

## lab10: Δομές

### Πριν ξεκινήσετε...

Πατήστε πάνω στο σύνδεσμο "Download handout" και κατεβάστε το αρχείο lab10.tar.gz στον υπολογιστή σας. Στη συνέχεια ανοίξτε το αρχείο lab10.tar.gz με το ποντίκι και σύρετε (drag & drop) τον κατάλογο lab10 μέσα στο φάκελο ce120 που βρίσκεται στο home directory σας. Θα δουλέψετε μέσα στον κατάλογο lab10 που μόλις μεταφέρατε.

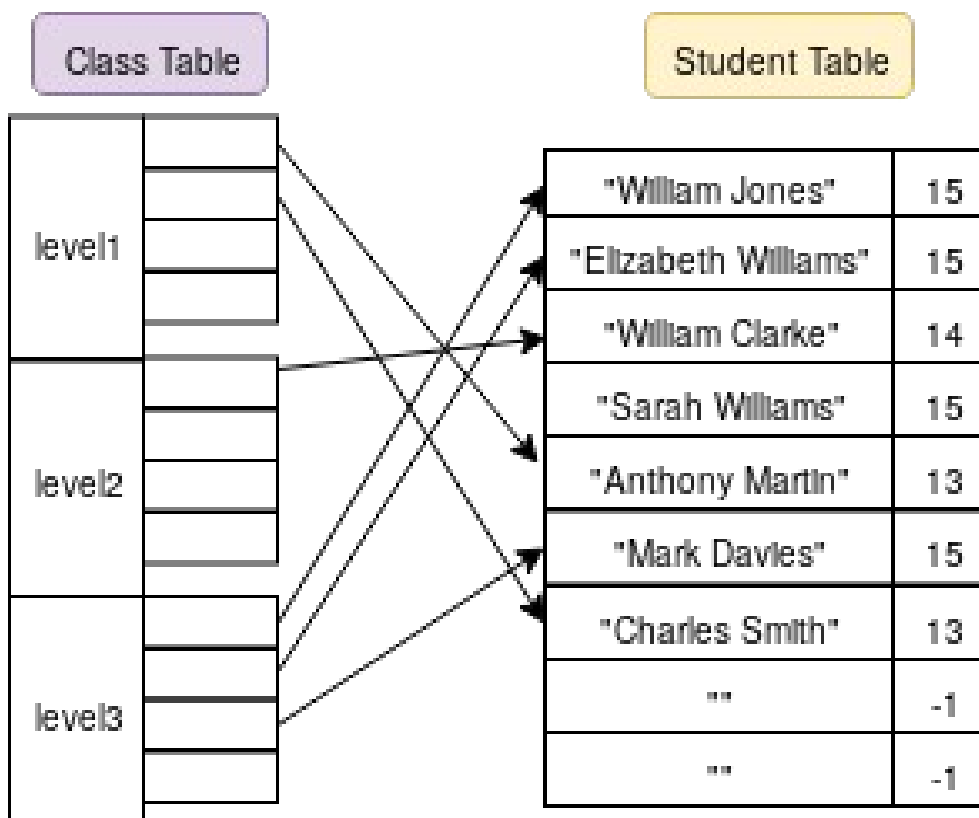
### Διαχωριστικό string

Όπου σας λέμε να εκτυπώσετε το **διαχωριστικό string**, εισάγετε μια `printf("\n#\n");`

### Υποβολή ασκήσεων

1. **Πριν παραδώσετε τις ασκήσεις βεβαιωθείτε ότι είναι σωστή η ομάδα σας στο Autolab κι αν δεν είναι, φτιάξτε τη ξανά!**
2. **Αντιγράψτε** το `lab10a.c` στον κατάλογο `lab10submit` που βρίσκεται μέσα στο `lab10`.
  - Δεν πρέπει να υπάρχουν άλλα αρχεία σε αυτόν τον κατάλογο.
3. Κάντε **δεξί κλικ** στον κατάλογο `lab10submit` και επιλέξτε Compress → Here as tar.gz
4. Θα έχει δημιουργηθεί ένα αρχείο με όνομα `lab10submit.tar.gz` το οποίο ένα μέλος της ομάδας (δεν έχει σημασία ποιο) **υποβάλει** στο Autolab.

**Προσοχή:** Τα προγράμματά σας πρέπει να είναι πάντα σωστά στοιχισμένα, να περιέχουν κατάλληλα σχόλια, να έχουν περιγραφικά ονόματα μεταβλητών/συναρτήσεων και σωστούς τύπους, και η έξοδός τους να συμμορφώνεται πλήρως με τις προδιαγραφές.



## Άσκηση 1

Προσθέστε τον κώδικά σας στο έτοιμο αρχείο με όνομα **lab10a.c** που βρίσκεται στον κατάλογο lab10 που προέκυψε από την αποσυμπίεση του lab10.tar.gz.

Η μεταγλώττιση γίνεται με την εντολή `gcc -Wall -g lab10a.c -o lab10a -llab10 -l.`

Διαβάστε όλη την εκφώνηση και δείτε τα περιεχόμενα των lab10.h και lab10a.c πριν ξεκινήσετε. Σας δίνονται έτοιμες οι συναρτήσεις `generate_students`, `print_students`, `print_class`. Οι υλοποιήσεις τους βρίσκονται στη στατική βιβλιοθήκη `liblab10.a` και τα `prototypes` τους στο lab10.h

Στην παρούσα άσκηση θα γράψετε ένα πρόγραμμα το οποίο κατατάσσει σε τάξεις μια σειρά μαθητών, τα στοιχεία των οποίων είναι αποθηκευμένα σε ένα πίνακα. Οι μαθητές αποθηκεύονται σε μεταβλητές τύπου `struct student` (το `struct` σας δίνεται έτοιμο στο αρχείο lab10.h).

Το πρόγραμμα που θα γράψετε θα κάνει τα εξής:

1. Δηλώνει ένα πίνακα τύπου `class_t` μεγέθους `MAX_CLASSES` και ένα πίνακα τύπου `student_t` μεγέθους `MAX_STUDENTS`. Ο πρώτος πίνακας αντιπροσωπεύει τις τάξεις που θα δημιουργηθούν και ο δεύτερος πίνακας το σύνολο των μαθητών που θα κατανεμηθούν στις τάξεις.
2. Ελέγχει τον αριθμό των ορισμάτων που λαμβάνονται από τη γραμμή εντολών. Εάν δεν λαμβάνεται σωστό πλήθος ορισμάτων, εκτυπώνεται το μήνυμα "**Incorrect arguments!**" ακολουθούμενο από **χαρακτήρα αλλαγής γραμμής** και τερματίζει. Το πρόγραμμα παίρνει ως όρισμα το συνολικό πλήθος μαθητών.
3. Καλεί την έτοιμη συνάρτηση `generate_students` για την αρχικοποίηση του πίνακα των μαθητών που δημιουργήσατε στο βήμα 1. Η δεύτερη παράμετρος της συνάρτησης αντιπροσωπεύει τον αριθμό των μαθητών που τελικά θα δημιουργηθούν και η τιμή της είναι αυτή που ελήφθη ως όρισμα προγράμματος. Η συνάρτηση δημιουργεί μία λίστα από μαθητές των οποίων η ηλικία είναι από 13 έως και 15 ετών.
4. Εκτυπώνει τους μαθητές που δημιουργήθηκαν με τη βοήθεια της έτοιμης συνάρτησης `print_students`. Στις τελευταίες γραμμές η συνάρτηση εκτυπώνει τον αριθμό των μαθητών που ανήκουν σε κάθε ηλικιακή ομάδα.
5. Γράψτε μια συνάρτηση η οποία λαμβάνει ως παραμέτρους α) έναν πίνακα τύπου `class_t` και β) το μέγεθος του και χρησιμοποιήστε τη για να αρχικοποιήσετε τον πίνακα των τάξεων (τύπου `class_t`) που ορίσατε στο βήμα 1, ώστε η τιμή του πεδίου `enum` στην πρώτη θέση να είναι `level1`, στη δεύτερη θέση `level2` και στην τρίτη θέση `level3`. Θα πρέπει να έχετε υπόψη σας ότι 1η θέση του πίνακα αντιστοιχεί σε τάξη με μαθητές ηλικίας 13 ετών, η 2η θέση σε τάξη με μαθητές 14 ετών και η τρίτη θέση σε τάξη με μαθητές 15 ετών.

**Σημείωση:** Δεν θα πρέπει να παραλείψετε να αρχικοποιήσετε κατάλληλα τους πίνακες δεικτών που περιέχονται στον πίνακα τύπου `class_t`.

Καλέστε τη συνάρτηση που γράψατε για να αρχικοποιήσετε τις τάξεις.

6. Δείτε το σχήμα της προηγούμενης σελίδας στο οποίο έχουν προστεθεί δείκτες στον πίνακα μαθητών κάθε μίας τάξης οι οποίοι δείχνουν στους μαθητές της αντίστοιχης ηλικίας. Προσθέστε κώδικα στη `main` ο οποίος κάνει αυτή την αντιστοίχιση για κάθε τάξη.
7. Για κάθε τάξη, καλέστε τη συνάρτηση `print_class` η οποία παίρνει ως παράμετρο ένα στοιχείο του πίνακα τάξεων (δηλαδή ένα στοιχείο τύπου `class_t`) και εκτυπώνει τα περιεχόμενα της τάξης.
8. (Προαιρετικά για όσους τελειώσουν νωρίς) Γράψτε μια συνάρτηση `my_print_class` η οποία υλοποιεί τη λειτουργία της `print_class` και χρησιμοποιήστε αυτή στο πρόγραμμά σας στη θέση της `print_class`. Η συνάρτηση εκτυπώνει **χαρακτήρα αλλαγής γραμμής**, το μήνυμα "**== levelX ==**" για X ίσο 1 ή 2 ή 3 ανάλογα με την τάξη, και **χαρακτήρα αλλαγής γραμμής**. Ακολουθούν τα ονόματα των μαθητών της τάξης, ένα ανά γραμμή. Μετά το τελευταίο όνομα υπάρχει μόνο ένας χαρακτήρας αλλαγής γραμμής.