

Δομές Δεδομένων

Εργασία 1

ΤΟΣΚΟΛΛΑΡΙ ΡΟΝΑΛΝΤ

p3160244

a)**StringStack/StringStackImpl**: Στην συγκεκριμένη διεπαφή γίνεται χρήση των Generics έτσι ώστε η στοίβα να μπορεί όλους τους τύπους δεδομένων. Αργότερα στο implementation της StringStack από την StringStackImpl γίνεται υλοποίηση όλων των μεθόδων της διεπαφής μια προς μια χρησιμοποιώντας αντικείμενα τυπου Node τα οποία υποστηρίζουν επίσης Generics. Η κλάση StringStackImpl περιέχει έναν κατασκευαστή ο οποίος αρχικοποιεί το μέγεθος της στοίβας ίσο με 0 , και το πρώτο στοιχείο(head) με null. Οι μέθοδοι pop/peek κάνουν σχεδόν την ίδια δουλειά μόνο που στην χρήση της pop διαγράφουμε και το στοιχείο που βρίσκεται στην κορυφή της στοίβας. Στην κλήση της push απλά βάζουμε ένα στοιχείο στην κορυφή της στοίβας και αυξάνουμε το μέγεθος της κατά 1. Υπάρχουν και οι μέθοδοι isEmpty η οποία ελέγχει αν η στοίβα είναι άδεια, printStack η οποία απλά εκτυπώνει την στοίβα μας, και size η οποία μας δίνει το μέγεθος της στοίβας, δηλαδή πόσα στοιχεία περιέχει.

IntQueue/IntQueueImpl: Ομοίως και σε αυτήν την υλοποίηση της διεπαφής γίνεται χρήση των Generics. Όπως και πριν γίνεται υλοποίηση όλων των μεθόδων μια προς μια κάνοντας χρήση αντικειμένων Node. Ο κατασκευαστής μας εδώ αρχικοποιεί το head/tail της ουράς με null και το μέγεθος της ουράς με 0. Οι μέθοδοι get/peek κάνουν την ίδια εργασία αλλά στην get κάνουμε και αφαίρεση του στοιχείου που παίρνουμε δηλαδή του πρώτου που μπήκε στην ουρά. Στην κλήση της put προσθέτουμε ένα επιπλέον στοιχείο στο τέλος της ουράς και αυξάνουμε το μέγεθος κατά 1. Επίσης

υπάρχουν οι μέθοδοι isEmpty η οποία ελέγχει αν η ουρά είναι άδεια, printQueue όπου εκτυπώνει την ουρά, και size όπου μας δίνει το μέγεθος της ουράς.

b) TagMatching: Η κλάση αυτή αποφασίζει εάν ένα αρχείο html είναι ορθό. Αρχικά έχουμε τις μεταβλητές validFile η οποία παίρνει την τιμή true εάν το αρχείο μας είναι ορθό και false εάν δεν είναι, reader η οποία είναι της κλάσης BufferedReader που μας βοηθάει να διαβάσουμε το αρχείο, path όπου η τιμή της είναι πάντα args[0] και δίνεται από το χρήστη κατά την εκτέλεση του προγράμματος, έχουμε την line η οποία θα έχει την γραμμή του html αρχείου, φτιάχνουμε την στοίβα μας η οποία θα έχει μέσα τα tags του html αρχείου, και τέλος lastPush όπου εκεί θα έχουμε το προτελευταίο tag της στοίβας. Σε πρώτη φάση διαβάζουμε το αρχείο γραμμή προς γραμμή μέχρι να βρεθούμε σε EOF = null, δηλαδή μέχρι να φτάσουμε στο τέλος του αρχείου. Τώρα για κάθε γραμμή του αρχείου φτιάχνουμε ένα String tag το οποίο θα έχει το tag ή τα tags κάθε γραμμής. Τρέχουμε την γραμμή σε όλο το μήκος της, και κρατάμε την θέση που βρισκόμαστε κάθε φορά. Εάν βρούμε το αρχικό σύμβολο μιας ετικέτας(<), τότε φτιάξε το την ετικέτα, εάν βρεθεί το τελευταίο σύμβολο της ετικέτας(>), τότε κράτα την τελευταία προσθήκη και βάζουμε το νέο tag στην στοίβα. Κάνουμε έλεγχο της τελευταίας προσθήκης (tag) και της προτελευταίας(lastPush) εάν η πρώτη είναι closing tag της δεύτερης. Σε περίπτωση που είναι τότε αφαιρούμε τα tags από την στοίβα. Και κρατάμε πάλι το τελευταίο στοιχείο της στοίβας. Κάνουμε μια εκτύπωση της τωρινής κατάστασης της στοίβας και συνεχίζουμε στην επόμενη γραμμή. Τέλος σε περίπτωση που η στοίβα μας είναι άδεια, τότε το αρχείο μας είναι σωστά γραμμένο.

c)**NetBenefit:** Σε αυτήν την κλάση όταν τρέχουμε το πρόγραμμα δίνουμε ως είσοδο το μονοπάτι ενός αρχείου txt το οποίο περιέχει κάποιες συναλλαγές. Ομοίως και σε αυτή την κλάση έχουμε τα reader,path,line που κάνουν την ίδια δουλειά όπως και στην TagMatching, φτιάχνουμε 2 ουρές queueQuantity και queuePrice, η πρώτη περιέχει την ποσότητα που αγοράζουμε ενώ η δεύτερη την τιμή της ποσότητας που αγοράζουμε. Φτιάχνουμε επίσης έναν πίνακα lineTokens ο οποίος θα περιέχει τα tokens κάθε γραμμής που θα διαβάζουμε από το txt αρχείο. Η μεταβλητή sellPrice έχει την τιμή που πουλάμε τις μετοχές μας, και η sellQuantity έχει την ποσότητα των μετοχών που πουλάμε. Στην μεταβλητή total έχουμε το σύνολο των διαθέσιμων μετοχών και τέλος στην benefit το όφελος που έχουμε από τις ενέργειες αυτές. Διαβάζουμε το αρχείο μέχρι να βρούμε EOF = null, δηλαδή μέχρι να φτάσουμε στο τέλος του αρχείου. Χωρίζουμε την γραμμή που διαβάσαμε σε tokens και βλέπουμε την κίνηση που κάνουμε (buy ή sell) και εκτελούμε τις ανάλογες ενέργειες. Αν είναι buy τότε βάζουμε τις μετοχές και τις τιμές τους στις ουρές και αυξάνουμε τις συνολικές μετοχές. Αλλιώς αν είναι sell, βάζουμε την ποσότητα και την τιμή στις αντίστοιχες μεταβλητές που ορίσαμε στην αρχή. Κάνουμε έλεγχο αν είναι έγκυρη κίνηση. Όσο έχω ακόμη μετοχές που πρέπει να πουληθούν, ελέγχω εάν είναι η τελευταία πώληση που κάνω ή όχι. Αποθηκεύω εάν υπάρχουν τις μετοχές που έχουν μείνει στην μεταβλητή remaining και την αντίστοιχη τιμή τους στην rPrice, βρίσκω το όφελος και βάζω στην ουρά τις μετοχές που έχουν μείνει και θέτω τις μετοχές που πρέπει να πουληθούν σε 0. Εάν δεν είναι η τελευταία πώληση αφαιρώ τις μετοχές που έχω στην ουρά και υπολόγισε το όφελος για αυτή την πώληση. Τέλος εκτυπώνω το όφελος από τις κινήσεις μας στο txt αρχείο.