

lab10: Δομές

Προετοιμασία:

1. Πλοηγηθείτε στον κατάλογο lab10 ο οποίος πρέπει να βρίσκεται μέσα στον κατάλογο ce120.
2. Προσθέστε κώδικα στο lab10.c
3. Θα πρέπει στον ίδιο κατάλογο με το lab10.c να βρίσκονται τα lab10.h και libsocial.a
4. Η εντολή για να κάνετε compile είναι:

```
gcc -Wall -g lab10.c -o lab10 -lsocial -L.
```

Θέματα εργασίας:

- struct
- enum
- Συναρτήσεις
- Συμβολοσειρές

Αποστολή Προγράμματος για σχολιασμό και βαθμολόγηση

- Πλοηγηθείτε στον κατάλογο **lab10**. Θα πρέπει να βρίσκεται μέσα το αρχείο **lab10.c**
- Κατασκευάστε να κατάλογο με όνομα lab10_επώνυμο1_AEM1_επώνυμο2_AEM2 (ΜΕ ΛΑΤΙΝΙΚΟΥΣ ΧΑΡΑΚΤΗΡΕΣ) όπου επώνυμο1, AEM1 αναφέρονται στο ένα μέλος της ομάδας και επώνυμο2, AEM2 στο δεύτερο μέλος. Για παράδειγμα, αν τα δύο μέλη της ομάδας ήταν οι Ντουφεξή με AEM 01234 και Θάνος με AEM 05678, θα γράφατε:

```
mkdir lab10_Doufexi_01234_Thanos_05678
```

- Αντιγράψτε το αρχείο C μέσα στον κατάλογο. lab10_επώνυμο1_AEM1_επώνυμο2_AEM2
- Πακετάρετε τον κατάλογο (compress here as tar.gz).
- Το ένα μέλος της ομάδας:
 - μπαίνει στο λογαριασμό του στο eclass,
 - πηγαίνει στην ενότητα Εργασίες,
 - κάνει click στην εργασία lab10 για το συγκεκριμένο τμήμα
 - **ΠΡΟΣΟΧΗ!!!!** Στο πεδίο Ομάδα Χρηστών επιλέγει τη σωστή ομάδα (με βάση τα ονόματα των μελών)
 - Ανεβάζει το .tar.gz αρχείο μέσω του Browse **και πατά Αποθήκευση.**
 - **ΠΡΟΣΟΧΗ!!!!** Επιβεβαιώνει ότι ανέβασε το σωστό αρχείο.

Άσκηση

Το πρόγραμμα που θα γράψετε για την άσκηση πρέπει να το αποθηκεύσετε στο αρχείο που σας δίνεται με όνομα **lab10.c** μέσα στον κατάλογο **lab10**. Επιπλέον, σας δίνεται έτοιμο το αρχείο **lab10.h** και η βιβλιοθήκη **libsocia1.a** την οποία δε μπορείτε να διαβάσετε. Για να μεταγλωττίσετε αρκεί να γράψετε

```
gcc -Wall -g lab10.c -o lab10 -lsocial -L.
```

Στην άσκηση αυτή σας δίνεται ο σκελετός ενός προγράμματος το οποίο εξετάζει τις συνδέσεις των χρηστών ενός κοινωνικού δικτύου. Για κάθε χρήστη αποθηκεύουμε το όνομά του και ποιοι τον ακολουθούν (follow).

users[0]	name: B o b \0	followers: NO YES YES NO NO	Οι followers του Bob είναι η Alice και ο James.
users[1]	name: A l i c e \0	followers: YES NO YES YES NO	Οι followers της Alice είναι ο Bob ο James και η Mary-Anne
users[2]	name: J a m e s \0	followers: YES YES NO NO NO	Οι followers του James είναι η Alice και ο Bob.
users[3]	name: M a r y - A n n e \0	followers: NO NO YES NO NO	Οι followers της Mary-Anne είναι μόνο ο James.
users[4]	name: \0	followers: NO NO NO NO NO	Αυτή η θέση δεν αντιστοιχεί σε χρήστη.

Σας δίνονται έτοιμα:

- [στο αρχείο lab10.h] τύπος enum με όνομα **is_follower_t** και τιμές NO και YES
- [στο αρχείο lab10.h] τύπος struct με όνομα **user_t** που περιγράφει ένα χρήστη του κοινωνικού δικτύου και παρέχει τα πεδία:
 - **name**: πίνακας χαρακτήρων μεγέθους NAME_SIZE που αποθηκεύει το όνομα του χρήστη.
 - **followers**: πίνακας **is_follower_t** μεγέθους MAX_USERS που περιέχει YES στις θέσεις που αντιστοιχούν σε followers του χρήστη, NO στις υπόλοιπες.
- [στο αρχείο lab10.c, main] **users**: πίνακας **user_t** μεγέθους MAX_USERS στον οποίο αποθηκεύονται τα στοιχεία όλων των χρηστών όπως φαίνεται στο παραπάνω σχήμα.
- [στη βιβλιοθήκη libsocia1.a] Η συνάρτηση **init_users_table** η οποία λαμβάνει ως είσοδο ένα πίνακα από **user_t** και προσθέτει ένα αριθμό από χρήστες σε αυτόν. Σε όσες θέσεις δεν υπάρχουν χρήστες, το πεδίο **name** αρχικοποιείται στην κενή συμβολοσειρά και όλες οι θέσεις του πίνακα **followers** λαμβάνουν την τιμή **NO**.
- [στο αρχείο lab10.c, main] μερική υλοποίηση της **main** στην οποία δημιουργείται και αρχικοποιείται ο πίνακας **users** με τυχαίες τιμές. Επίσης παρέχεται ένα αρχικό μενού. Καλείστε να συμπληρώσετε τη συνάρτηση **main** όπου απαιτείται καλώντας συναρτήσεις που θα κατασκευάσετε εσείς.

Τι πρέπει να κάνετε εσείς:

1. Γράψτε μία συνάρτηση η οποία λαμβάνει ως παράμετρο ένα πίνακα από `user_t` που περιέχει τους χρήστες του κοινωνικού δικτύου και εκτυπώνει στην οθόνη τα περιεχόμενα του πίνακα. Για κάθε έγκυρο χρήστη εκτυπώνει το όνομά του, άνω κάτω τελεία, τα ονόματα των followers του με τη σειρά που εμφανίζονται στον πίνακα με ένα κενό πριν από κάθε όνομα, και στο τέλος αλλαγή γραμμής.
2. Στη `main` καλέστε την παραπάνω συνάρτηση στην επιλογή `L` ή `I` ώστε να εκτυπώνονται τα περιεχόμενα του πίνακα.
3. Γράψτε μία συνάρτηση η οποία λαμβάνει ως παραμέτρους τον πίνακα χρηστών και ένα δείκτη σε ακέραιο και έχει επιστρεφόμενο τύπο `int`. Η συνάρτηση υπολογίζει το χρήστη με τους περισσότερους followers, θέτει το πλήθος των followers στη διεύθυνση που δείχνει ο δείκτης-παράμετρος και επιστρέφει την θέση αυτού του χρήστη. Σε περίπτωση που δύο ή περισσότεροι χρήστες έχουν τον ίδιο αριθμό followers επιλέγει τον χρήστη που βρίσκεται πιο κοντά στην αρχή του πίνακα.
4. Στη `main` καλέστε την παραπάνω συνάρτηση στην επιλογή `P` ή `p` και μετά εκτυπώστε αλλαγή γραμμής, το μήνυμα `Most popular: N with X followers` όπου `N` το όνομα του χρήστη και `X` το πλήθος του followers που έχει, και αλλαγή γραμμής.

Ιδέα για νέα λειτουργία, αφού στείλετε την άσκηση:

Προσθέστε μια συνάρτηση (και αντίστοιχη επιλογή μενού) η οποία θα παίρνει το όνομα ενός χρήστη `X` και τον πίνακα χρηστών και θα του προτείνει νέα άτομα που μπορεί να ακολουθήσει. Αν ο `X` ακολουθεί ένα χρήστη `Y` και ο `Y` ακολουθεί ένα χρήστη `Z`, τότε, αν ο `X` δεν ακολουθεί ήδη τον `Z`, η συνάρτηση προτείνει τον `Z`.