

8η Προγραμματιστική Εργασία

Στόχοι: Εξοικείωση με τους δείκτες και τους δυναμικούς πίνακες.

Άσκηση 1:

Να υλοποιήσετε ένα πρόγραμμα, το οποίο ζητά από το χρήστη να δώσει δύο ακέραιους αριθμούς στην είσοδο. Το πρόγραμμα σας θα πρέπει να αποθηκεύσει τις τιμές που θα δώσει ο χρήστης στις ακέραιες μεταβλητές `a` και `b` αντίστοιχα.

Στο πρόγραμμα σας θα πρέπει να υπάρχουν και δύο ακέραιοι δείκτες με τα ονόματα `ptrA` και `ptrB`. Το πρόγραμμα θα πρέπει να εκτυπώσει τις τιμές των μεταβλητών `a` και `b` στο τερματικό χρησιμοποιώντας τους δείκτες `ptrA` και `ptrB` και μόνο αυτούς.

Άσκηση 2:

Υλοποιήστε την κλάση `DynamicStringArray` η οποία προσομοιώνει έναν δυναμικό πίνακα αλφαριθμητικών (`string`). Ο δυναμικός πίνακας πρέπει να αυξάνει το μέγεθός του αυτόματα όταν δεν υπάρχει επαρκής χώρος για να εισαχθεί νέα τιμή. Η κλάση πρέπει να περιλαμβάνει τα ακόλουθα ιδιωτικά πεδία:

- `int capacity;` μεταβλητή για το μέγιστο μέγεθος του πίνακα
- `int numberOfElements;` μεταβλητή για τον αριθμό των αποθηκευμένων στοιχείων
- `string *dynamicArray;` (δυναμικός) πίνακας για την αποθήκευση των δεδομένων

Αντικείμενα της κλάσης `DynamicStringArray` θα πρέπει να υποστηρίζουν τις παρακάτω δημόσιες μεθόδους:

- `DynamicStringArray()`, κατασκευαστής που αρχικοποιεί τον πίνακα με χωρητικότητα 2, χωρίς αποθηκευμένα στοιχεία.
- `int size() const`, επιστρέφει τον αριθμό των αποθηκευμένων στοιχείων στον πίνακα.
- `bool empty() const`, επιστρέφει `true` αν ο πίνακας είναι κενός, διαφορετικά `false`.
- `string at(int i) const`, επιστρέφει το στοιχείο στη θέση `i`. Αν η θέση `i` είναι εκτός ορίων (αρνητική ή μεγαλύτερη-ίση από `numberOfElements`), επιστρέφει μήνυμα σφάλματος.
- `void push_back(const string &s)`, προσθέτει το αλφαριθμητικό `s` στο τέλος του πίνακα και αυξάνει τη μεταβλητή `numberOfElements`. Αν ο πίνακας δεν έχει επαρκή χωρητικότητα, διπλασιάζει το μέγεθός του καλώντας τη μέθοδο `extend()`.

Επίσης, αντικείμενα της κλάσης `DynamicStringArray` θα πρέπει να υποστηρίζουν τις παρακάτω ιδιωτικές μεθόδους:

- `bool isValidIndex(int i) const`, επιστρέφει `true` αν το `i` είναι εντός ορίων, διαφορετικά `false`.
- `void extend()`, διπλασιάζει τη χωρητικότητα του πίνακα, μεταφέροντας τα υπάρχοντα στοιχεία στον νέο πίνακα.

Στόχος της άσκησης είναι να υλοποιήσουμε το παιχνίδι της κρεμάλας, για την υλοποίηση του οποίου θα χρειαστούμε δύο αρχεία τα οποία σας δίνονται μερικώς συμπληρωμένα:

Το `DynamicStringArray.cpp` που αφορά στην υλοποίηση της κλάσης που περιγράφηκε παραπάνω, στο οποίο έχει υλοποιηθεί ο κατασκευαστής και η ιδιωτική μέθοδος `extend`.

Το `Hangman.cpp` που περιλαμβάνει την υλοποίηση του παιχνιδιού. Στο αρχείο αυτό περιλαμβάνεται η υλοποίηση του μεγαλύτερου μέρους του παιχνιδιού, το οποίο θα πρέπει να μελετήσετε προσεκτικά και να προσπαθήσετε να κατανοήσετε.

Στη συνέχεια, θα πρέπει να επεκτείνετε τη λειτουργικότητα του παιχνιδιού ως εξής:

Όταν ο χρήστης δίνει ένα γράμμα το οποίο δεν αντιστοιχεί σε κάποιο της λέξης, τότε θα πρέπει να του εμφανίζεται ένα μήνυμα λάθους και να αποθηκεύει το γράμμα αυτό σε μια λίστα. Δηλαδή, θα πρέπει να υπάρχει μια (δεύτερη) λίστα με τα γράμματα τα οποία έχει μαντέψει ο χρήστης και δεν περιέχονται στη λέξη. Επίσης, πριν ο χρήστης μαντέψει το επόμενο γράμμα, να του εμφανίζονται τα γράμματα που ήδη έχει μαντέψει και δεν περιέχονται στη λέξη, ώστε να μην δίνει πάλι το ίδιο γράμμα.