

Δομές Δεδομένων και Τεχνικές Προγραμματισμού (Τμήμα Αρτίων ΑΜ)

Εργασία 1 (ανακοινώθηκε στις 24 Φεβρουαρίου 2020, προθεσμία παράδοσης: 29 Μαρτίου 2020, 12 τα μεσάνυχτα)

10% του συνολικού βαθμού στο μάθημα. Άριστα=120 μονάδες

1. Γράψτε μια αναδρομική συνάρτηση η οποία παίρνει σαν όρισμα μια συνδεδεμένη λίστα L και υπολογίζει το μήκος της. Το μήκος $length(L)$ μιας λίστας L ορίζεται να είναι ο αριθμός των στοιχείων της. Θεωρήστε ότι ο τύπος λίστα ορίζεται από τα `typedef` της σελίδας 27 της πρώτης ενότητας διαλέξεων (Συνδεδεμένες Αναπαραστάσεις Δεδομένων). Θα πρέπει να γράψετε και κατάλληλο κύριο πρόγραμμα το οποίο να χρησιμοποιείτε για να δείξετε τη λειτουργικότητα της συνάρτησης σας.

(10 μονάδες)

2. Θεωρείστε το πρόγραμμα που παρουσιάστηκε στην πρώτη ενότητα διαλέξεων (Συνδεδεμένες Αναπαραστάσεις Δεδομένων) και βρίσκεται στην ιστοσελίδα του μαθήματος: <http://cgi.di.uoa.gr/~k08/code.html>. Το πρόγραμμα υλοποιεί μια συνδεδεμένη λίστα πληροφοριών για αεροπορικές πτήσεις και μερικές ενδιαφέρουσες λειτουργίες συνδεδεμένων λιστών. Γράψτε μια συνάρτηση `Select` η οποία θα παίρνει σαν όρισμα μία λίστα L και ένα ακέραιο i ($1 \leq i \leq length(L)$) και θα επιστρέφει ένα δείκτη στο i -στό στοιχείο της λίστας L . Επίσης γράψτε μια συνάρτηση `Replace` η οποία θα παίρνει σαν όρισμα μία λίστα L , ένα ακέραιο i ($1 \leq i \leq length(L)$) και ένα κωδικό αεροδρομίου Y , και θα αντικαθιστά τον κωδικό αεροδρομίου του i -στού στοιχείου της λίστας με το Y . Να τροποποιήσετε κατάλληλα το κύριο πρόγραμμα που σας δίνεται ώστε να τεστάρετε τη λειτουργία των συναρτήσεων σας.

(10 μονάδες)

3. Υποθέστε ότι έχετε στη διάθεση σας δύο στοίβες. Πως μπορείτε να τις χρησιμοποιήσετε για να υλοποιήσετε τη λειτουργικότητα μια ουράς αναμονής; Υλοποιήστε ένα κατάλληλο αφηρημένο τύπο δεδομένων με το

όνομα `Queue`. Η υλοποίησή σας πρέπει να τροποποιεί την υλοποίηση ουράς με συνδεδεμένη λίστα που υπάρχει στην ενότητα «Ουρές» του κώδικα από την ιστοσελίδα του μαθήματος. Παρομοίως, οι στοίβες που θα χρησιμοποιήσετε θα πρέπει να υλοποιούνται από τον κώδικα υλοποίησης στοίβας με συνδεδεμένη λίστα που βρίσκεται στην ενότητα «Στοιίβες» του κώδικα από την ιστοσελίδα του μαθήματος.

(20 μονάδες)

4. Ένα παλίνδρομο είναι μια φράση που διαβάζεται το ίδιο από την αρχή προς το τέλος και από το τέλος προς την αρχή. Ένα παράδειγμα παλίνδρομου στα Αγγλικά είναι "Madam, I'm Adam". Γράψτε ένα πρόγραμμα που θα χρησιμοποιεί μια στοίβα για να ελέγξει αν μια δοσμένη συμβολοσειρά είναι παλίνδρομο ή όχι. Το πρόγραμμά σας πρέπει να χρησιμοποιεί την υλοποίηση στοίβας με συνδεδεμένη λίστα που βρίσκεται στην ενότητα «Στοιίβες» του κώδικα από την ιστοσελίδα του μαθήματος. Χρησιμοποιείστε το αρχείο `palindromes.txt` που δίνεται για να τεστάρετε το πρόγραμμά σας.

(20 μονάδες)

5. Στο ερώτημα αυτό θα λύσουμε το πρόβλημα πλοήγησης ενός ρομπότ σε ένα λαβύρινθο. Το πρόβλημα αυτό ορίζεται στην ιστοσελίδα <https://www.cs.bu.edu/teaching/alg/maze/> όπου επίσης δίνεται μια αναδρομική λύση. Μελετήστε προσεκτικά αυτή τη ιστοσελίδα. Μετά γράψτε μια συνάρτηση η οποία να χρησιμοποιεί επανάληψη (loop) και μια στοίβα για να λύσει ένα δοσμένο πρόβλημα λαβύρινθου. Η υλοποίηση της στοίβας που θα χρησιμοποιήσετε θα πρέπει να είναι η (κατάλληλα τροποποιημένη) υλοποίηση στοίβας με συνδεδεμένη λίστα που δόθηκε στην ενότητα «Στοιίβες» του μαθήματος και η οποία βρίσκεται στην ιστοσελίδα του μαθήματος.

Υπόδειξη: Η στοίβα θα χρησιμοποιείται για να αποθηκεύουμε τις **καταστάσεις** στις οποίες μπαίνει το ρομπότ καθώς κινείται στο λαβύρινθο (θα είναι μια **στοίβα καταστάσεων**). Η κατάσταση που βρίσκεται το ρομπότ μας δίνεται από τις συντεταγμένες της θέσης στην οποία βρίσκεται. Όταν μια θέση δεν οδηγεί πουθενά τότε οπισθοδρομούμε και συνεχίζουμε από τη θέση από

την οποία περάσαμε πιο πρόσφατα και η οποία μπορεί να βρεθεί στην κορυφή της στοίβας (αυτή είναι η χρήση της στοίβας). Οι υπόλοιπες λεπτομέρειες είναι όπως στην αναδρομική λύση, μόνο που εμείς θα χρησιμοποιήσουμε επανάληψη (loop).

(60 μονάδες)

Παρατηρήσεις:

- Οι λύσεις σας στα ερωτήματα 3-5 πρέπει να είναι οργανωμένες σε modules της C όπως έχουμε συζητήσει στο μάθημα
- Τα προγράμματα σας θα πρέπει να είναι όσο πιο καλά οργανωμένα γίνεται, σύμφωνα με όσα έχετε μάθει στο μάθημα Εισαγωγή στον Προγραμματισμό.

Πως να παραδώσετε τις λύσεις σας: Οι λύσεις θα πρέπει να σταλούν στο e-mail ddproj@di.uoa.gr μέχρι την προθεσμία παράδοσης και να είναι οργανωμένες ως εξής. Θα στείλετε ακριβώς ένα συμπιεσμένο αρχείο με το όνομα `Askisi1-ΌνομαΕρwhnyto.rar` ή `.zip` σαν attachment όπου `Όνομα` και `Ερwhnyto` είναι το όνομα και το επώνυμο σας με λατινικούς χαρακτήρες. Στο body του e-mail σας θα υπάρχει μόνο το όνομα σας γραμμένο στα Ελληνικά και ο Αριθμός Μητρώου σας. Η λύση κάθε προβλήματος θα είναι σε χωριστό directory το οποίο θα περιλαμβάνει τα source files που χρειάζονται, και ένα Makefile το οποίο θα μπορεί να χρησιμοποιηθεί για να μεταγλωττιστούν τα αρχεία σας και να παραχθεί το αντίστοιχο executable. Σε κάθε directory θα υπάρχει και ένα αρχείο pdf (όχι Word) το οποίο θα περιέχει όσο documentation χρειάζεται ώστε οι βαθμολογητές να κατανοήσουν πλήρως τη λύση σας και να τη βαθμολογήσουν ανάλογα. Αν δεν συμπεριλάβετε τέτοιο αρχείο pdf, θα χάσετε βαθμούς. Ο κώδικας που θα υποβάλλετε θα ελεγχθεί με κατάλληλο λογισμικό ώστε να βρεθούν πιθανές ομοιότητες με λύσεις που θα υποβάλλουν οι συνάδελφοι σας. Αν διαπιστωθούν περιπτώσεις αντιγραφής, θα μηδενιστούν όλα τα εμπλεκόμενα

μέρη χωρίς να δοθούν περαιτέρω εξηγήσεις και χωρίς να γίνει καμία σχετική συζήτηση.