

Εργαστήριο #6: Συναρτήσεις

Αρχικά αρχεία

Στο αρχείο `lab6.h` περιλαμβάνονται ενδεικτικοί ορισμοί των σταθερών που πρέπει να χρησιμοποιήσετε στο πρόγραμμα:

- `MAX_STUDENTS`: Μέγιστο πλήθος φοιτητών σε ένα πανεπιστημιακό τμήμα.
- `NUM_COURSES`: Πλήθος υποχρεωτικών μαθημάτων σε ένα πανεπιστημιακό τμήμα.
- `APPROXIMATION_ERROR`: Αριθμός που αναπαριστά πόσο κοντά πρέπει να βρίσκονται δύο πραγματικοί αριθμοί κατ' απόλυτη τιμή ώστε να θεωρούνται ίσοι.

Θα γράψετε το πρόγραμμα σας στο αρχείο `lab6.c`. Το αρχείο αυτό περιέχει ήδη τη γραμμή `#include "lab6.h"` που χρειάζεται ώστε να συμπεριληφθούν οι ορισμοί των παραπάνω σταθερών στο πρόγραμμα. Βεβαιωθείτε πως το `lab6.h` (που σας δίνεται έτοιμο και μπορείτε να αλλάξετε για να κάνετε δοκιμές με διαφορετικά μεγέθη) βρίσκεται πάντα στον ίδιο κατάλογο με το `lab6.c`.

Γενικές απαιτήσεις

Το πρόγραμμα σας πρέπει να είναι σωστά στοιχισμένο, να περιέχει κατάλληλα σχόλια και να έχει περιγραφικά ονόματα μεταβλητών και συναρτήσεων.

Χρησιμοποιήστε την κατάλληλη δομή για κάθε είδος επανάληψης και φροντίστε οι συναρτήσεις σας να ακολουθούν τις προδιαγραφές.

Η έξοδος πρέπει να συμμορφώνεται πλήρως με τις προδιαγραφές. Σημειώστε πως στα μηνύματα που εκτυπώνονται υπάρχει ακριβώς ένα κενό (`space`) ανάμεσα σε διαδοχικές λέξεις και πάντα χαρακτήρας αλλαγής γραμμής στο τέλος.

Απαγορεύεται αυστηρά η χρήση `goto` και η χρήση καθολικών μεταβλητών.

Υποβολή ασκήσεων

Υποβάλετε το αρχείο `lab6.c` στο `autolab`.

Άσκηση

Σε αυτή την άσκηση θα γράψετε ένα πρόγραμμα που διαβάζει τους βαθμούς που πήραν οι φοιτητές ενός πανεπιστημιακού τμήματος στα υποχρεωτικά μαθήματα και, για έναν συγκεκριμένο φοιτητή, υπολογίζει πόσοι άλλοι φοιτητές είχαν καλύτερους, χειρότερους και ίδιους βαθμούς με αυτόν.

Ορίστε στη συνάρτηση `main` ένα δισδιάστατο πίνακα πραγματικών αριθμών με `MAX_STUDENTS` γραμμές και `NUM_COURSES` στήλες στον οποίο πρόκειται να αποθηκευτούν για κάθε φοιτητή του τμήματος (ανά γραμμή) οι βαθμοί του σε κάθε μάθημα (ανά στήλη).

Γράψτε μια συνάρτηση `αρχικοποίησης` που παίρνει ως παραμέτρους τον πίνακα βαθμών και έναν δείκτη προς ακέραιο. Η συνάρτηση εκτυπώνει για κάθε έναν φοιτητή το μήνυμα "Enter grades for the next student (y/n/q)?\n" και διαβάζει έναν χαρακτήρα. Το μήνυμα και η ανάγνωση επαναλαμβάνονται μέχρι να δοθεί ένας από τους χαρακτήρες 'y', 'n', 'q'. Αν δοθεί 'y' κι εφόσον υπάρχει χώρος στον πίνακα βαθμών, τότε διαβάζει τους βαθμούς του επόμενου φοιτητή για κάθε μάθημα και τους αποθηκεύει στη σειρά του πίνακα που αντιστοιχεί στον φοιτητή. Υποθέστε ότι πάντα δίνονται στην είσοδο του προγράμματος ακριβώς τόσοι βαθμοί όσα είναι τα μαθήματα και ότι οι τιμές είναι έγκυρες. Η ανάγνωση βαθμών ολοκληρώνεται με επιτυχία αν γεμίσει ο πίνακας (διαβαστούν οι βαθμοί του τελευταίου φοιτητή) ή αν δοθεί απάντηση 'n' στο παραπάνω ερώτημα, ή με αποτυχία αν δοθεί απάντηση 'q' στο παραπάνω ερώτημα ή δεν έχουν εισαχθεί βαθμοί στον πίνακα. Αν η ανάγνωση ολοκληρωθεί με επιτυχία, τότε αποθηκεύεται στον ακέραιο που δείχνει ο δείκτης που δόθηκε ως δεύτερη παράμετρος το πλήθος φοιτητών για το οποίο διαβάστηκαν βαθμοί και η συνάρτηση επιστρέφει 0. Διαφορετικά, η συνάρτηση επιστρέφει -1.

Προσθέστε κώδικα στη συνάρτηση `main` που καλεί τη συνάρτηση αρχικοποίησης. Εάν η συνάρτηση επιστρέψει -1, εκτυπώνεται το μήνυμα "Program exited.\n" και το πρόγραμμα τερματίζει επιστρέφοντας 1.

Βεβαιωθείτε ότι το πρόγραμμά σας λειτουργεί σωστά έως αυτό το σημείο πριν συνεχίσετε.

Γράψτε μια συνάρτηση που παίρνει ως παραμέτρους δύο πραγματικούς αριθμούς κι επιστρέφει μια ακέραια τιμή. Η τιμή που επιστρέφεται είναι 0 αν η απόλυτη τιμή της διαφοράς των παραμέτρων είναι μικρότερη από ή ίση με το `APPROXIMATION_ERROR` (το οποίο σημαίνει ότι θεωρούνται ίσοι). Αν η απόλυτη τιμή της διαφοράς τους είναι μεγαλύτερη από το `APPROXIMATION_ERROR`, τότε η συνάρτηση επιστρέφει -1 αν ο πρώτος είναι μικρότερος του δεύτερου ή 1 αν ο πρώτος είναι μεγαλύτερος του δεύτερου.

Γράψτε μια συνάρτηση που παίρνει ως παραμέτρους (1) τον πίνακα βαθμών, (2) το πλήθος φοιτητών οι βαθμοί των οποίων έχουν αποθηκευτεί στον πίνακα, (3) τη θέση (γραμμή) ενός φοιτητή στον πίνακα, (4) τη θέση (στήλη) ενός μαθήματος στον πίνακα, (5) δείκτη προς ακέραιο, και (6) δείκτη προς ακέραιο. Η συνάρτηση υπολογίζει το πλήθος φοιτητών που έχουν γράψει μικρότερο βαθμό στο δεδομένο μάθημα από τον δεδομένο φοιτητή και το αποθηκεύει στον ακέραιο που δείχνει η 5η παράμετρος. Επιπλέον, υπολογίζει το πλήθος φοιτητών που έχουν γράψει μεγαλύτερο βαθμό στο δεδομένο μάθημα από τον δεδομένο φοιτητή και το αποθηκεύει στον ακέραιο που δείχνει η 6η παράμετρος. Ο έλεγχος για το αν ο βαθμός είναι μεγαλύτερος ή μικρότερος γίνεται μέσω κλήσης στη συνάρτηση που γράψατε προηγουμένως. Η συνάρτηση δεν επιστρέφει κάτι.

Προσθέστε κώδικα στη συνάρτηση `main` που εκτυπώνει το μήνυμα "Which student?\n" και διαβάζει τη θέση ενός φοιτητή στον πίνακα. Όσο δίνεται θέση στην οποία δεν έχουν αποθηκευτεί βαθμοί, η εκτύπωση και ανάγνωση επαναλαμβάνονται. Μετά, εκτυπώνει το μήνυμα "Which course?\n" και διαβάζει τη θέση ενός μαθήματος στον πίνακα. Όσο δίνεται θέση εκτός των ορίων του πίνακα, η εκτύπωση και ανάγνωση επαναλαμβάνονται. Στη συνέχεια, καλεί την προηγούμενη συνάρτηση κι εκτυπώνει το μήνυμα:

```
Better: X
Worse: Y
Same: Z
```

όπου X το πλήθος φοιτητών που πήγαν καλύτερα από τον επιλεγμένο φοιτητή, Y το πλήθος φοιτητών που πήγαν χειρότερα και Z το πλήθος άλλων φοιτητών που έγραψαν το ίδιο.

Μετά την εκτύπωση, το πρόγραμμά τερματίζει επιστρέφοντας 0.