

Εργαστήριο #6
24/11/2015

I. ΑΡΧΕΙΑ

Στο αρχείο `data.txt` υπάρχει μια σειρά από ακεραίους αριθμούς. Ο πρώτος από αυτούς τους αριθμούς καθορίζει πόσοι ακόμα αριθμοί υπάρχουν στο αρχείο. Αν για παράδειγμα αυτός ο αριθμός είναι ο 4, τότε ακολουθούν άλλοι 4 ακέραιοι αριθμοί στο αρχείο.

- α) Να γραφεί πρόγραμμα το οποίο ανοίγει αυτό το αρχείο, διαβάζει τα δεδομένα και τα αποθηκεύει σε έναν κατάλληλο πίνακα ακεραίων που θα δημιουργηθεί δυναμικά με την χρήση της `malloc()`. Στην συνέχεια το πρόγραμμα θα πρέπει να αποθηκεύσει τα δεδομένα που διάβασε σε δύο διαφορετικά αρχεία: το αρχείο `positive.txt` και το αρχείο `negative.txt`. Στο αρχείο `positive.txt` θα αποθηκευτούν οι θετικοί αριθμοί από αυτούς που διάβασε και στο αρχείο `negative.txt` θα αποθηκευτούν οι αρνητικοί αριθμοί. Τόσο το αρχείο `positive.txt` όσο και το αρχείο `negative.txt` να διατηρήσουν τη μορφοποίηση του αρχικού αρχείου, δηλαδή στην πρώτη γραμμή κάθε αρχείου θα πρέπει να υπάρχει το πλήθος των αριθμών που ακολουθούν.
- β) Να προστεθεί κώδικας ο οποίος επαναφέρει τη θέση του αρχείου `positive.txt` στην αρχή, ξαναδιαβάζει τους αριθμούς από το αρχείο `positive.txt` και υπολογίζει το άθροισμά τους.

II. STRUCTS

Θέλουμε να φτιάξουμε ένα πρόγραμμα βαθμολογίου για μία τάξη. Για κάθε μαθητή πρέπει να φυλάμε όνομα, αριθμό μητρώου και τη βαθμολογία του. Αυτό που σας ζητείται είναι:

1. Να διαβάσετε το πλήθος των μαθητών και τα στοιχεία τους από το πληκτρολόγιο.
2. Να αποθηκεύσετε τα στοιχεία χρησιμοποιώντας μία δομή (`struct`) ανά μαθητή.
3. Με χρήση των δομών, να υπολογίσετε τον μέσο βαθμό.
4. Με χρήση των δομών, να τυπώσετε τα ονόματα των μαθητών που κόπηκαν (δηλαδή είχαν βαθμολογία κάτω του 10).
5. Να αλλάχουν τα ονόματα όσων κόπηκαν ώστε το τέλος να υπάρχει το «(failed)». Π.χ. ο «Vasilios Vasiliou» να γίνει «Vasilios Vasiliou (failed)».

Υλοποιήστε τα ζητούμενα χρησιμοποιώντας έναν **πίνακα από δομές**. Θα πρέπει να δηλώσετε τη δομή, να δεσμεύσετε **δυναμικά** τον πίνακα από δομές και να αποθηκεύσετε στον πίνακα αυτόν τα στοιχεία των μαθητών. Τέλος θα πρέπει να κάνετε και τις αντίστοιχες λειτουργίες χρησιμοποιώντας τα υποδείγματα των συναρτήσεων όπως σας δίνονται στο αρχείο `skel1.c`.

III. STRUCTS + ΑΡΧΕΙΑ

Πρέπει να επαναλάβετε το Πρόβλημα II, όπου όμως τώρα τα στοιχεία των μαθητών θα πρέπει να φυλάσσονται σε αρχείο. Συγκεκριμένα:

- Κατά την έξοδο του προγράμματος, τον πίνακα από δομές τον αποθηκεύετε σε ένα αρχείο κειμένου `students.txt` (αποθηκεύετε ένα-ένα τα πεδία, σε ξεχωριστές γραμμές, με `fprintf()`). Το πρώτο πράγμα που πρέπει να αποθηκεύσετε είναι το πλήθος των μαθητών.
- Κατά την επόμενη εκκίνηση του προγράμματός σας, πρέπει να ελέγχετε αν υπάρχει το αρχείο `students.txt`. Αν υπάρχει, θα πρέπει να το διαβάσετε και να φτιάξετε τις δομές από τα περιεχόμενά του (θυμηθείτε ότι στην αρχή του αρχείου υπάρχει το πλήθος των μαθητών). Αλλιώς, θα τα διαβάσετε από το πληκτρολόγιο, όπως στο Πρόβλημα II.

IV. ΕΠΙΠΛΕΟΝ ΕΞΑΣΚΗΣΗ: STRUCTS + ΔΥΝΑΜΙΚΗ ΛΙΣΤΑ 1

Υλοποιήστε το Πρόβλημα II χρησιμοποιώντας μία **λίστα από δομές** (αντί για πίνακα).

Θα πρέπει να δηλώσετε τη δομή και να συμπεριλάβετε και ένα πεδίο `next`, το οποίο θα δείχνει στον επόμενο κόμβο της λίστας. Επίσης, θα πρέπει να δηλώσετε και μία μεταβλητή `head`, η οποία θα είναι δείκτης με αρχική τιμή `NULL` (άδεια λίστα) και στη συνέχεια θα δείχνει πάντα στη δομή που μπήκε πιο πρόσφατα στη λίστα. Κάθε κόμβος της λίστας θα πρέπει να δεσμεύεται δυναμικά και θα εισάγεται στο σημείο που δείχνει η μεταβλητή `head`.

Τέλος θα πρέπει να κάνετε και τις αντίστοιχες λειτουργίες χρησιμοποιώντας τα υποδείγματα των συναρτήσεων όπως σας δίνονται στο αρχείο `skel2.c`.

V. ΕΠΙΠΛΕΟΝ ΕΞΑΣΚΗΣΗ: STRUCTS + ΔΥΝΑΜΙΚΗ ΛΙΣΤΑ 2

Κάντε πάλι το προηγούμενο πρόβλημα αλλά, αυτή τη φορά, κάθε κόμβος να εισάγεται στο **τέλος** και όχι στην **αρχή** της λίστας.