



Εργασία 1 Δομές Δεδομένων

Σαραντος Τζωρτζης

-

Αναστάσιος Ζαχαριουδάκης

P3170157

p3170048

ΕΡΩΤΗΜΑ «Α»

- Οι διεπαφές στο μέρος Α υλοποιήθηκαν με τη βοήθεια του κωδικα απο το εργαστήριο
- Η Stack και η Queue έγιναν generic κλάσεις !
- Ποιο συγκεκριμένα:
 - **StringStackImpl** -> Η μέθοδος **push(T item)** λαμβάνει σαν παραμετρο ενα αντικείμενο τύπου <T> (καθώς είναι generic) και το τοποθετεί "πάνω-πάνω" στη λίστα ,γίνεται δηλατη το first node, και ελεγχει επισης αν η στοιβα είναι αδεια, σε αυτην την περίπτωση first=last=item
 - Η μέθοδος **pop()** επιστρέφει αντικειμενο τυπου «T», διότι είναι generic, και αυτό που κάνει είναι οτι , ελεγχει αν η λιστα είναι αδεια, εαν ναι πεταει μήνυμα μεσω της NoSuchElementException, εάν όχι, κάνει τις καταλληλες μεταθεσεις στα Node ωστε να γίνει μια σωστη "διαγραφή" απο τη στοιβα.
 - Η μέθοδος **peek()** επιστρέφει το στοιχειο που εισήλθε τελευταίο στη στοιβα (τύπου <T>), παρομοια και αυτη ελεγχει για το αν η στοιβα είναι αδεια
 - Η μέθοδος **printStack(PrintStream stream)** τυπώνει την στοίβα , κομβο προς κόμβο , μεχρι δηλαδη η βοηθητική μεταβλητη current να γίνει Null (η current στην αρχη της while που διασχίζει τα nodes παίρνει σαν τιμη τον πρώτο κομβο)
 - Η μέθοδος **size()** που επιστρέφει εναν ακεραιο αριθμο που αντιπροσωπευει το μέγεθος της στοίβας
 - Η μέθοδος **isEmpty()** που επιστρέφει true/false ανάλογα αν η στοίβα είναι αδεια η όχι, καθώς ελεγχει αν ο πρώτος κόμβος ισούται με το Null

Συνέχεια ερωτήματος «Α»

- Η `IntQueueImpl` υλοποιήθηκε παρόμοια με την `StringStackImpl`
- Οι μέθοδοι που είναι ίδιοι είναι οι `isEmpty()`, `size()`
- Οι μέθοδοι που έχουν κάποιες μετατροπές είναι οι
 - `get()` Η οποία βγαζει απο την ουρα το παλαιότερο στοιχείο της, το `firstNode`, εαν η ουρα δεν είναι αδεια
 - `peek()` Η οποία κάνει πλέον return το `lastNode` (εάν δεν είναι άδεια η ουρα)
 - `put(T insertItem)` Η οποία τοποθετεί ένα νέο στοιχείο τύπου `<T>` (καθώς είναι generic) στο τέλος της ουράς, κάνοντας έλεγχο για το αν η ουρα είναι αδεια, στην περίπτωση αυτή `firstNode=lastNode=insertItem`
 - Και επιπλέον προσθέσαμε **την `setFirstNode(T item)`** ώστε στο ερώτημα «Γ» όταν υπάρχει κάποιο `sell`, θέτουμε σαν `FirstNode`, τη διαφορά του `FirstNode` και `sell`
πχ: εαν `sell=100`
και `FirstNode=110`
τότε θέτουμε με την `setFirstNode`, στο `FirstNode` την τιμή 10

ΕΡΩΤΗΜΑ «B»

- Στο ερώτημα B , υλοποιούνται οι μέθοδοι που κατασκευάστηκαν στο ΕΡΩΤΗΜΑ «Α», ποιο συγκεκριμένα :
 - Η μέθοδος **push(T item)** χρησιμοποιείται για να εισάγουμε ένα **tag** στην στοίβα , όταν αυτό είναι **tag** ανοίγματος (`<tagname>`)
 - Η μέθοδος **peek()** χρησιμοποιείται για να ελέγξουμε αν το τελευταίο **tag** που έχει εισαχθεί στην στοίβα είναι ίδιο με το **tag** κλεισμάτων που διαβάστηκε (`</tagname>`)
 - Η μέθοδος **pop()** χρησιμοποιείται για να βγάλουμε ένα tag από την στοίβα , ελέγχοντας πριν πάντα με την μέθοδο **peek()** , για το αν ισχύει αυτή η ταύτιση
 - Η μέθοδος **printStack()** χρησιμοποιείται για να τυπώσουμε την στοίβα
 - Η μέθοδος **isEmpty()** χρησιμοποιείται για να ελέγξουμε αν η στοίβα είναι άδεια, σε περίπτωση δηλαδή που η στοίβα είναι άδεια και το πρώτο tag που δοθεί είναι **tag** κλεισμάτων (`"</>"`), τότε γίνεται brake και τυπώνεται μήνυμα ότι τα **tags** είναι λάθος
 - Επίσης χρησιμοποιείται και για τον τελικό έλεγχο, δηλαδή αν είναι άδεια η στοίβα και δεν έχει προηγηθεί κάποιος brake , τότε τυπώνεται μήνυμα ότι τα tags είναι Matching

ΕΡΩΤΗΜΑ «Γ»

- Χρησιμοποιούμε την ουρά FIFO που υλοποιήσαμε στο Α ερώτημα, αρχικά δημιουργώντας δυο ουρές: μία για να κρατούνται οι μετοχές που αγοράστηκαν και μία για τη τιμή τους.
- Χρησιμοποιούμε την μέθοδο `put(T insertItem)` για να εισάγουμε στοιχείο στην καθε ουρά. Δηλαδή εάν διαβαστεί στη γραμμή η λέξη buy τότε ο αριθμος που βρίσκεται μετα απο αυτή καταχωρείται σε μια μεταβλητη,μετατρέπεται σε "int" και τοποθετείται στην ουρά, αντίστοιχα και για την λέξη price.
- Χρησιμοποιούμε την μέθοδο `get()` για να αφαιρέσουμε στοιχεία απο τις ουρές.Όταν δηλαδη διαβαστεί η λέξη sell, τοτε ο αριθμος που υπαρχει μετα απο αυτή καταχωρείται σε μια μεταβλητη,μετατρέπεται σε "int" και αναλογα με αυτον τον αριθμο αφαιρούνται είται μειώνονται στοιχεία απο την ουρά buy.
 - Επίσης οταν αφαιρεθεί ενα στοιχειο απο την ουρά buy , αφαιρείτα και ενα στοιχείο απο την ουρά price.
- Χρησιμοποιούμε την μέθοδο `printQueue()`, για να τυπώσουμε τις ουρές όπου χρειάζεται