

ΑΝΤΙΚΕΙΜΕΝΟΣΤΡΑΦΗΣ ΠΡΟΓΡΑΜΜΑΤΙΣΜΟΣ

Χειμερινό εξάμηνο ακ. έτους 2021-2022

Α' ομάδα ασκήσεων

1. Η παρούσα άσκηση σκοπό έχει να σας εξασκήσει στις συναρτήσεις κατασκευής-καταστροφής στην περίπτωση που έχουμε σύνθεση κλάσεων και στην προσβασιμότητα και εμβέλεια μεταξύ των μελών των κλάσεων.

Υλοποιήστε σε C++ μια προσομοίωση της λειτουργικότητας μιας βιβλιοθήκης. Η βιβλιοθήκη αποτελείται από μια βάση, ένα ντουλάπι και τρία ράφια (με αυτή τη σειρά). Το ντουλάπι εσωτερικά έχει δύο ράφια.

Στη βιβλιοθήκη, ανά πάσα στιγμή βρίσκονται N βιβλία (το N μπορεί να είναι ίσο και με 0). Ένα βιβλίο δεν είναι τίποτα περισσότερο από μια δόμηση δεδομένων που αποτελείται από τον τίτλο του βιβλίου, το ονοματεπώνυμο του συγγραφέα του και το ISBN του (ακέραιος). Το κάθε ράφι (είτε εξωτερικό είτε εσωτερικό) φιλοξενεί κάποια από τα βιβλία. Κάθε ράφι έχει μέγιστη χωρητικότητα $NMax$. Η χωρητικότητα είναι ίδια για όλα τα ράφια.

Κατά την κατασκευή μιας βιβλιοθήκης, εκτυπώνεται ένα μήνυμα ότι μια βιβλιοθήκη κατασκευάζεται. Κατά την κατασκευή ενός ραφιού, εκτυπώνεται ένα μήνυμα ότι ένα ράφι κατασκευάζεται. Κατά την κατασκευή ενός ντουλαπιού, εκτυπώνεται ένα μήνυμα ότι ένα ντουλάπι κατασκευάζεται. Κατά την κατασκευή μιας βάσης, εκτυπώνεται ένα μήνυμα ότι μια βάση κατασκευάζεται. Κατά την κατασκευή τους, τα ράφια δεν περιέχουν κανένα βιβλίο.

Κατά την καταστροφή της βιβλιοθήκης, εκτυπώνεται σχετικό μήνυμα. Σχετικά μηνύματα εκτυπώνονται και στις περιπτώσεις καταστροφής ραφιών, ντουλαπιών και βάσης βιβλιοθήκης.

Στη βιβλιοθήκη τοποθετούμε ένα βιβλίο (`place_book`) προσδιορίζοντας με ένα αριθμό το χώρο στον οποίο θα τοποθετηθεί και τοποθετώντας το σε αυτόν το χώρο. Δηλαδή, αντιστοιχώντας

- 1: στο πάνω ράφι της βιβλιοθήκης
- 2: στο μεσαίο ράφι της βιβλιοθήκης
- 3: στο κάτω ράφι της βιβλιοθήκης
- 4: στο πάνω ράφι του ντουλαπιού της βιβλιοθήκης
- 5: στο κάτω ράφι του ντουλαπιού της βιβλιοθήκης

Σε αυτή την περίπτωση, αυξάνεται κατά 1 ο αριθμός των βιβλίων που είναι τοποθετημένα στη βιβλιοθήκη. Ταυτόχρονα γίνεται εκτύπωση του χώρου στον οποίο γίνεται η τοποθέτηση, π.χ. αν ο προσδιοριστής είναι ο αριθμός 2, η εκτύπωση είναι "`placing book in middle bookcase self`" ή κάτι ανάλογο.

Μπορούμε επίσης να πάρουμε ένα βιβλίο από τη βιβλιοθήκη (`take_book`) προσδιορίζοντας το χώρο από το οποίο το αφαιρούμε και αφαιρώντας το από το χώρο αυτό. Ταυτόχρονα γίνεται εκτύπωση του χώρου από τον οποίο γίνεται η αφαίρεση. Ο προσδιορισμός του χώρου γίνεται με τρόπο ανάλογο με την τοποθέτηση (Σημείωση: το βιβλίο που αφαιρείται είναι τυχαίο, όχι προσδιορισμένο). Σε αυτή την περίπτωση, ο αριθμός των βιβλίων που βρίσκονται στη βιβλιοθήκη μειώνεται κατά 1.

Τέλος, γίνεται εκτύπωση (`print`) στη βιβλιοθήκη, κάνοντας εκτύπωση στο ντουλάπι και στα ράφια της.

Η τοποθέτηση ενός βιβλίου στο ντουλάπι (`place_book`), πραγματοποιείται προσδιορίζοντας το ράφι στο οποίο θα γίνει η τοποθέτηση και τοποθετώντας το στο ράφι αυτό. Για τον προσδιορισμό του ραφίου, με 1 προσδιορίζουμε το πάνω ράφι και με 2 το κάτω ράφι του ντουλαπιού της βιβλιοθήκης. Η αφαίρεση ενός βιβλίου από το ντουλάπι (`take_book`), πραγματοποιείται προσδιορίζοντας το ράφι από οποίο θα γίνει η αφαίρεση και αφαιρώντας το από το ράφι αυτό. Η εκτύπωση του ντουλαπιού (`print`) ανάγεται στις εκτυπώσεις των ραφιών του.

Η τοποθέτηση ενός βιβλίου σε ένα ράφι (`place_book`), πραγματοποιείται αυξάνοντας τον αριθμό των βιβλίων που είναι τοποθετημένα σε αυτό κατά 1, προσθέτοντας το βιβλίο αυτό στα βιβλία που φιλοξενεί και εκτυπώνοντας τα περιεχόμενα του βιβλίου. Αντίστοιχα, η αφαίρεση ενός βιβλίου από ένα ράφι (`take_book`), πραγματοποιείται μειώνοντας τον αριθμό των βιβλίων που είναι τοποθετημένα σε αυτό κατά 1 και αφαιρώντας το βιβλίο από τα βιβλία που φιλοξενεί. Τέλος, γίνεται εκτύπωση (`print`) ενός ραφίου εκτυπώνοντας τα στοιχεία των βιβλίων που περιέχει. Αν είναι άδειο, να εκτυπώνεται σχετικό μήνυμα.

Στις περιπτώσεις αφαιρέσεων βιβλίων από κενούς χώρους ή τοποθετήσεων σε γεμάτους, να επιστρέφεται από τις αντίστοιχες συναρτήσεις τιμή `false` και να τυπώνεται μήνυμα λάθους. Διαφορετικά, να επιστρέφεται η τιμή `true`. Το ίδιο ισχύει και για αφαιρέσεις και τοποθετήσεις βιβλίων στη βιβλιοθήκη.

Η βάση της βιβλιοθήκης έχει μόνο δομικό ρόλο για τη βιβλιοθήκη –απλά συμμετέχει στις διαδικασίες αρχικοποίησης και καταστροφής της βιβλιοθήκης.

Υλοποιήστε τα παραπάνω, μέσω των κατάλληλων κλάσεων, ορίζοντας τα μέλη-δεδομένα που χρειάζονται και τις συναρτήσεις-μέλη που υλοποιούν την παραπάνω συμπεριφορά.

Για να αναδείξετε την προσομοίωσή σας, να υλοποιήσετε μια συνάρτηση `main` η οποία:

- (α') να δημιουργεί μια βιβλιοθήκη
- (β') να δημιουργεί L βιβλία
- (γ') να πραγματοποιεί $K1$ τυχαίες τοποθετήσεις βιβλίων στη βιβλιοθήκη
- (δ') να πραγματοποιεί $K2$ τυχαίες αφαιρέσεις βιβλίων από τη βιβλιοθήκη
- (ε') να κάνει εκτύπωση στην βιβλιοθήκη

Σημείωση: Τα $NMax$, L , $K1$ και $K2$ είναι παράμετροι της προσομοίωσης και να διαβάζονται από τη γραμμή εντολής με τη σειρά αυτή.

2. Να υλοποιηθεί σε C++ μια προσομοίωση εξυπηρέτησης πελατών σε μια τράπεζα. Η τράπεζα εξυπηρετεί τους πελάτες της μέσω των ταμείων της. Οι πελάτες δεν έχουν ξεχωριστή αναπαράσταση. Απλά αντικατροπτίζονται από τον αριθμό προτεραιότητας που παίρνουν κατά την είσοδό τους στην τράπεζα. Για την προσομοίωση να χρησιμοποιηθούν οι κλάσεις “Τράπεζα” και “Ταμείο”.

Η κλάση “Τράπεζα”

- έχει έναν μετρητή που αυξάνει κάθε φορά που μπαίνει ένας πελάτης και η τιμή του αντιστοιχεί στον αριθμό προτεραιότητας του πελάτη αυτού (`last_customer`)—είναι, δηλαδή, ο αριθμός που εκτυπώνεται στο “χαρτάκι ” που παίρνει ο πελάτης όταν μπαίνει.
- έχει μια ένδειξη που αντιστοιχεί στον αριθμό προτεραιότητας του πελάτη που καλείται να εξυπηρετηθεί (`curr_serving`) —είναι ο αριθμός που εμφανίζεται, συνήθως, σε ένα φωτεινό πίνακα. Σημειώνεται ότι είναι πιθανόν, σε μια δεδομένη χρονική στιγμή, να μην υπάρχει πελάτης στην τράπεζα. Ωστόσο ο αριθμός `curr_serving` είναι ο αριθμός προτεραιότητας του πελάτη που θα κληθεί να εξυπηρετηθεί, έστω και αν αυτός δεν έχει έρθει ακόμα.

- έχει πέντε ταμεία (`cashiers`)

Η κλάση “Τράπεζα” χαρακτηρίζεται από την εξής συμπεριφορά:

- αρχικά ανοίγει το πρώτο ταμείο, η τράπεζα δεν έχει κανένα πελάτη και αρχικοποιεί την ένδειξη για τον αριθμό προτεραιότητας, `last_customer`, σε 1.
- όταν εισέρχεται ένας νέος πελάτης (`enter`), ελέγχεται αν τα ανοικτά ταμεία επαρκούν για την εξυπηρέτηση των πελατών που περιμένουν. Αν ναι, αυξάνεται η τιμή του μετρητή των αριθμών προτεραιότητας. Αν όχι, ανοίγει το επόμενο κλειστό ταμείο και μετά αυξάνεται η τιμή του μετρητή των αριθμών προτεραιότητας. Αν δεν υπάρχει τέτοιο ταμείο, απλώς τυπώνεται το μήνυμα "Sorry you cannot enter until some customers are served!".
- η εξυπηρέτηση ενός πελάτη (`serve`) αντιστοιχεί στην εξής διαδικασία.
Ελέγχεται μήπως όλα τα ταμεία είναι κλειστά οπότε ανοίγει το πρώτο ταμείο και το επιλέγει. Αλλιώς, επιλέγεται το επόμενο ταμείο που είναι ανοικτό και ελεύθερο, έστω το i_{th} . Η επιλογή γίνεται κυκλικά. (Δηλαδή, το i_{th} ταμείο είναι το επόμενο ανοικτό ταμείο από αυτό που εξυπηρέτησε την προηγούμενη φορά. Αν αυτό ήταν το τελευταίο, επιλέγεται το πρώτο ανοικτό.) (Υπόδειξη: μήπως χρειάζεστε κάποιο επιπλέον μέλος-δεδομένο για την υλοποίηση αυτή;).
Κατόπιν, τυπώνονται τα εξής: το μήνυμα "Customer no:", ακολουθεί η τιμή της ένδειξης `curr_serving`, ακολουθεί το μήνυμα "by cashier no:" και τέλος εκτυπώνεται η τιμή i . Στη συνέχεια, το επιλεγμένο ταμείο εξυπηρετεί τον πελάτη και κατόπιν ελευθερώνεται. Στη συνέχεια πραγματοποιείται έξοδος πελάτη.
- η έξοδος πελάτη (`exit`) αντιστοιχεί στην αύξηση της τιμής της ένδειξης για το ποιός αριθμός προτεραιότητας καλείται να εξυπηρετηθεί και γίνεται έλεγχος πλεονάσματος ταμείου. Αν τα ταμεία πλεονάζουν, κλείνει το πρώτο ανοικτό ταμείο –για να μη γίνεται “σπατάλη” ταμείων.
- ο έλεγχος επάρκειας ταμείων (`check_to_open`) γίνεται ελέγχοντας αν η διαφορά των αριθμών (`last_customer – curr_serving`) είναι μεγαλύτερος από τον αριθμό των ανοικτών ταμείων επί έναν αριθμό K .
- ο έλεγχος πλεονάσματος ταμείου (`check_to_close`) γίνεται ελέγχοντας αν (`last_customer – curr_serving`) είναι μικρότερος ή ίσος από τον αριθμό των ανοικτών ταμείων μειωμένο κατά ένα, επί τον αριθμό K .
- ανοίγει ένα ταμείο (`open`), ανοίγοντας το ταμείο αυτό και εκτυπώνοντας ένα μήνυμα με το ποιο ταμείο άνοιξε.
- κλείνει ένα ταμείο (`close`), κλείνοντας το ταμείο αυτό και εκτυπώνοντας ένα μήνυμα με το ποιο ταμείο έκλεισε.
- ανακοινώνει τον αριθμό των πελατών που περιμένουν να εξυπηρετηθούν (`waiting_customers`), δηλαδή τη διαφορά των αριθμών (`last_customer – curr_serving`).
- ανακοινώνει τον αριθμό των ανοικτών ταμείων (`open_cashiers`).

Η κλάση “Ταμείο”

- έχει μια ένδειξη αν είναι ανοικτό ή κλειστό (`open_flag`)
- έχει μια ένδειξη αν εξυπηρετεί πελάτη ή όχι (`serving_flag`)
- έχει έναν μετρητή για το πόσους πελάτες έχει εξυπηρετήσει χωρίς διακοπή (`customers_served`)

Η κλάση “Ταμείο” χαρακτηρίζεται από την εξής συμπεριφορά:

- κατά την κατασκευή του είναι κλειστό και άδειο.
- ανοίγει (`open`), ενημερώνοντας τη σχετική ένδειξη και μηδενίζοντας τον μετρητή `customers_served`.
- κλείνει (`close`), ενημερώνοντας τη σχετική ένδειξη.
- εξυπηρετεί πελάτη (`serve`), ενημερώνοντας τη σχετική ένδειξη.
- ελευθερώνεται (`free`), ενημερώνοντας τη σχετική ένδειξη και αυξάνοντας τον μετρητή `customers_served` κατά ένα. Αν η τιμή του γίνει ίση με μια σταθερά L , θεωρείται ότι το ταμείο υπερλειτουργήσε, οπότε κλείνει και εκτυπώνεται σχετικό μήνυμα.
- ελέγχεται αν το ταμείο είναι ανοικτό (`is_open`).
- ελέγχεται αν το ταμείο είναι ελεύθερο (`is_free`).

Να υλοποιήσετε τις παραπάνω κλάσεις σε C++. Επίσης να υλοποιηθεί και μια συνάρτηση `main` που

(α') Να δημιουργεί μια τράπεζα.

(β') Για κάθε N πελάτες που εισέρχονται στην τράπεζα αυτή να εξυπηρετεί όσους είναι δυνατό με βάση τα ταμεία που είναι ανοικτά (αυτό εξαρτάται από το πλήθος N των πελατών που μπήκαν). Η διαδικασία αυτή να επαναλαμβάνεται M φορές.

(γ') Στη συνέχεια να εξυπηρετούνται όλοι οι υπόλοιποι πελάτες που περιμένουν να εξυπηρετηθούν.

Τα K , M , N και L είναι παράμετροι της προσομοίωσης και να δίδονται κατά την κλήση του προγράμματος από τη γραμμή εντολής, με αυτή τη σειρά.

Σημείωση: Στην υλοποίηση των ασκήσεων της Α' Ομάδας, ο μόνος τύπος από τύπους βιβλιοθήκης που επιτρέπεται να χρησιμοποιήσετε είναι ο τύπος `string`.