

Γενικά σχόλια lab11

Για τη μεταγλώττιση, στοίχιση και ονόματα ισχύει ότι και για προηγούμενα εργαστήρια.

Άσκηση 1:

Αρχικοποίηση πίνακα: Εφόσον είναι πίνακας από δείκτες, είναι λογικό να αρχικοποιηθούν τα στοιχεία του σε NULL. Αυτό μπορεί να γίνει κατά τη δήλωση: `char p_sentences[SENTENCES] = {NULL};`

Μέγεθος fgets: Η δεύτερη παράμετρος της fgets πρέπει να είναι SIZE, εφόσον η fgets εξ'ορισμού διαβάζει μέχρι και SIZE-1 χαρακτήρες ώστε να βάλει το '\0' στην τελευταία θέση.

Ανίχνευση ορίων προτάσεων: Εφόσον επιτρέπεται να αλλαχθεί η συμβολοσειρά, ο πιο πρακτικός τρόπος να ανιχνεύσουμε τις προτάσεις είναι

1. κράτα την αρχή της πρότασης (αρχικά η αρχή όλου του κειμένου)
2. βρες την πρώτη τελεία που ακολουθεί (χρήση strchr)
3. αντικατέστησε τη με '\0'
4. χρησιμοποίησε είτε strdup ή συνδυασμό malloc/strcpy για να φτιάξεις ένα αντίγραφο της συμβολοσειράς (από την αρχή της πρότασης μέχρι το '\0'). Είτε χρησιμοποιηθεί strdup είτε malloc, πρέπει να γίνει έλεγχος για το αν επέστρεψε NULL ή όχι. **Προσοχή:** σε περίπτωση χρήσης malloc, πρέπει να δεσμευθεί ακριβώς `strlen(sentence)+1` μνήμη, όπου sentence η πρόταση που εντοπίσαμε. Το +1 είναι απαραίτητο για να χωρέσει και το '\0'. Αυτό ήταν το πιο συχνό λάθος που έκαναν όσοι χρησιμοποίησαν malloc.
5. θέσε την αρχή της πρότασης στη διεύθυνση ακριβώς μετά την τελεία που βρήκες στο βήμα 2
6. επανέλαβε τα βήματα 2-6 μέχρι να μην υπάρχουν άλλες τελείες. Αυτό ανιχνεύεται ελέγχοντας το αποτέλεσμα της κλήσης strchr. **Προσοχή:** Πρώτα ελέγχουμε αν η strchr επέστρεψε NULL ώστε να βγούμε από το loop αν χρειάζεται, και μετά γράφουμε κώδικα που προσπελάζει τον δείκτη που επέστρεψε η strchr. Αν γίνει η προσπέλαση πριν τον έλεγχο, τότε στην περίπτωση του NULL το πρόγραμμα θα πετάξει segmentation fault.

Ένα συχνό λάθος που έγινε εδώ ήταν επανάληψη `for (i = 0; i < SIZE; i++)` για την εύρεση των τελειών. Αυτό είναι λάθος γιατί η αρχική **συμβολοσειρά τελειώνει στο '\0' και ΟΧΙ στο τέλος του πίνακα**. Αν τύχει και βρίσκονται τελείες ανάμεσα στα σκουπίδια που υπάρχουν μετά το '\0', τότε το πρόγραμμά σας δε θα λειτουργήσει σωστά.

strdup ή malloc/strcpy: Δείτε το βήμα 4 πιο πάνω για παρατηρήσεις

Εκτύπωση: Θέλουμε η εκτύπωση να γίνει αφού ολοκληρωθούν τα παραπάνω. Το loop πρέπει να σταματά όταν βρεθεί NULL (βλ. αρχικοποίηση) ή τουλάχιστον να παρακάμπτει τα NULL. Εναλλακτικά, μπορείτε να μετρήσετε πόσες προτάσεις έχουν εισαχθεί και να τρέξετε το loop μόνο τόσες φορές.

free: Αφού ολοκληρωθεί και η εκτύπωση, διατρέχουμε τον πίνακα και κάνουμε free ένα-ένα στοιχείο του, δηλαδή `free(p_sentences[i]);`. Καλό είναι να αποφύγουμε να κάνουμε free τα στοιχεία που είναι NULL. Δεν είναι λάθος, αλλά είναι άσκοπο. Αν δεν έχει γίνει αρχικοποίηση, είναι σίγουρα λάθος να προσπαθήσουμε να κάνουμε free στοιχεία του πίνακα που δε δείχνουν σε έγκυρες προτάσεις (και θα οδηγήσει σε segmentation fault).

Άσκηση 2:

prototype add_book: Οι τύποι των παραμέτρων πρέπει να είναι `bookInfoT [], int *` και `char *`. Η συνάρτηση επιστρέφει `int`.

αλγόριθμος αναζήτησης: Η συνάρτηση διατρέχει τον πίνακα (όχι μέχρι `MAX_BOOKS` μέχρι το πλήθος βιβλίων που έχουν αποθηκευτεί σε αυτόν). Αν βρει το βιβλίο που ψάχνουμε, αυξάνει τον αριθμό αντιτύπων και επιστρέφει άμεσα. ΔΕΝ ΕΙΝΑΙ ΣΩΣΤΟ να βάλετε `else` σε αυτό το σημείο. Αν η επανάληψη τελειώσει χωρίς να βρεθεί το βιβλίο, τότε πρέπει να γίνει έλεγχος για το αν χωράει, κι αν ναι, να εισαχθεί.

εισαγωγή/malloc: Όπως πριν, θέλουμε είτε `strdup` για το `isbn` ή συνδυασμό `malloc/strcpy` φροντίζοντας η `malloc` να δεσμεύσει ακριβώς όσο χώρο χρειάζεται (χωρίς να ξεχνάμε κι ένα επιπλέον `byte` για το `'\0'`). Ο αριθμός αντιτύπων γίνεται 1 και το πλήθος βιβλίων πρέπει να αυξηθεί.

επιστρεφόμενη τιμή: 0 αν αποτύχει η `malloc` ή η `strdup`, ή αν ο πίνακας ήταν γεμάτος. Διαφορετικά 1.

main: scanf: Για τη `scanf` που διαβάζει το `isbn`

main: κλήσεις συναρτήσεων: Η `init` πρέπει να κληθεί πριν το `do-while`. Η `print` στην περίπτωση 'p' και η `add_books` στην 'a' αφού διαβαστεί το `isbn` από το χρήστη. Το αποτέλεσμα της `init` πρέπει να αποθηκευτεί σε κατάλληλη μεταβλητή, τη διεύθυνση της οποίας περνάμε στην `add_books` (πλήθος βιβλίων).

main: αποτέλεσμα add_books: Ελέγχεται ώστε να εκτυπωθεί το κατάλληλο μήνυμα σε κάθε περίπτωση

main: free: Μπορεί να γίνει και σε συνάρτηση. Πρέπει να διατρέξουμε τον πίνακα και για κάθε έγγυρο στοιχείο να κάνουμε `free` το `isbn`: `free(books[i].isbn);`