

Εργαστήριο #3
1/11/2016

I. ΕΥΡΕΣΗ ΣΤΟΙΧΕΙΟΥ (ΔΕΥΤΕΡΗ ΕΚΔΟΧΗ)

Επαναλάβετε την 3η Άσκηση του προηγούμενου εργαστηρίου, όπου όμως η συνάρτηση `search()` δεν επιστρέφει τίποτα (θα είναι `void`). Επομένως, θα πρέπει να έχει μία τέταρτη παράμετρο, και αυτή θα χρησιμοποιείται για να πάρουμε δείκτη προς το στοιχείο του πίνακα που αναζητούμε.

II. ΜΙΚΡΟ ΛΕΞΙΚΟ

Υλοποιήστε ένα μικρό λεξικό το οποίο φυλάει ένα σύνολο από γνωστές λέξεις. Το λεξικό θα είναι ένας πίνακας από δείκτες που δείχνουν σε κάποιες σταθερές συμβολοσειρές της αρεσκείας σας. Κατά τη διάρκεια της εκτέλεσης, ο χρήστης θα δίνει συνεχώς λέξεις και το πρόγραμμά σας θα ελέγχει αν υπάρχουν ή όχι στο λεξικό. Αν δοθεί η λέξη "fin", το πρόγραμμά σας πρέπει να τερματίσει.

III. ΠΑΡΑΜΕΤΡΟΙ ΣΤΗ `main()`

Υλοποιήστε ένα πρόγραμμα-κομπιουτεράκι, το οποίο χρησιμοποιεί παραμέτρους στην `main()` και λειτουργεί ως εξής:

- Διαβάζει το πρώτο όρισμα που του δίνεται και το οποίο δηλώνει την πράξη που πρέπει να κάνει. Υποστηρίζονται: + (πρόσθεση), - (αφαίρεση), * (πολλαπλασιασμός), / (διαίρεση).
- Διαβάζει όλα τα υπόλοιπα ορίσματα του και εκτελεί την δεδομένη πράξη με αυτά.

Για παράδειγμα, η εκτέλεση:

```
$ ./a.out / 10 2
```

πρέπει να τυπώσει 5, ενώ το:

```
$ ./a.out + 2 4 6 8
```

πρέπει να τυπώσει 20.

IV. ΣΤΑΤΙΣΤΙΚΑ ΓΡΑΜΜΑΤΩΝ

Δημιουργήστε μία συνάρτηση `charstats()`, η οποία υπολογίζει πόσες φορές εμφανίζεται κάθε γράμμα του Αγγλικού αλφαβήτου σε μία συμβολοσειρά η οποία θα δίνεται ως παράμετρος. Συγκεκριμένα, θα πρέπει να υπάρχει ένας πίνακας 26 θέσεων, όπου φυλάσσεται ένας μετρητής για κάθε γράμμα. Κάθε φορά που στη δοθείσα συμβολοσειρά εμφανίζεται ένα γράμμα (είτε μικρό είτε κεφαλαίο), θα πρέπει να αυξάνεται ο αντίστοιχος μετρητής.

Σημειώσεις:

- Ο πίνακας με τους μετρητές θα πρέπει να είναι καθολικός (global).
- Η `main()` θα πρέπει να διαβάζει συμβολοσειρές από το πληκτρολόγιο με χρήστη της `fgets()` και να καλεί τη συνάρτηση `charstats()`, μέχρι να δοθεί συμβολοσειρά που ξεκινά με τρία «καγκελάκια» (###). Στο τέλος να εκτυπώσει τα στατιστικά.

V. ΕΠΙΠΛΕΟΝ ΕΞΑΣΚΗΣΗ ΜΕ ΤΟΝ GDB

Εξασκηθείτε μόνοι σας περισσότερο με τον gdb. Θα σας βοηθήσει πάρα πολύ στη συνέχεια των εργαστηρίων.

VI. ΕΠΙΠΛΕΟΝ ΕΞΑΣΚΗΣΗ: ΕΡΩΤΗΜΑ ΣΤΗ ΣΕΛ. 28

Προσπαθήστε να απαντήσετε στο ερώτημα της σελίδας 28 των διαφανειών (04-strings.pdf), δηλαδή να φτιάξετε μία συνάρτηση: `void int_to_string(int num, char *string);`

Η συνάρτηση θα πρέπει να δέχεται έναν ακέραιο, να τον μετατρέπει σε συμβολοσειρά και να αποθηκεύει τους χαρακτήρες της στην παράμετρο `string`, η οποία υποτίθεται ότι δείχνει σε αρκετό χώρο ώστε να χωράει το μέγιστο δυνατό πλήθος ψηφίων.

Σημείωση: μπορείτε να κάνετε οποιαδήποτε υλοποίηση, όμως υπάρχει μία ιδιαίτερα απλή, η οποία απαιτεί μόλις 1 εντολή!

VII. ΕΠΙΠΛΕΟΝ ΕΞΑΣΚΗΣΗ: ΕΟΡΤΟΛΟΓΙΟ

Φτιάξτε ένα μικρό δικό σας εορτολόγιο. Συγκεκριμένα, φτιάξτε δύο πίνακες όπου ο ένας θα φυλάει την ημερομηνία και ο άλλος τις αντίστοιχες ονομαστικές εορτές (ο δεύτερος θα είναι πίνακας με δείκτες σε σταθερές συμβολοσειρές). Ο χρήστης θα πρέπει να δίνει, μέσω ορισμάτων στην `main()`, είτε το όνομα είτε την ημερομηνία και θα του τυπώνει την αντίστοιχη ημερομηνία ή τη γιορτή. Η αναζήτηση του ονόματος να γίνεται με χρήση της `strstr()` μιας και την ίδια ημέρα μπορεί να γιορτάζουν πολλά ονόματα.