμένο nodeid. Εφόσον υπάρχει, τότε η μέθοδος removeR αναλαμβάνει να διαγράψει τον κατάλληλο κόμβο. Σε περίπτωση που το κλειδί του κόμβου δεν ισούται με το δοσμένο κλειδί, τότε η μέθοδος less μας οδηγεί που να ψάξουμε μετά. Όταν βρούμε τον κατάλληλο κόμβο τότε τον αφαιρούμε με την joinLR η οποία αναλαμβάνει να αντικαταστήσει τον κόμβο με έναν άλλον που βρίσκεται πιο κάτω στο

δέντρο(παίρνουμε πιθανότητες, για να υλοποιηθεί η μέθοδος, σύμφωνα με τις αρχές των τυχαιοποιημένων δέντρων) . Πάλι στο τέλος της διαγραφής ενημερώνεται το πεδίο N.

Όσο για τη μέθοδο `removeBook(int nodeid, int isbn)`, ψάχνουμε με την `found` εάν υπάρχει η αποθήκη με κωδικό `nodeid`. Εφόσον υπάρχει τότε ψάχνουμε στη λίστα της για το βιβλίο με κωδικό `isbn`. Εφόσον και αυτό υπάρχει, τότε πάμε και αφαιρούμε ένα αντίτυπο από το εν λόγω βιβλίο. Εάν τα εναπομείναντα αντίτυπα είναι 0, τότε διαγράφουμε το βιβλίο από την αποθήκη. Σε διαφορετική περίπτωση ουσιαστικά ανανεώνουμε τα αντίτυπά του.

Αναζήτηση:

Η μέθοδος `searchByWarehouse(int nodeid)` εντοπίζει μέσω της αναδρομικής αναζήτησης `searchR(TreeNode h, int nodeid)` τον κόμβο(αποθήκη) με κλειδί `nodeid` και σε περίπτωση που υπάρχει εκτυπώνει τα περιεχόμενα της λίστας του μέσω της μεθόδου `display()` του αρχείου `List.java`. Η `searchR(TreeNode h, int nodeid)` κάνει το εξής: εάν το δοσμένο κλειδί ισούται με αυτό του κόμβου `h`, τότε επιστρέφει τον `h`, εάν είναι μικρότερο καλεί αναδρομικά πάλι τη συνάρτηση με `h` το `h.l` αλλιώς με το `h.r`.

Η συνάρτηση `searchBookInWarehouse(int nodeid, int isbn)` πάλι μέσω της `searchR` βρίσκει τον κόμβο με κλειδί `nodeid` και στην λίστα αυτού αναζητά το βιβλίο με κλειδί `isbn` μέσω της μεθόδου `find2`. Εάν το βρει, τότε εκτυπώνει τα αντίτυπα του στην συγκεκριμένη αποθήκη αλλιώς μήνυμα λάθους.

Η μέθοδος `searchBook(int isbn)` διασχίζει το δέντρο και εκτυπώνει τις αποθήκες που έχουν το βιβλίο με κλειδί `isbn` καθώς και τα αντίτυπα αυτού. Η μέθοδος διάσχισης που χρησιμοποιήθηκε είναι η `preorder` όπου γίνεται η επεξεργασία πρώτα σε έναν κόμβο, μετά στον αριστερό του και τέλος στον δεξιό του.

Εκτύπωση δέντρου:

Η μέθοδος `printTree(PrintStream stream)` κάνοντας πάλι χρήση της `preorder` διάσχισης, εκτυπώνει κάθε αποθήκη με τα βιβλία της και τα αντίστοιχα αντίτυπα.

Μενού Διαχείρισης:

Στο μενού διαχείρισης τέλος έχω προσθέσει τις 3 αποθήκες με τα βιβλία που αναφέρονται στο παράδειγμα της εργασίας. Από εκεί και πέρα μέσω του μενού παρέχονται στον χρήστη όλες οι δυνατότητες της δομής, δηλαδή να εισάγει αποθήκες βιβλία, να τα διαγράψει καθώς και να τα αναζητήσει με κάθε πιθανό τρόπο. Με την επιλογή 8 μπορεί μετά από κάθε του ενέργεια να εκτυπώνει το δέντρο, για να ελέγχει την ορθότητα των ενεργειών του.