

Προγραμματιστικές Ασκήσεις, Φυλλάδιο 3: Files and Structs

Εκφώνηση: 21 Μαΐου 2012

Παράδοση: Πέμπτη, 07/06/12

Άσκηση 0

Να μετατρέψετε την άσκηση της σειράς 2, που αφορά την αποθήκευση και ανάλυση στατιστικών δεδομένων, έτσι ώστε να λειτουργεί με structs. Το struct των στατιστικών δεδομένων ανά χώρα(struct statisticalData) θα περιλαμβάνει:

1. Το όνομα της χώρας (char name[30]).
2. Το πλήθος των ετών (int num_years) για τα οποία έχουμε στατιστικά για την χώρα
3. Έναν πίνακα που αποθηκεύει τα έτη για τα οποία έχουμε στατιστικά (int *years).
4. Από ένα πίνακα, μεγέθους όσο το πλήθος των ετών, για τα ακόλουθα στοιχεία: (α) Πληθωρισμό (float *inflation), (β) αύξηση του ΑΕΠ (float *aep_growth), (γ) ποσοστό επενδύσεων (float *investments)
5. Από ένα float, για τα ακόλουθα στοιχεία: Μέσος, ελάχιστος, μέγιστος πληθωρισμός (float avg_inflation, min_inflation, max_inflation)
6. Από ένα float, για τα ακόλουθα στοιχεία: Μέση, ελάχιστη, μέγιστη αύξηση του ΑΕΠ
7. Από ένα float, για τα ακόλουθα στοιχεία: Μέσο, ελάχιστο, μέγιστο ποσοστό επενδύσεων

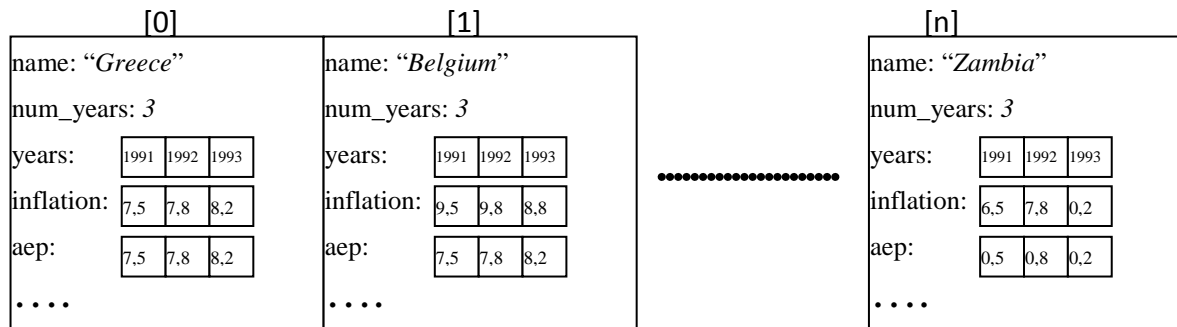
Για παράδειγμα, ο ορισμός του σχετικού struct μπορεί να είναι ως εξής:

```
struct statisticalData{
    char name[30];
    int num_years;
    int *years;
    float *inflation, *aep_growth, *investments;
    float avg_inflation, min_inflation, max_inflation;
    float avg_aep_growth, min_aep_growth, max_aep_growth;
    float avg_investmens, min_investmens, max_investmens;
};
```

/ για λόγους συντομίας, έχουμε μαζέψει πολλά πεδία σε μία γραμμή – μπορείτε να έχετε ένα πεδίο ανά γραμμή */*

Η κάθε χώρα θα αποθηκεύεται σε κατάλληλη θέση ενός πίνακα από struct `statisticalData`, ο οποίος θα ορίζεται μέσα στη `main` ως:

```
struct statisticalData * all_data;
```



Δομή αρχείων

Οι χώρες παραμένουν οι ίδιες με αυτές της προγραμματιστικής εργασίας 2. Στο

<http://www.cs.uoi.gr/~cs423/exercises/hw3/>

υπάρχουν τρία αρχεία, ένα για την αύξηση του ΑΕΠ, ένα για τον πληθωρισμό και ένα για τις επενδύσεις. Κάθε γραμμή αφορά και μία χώρα, και περιέχει μετρήσεις για 13 χρονιές, χωρισμένες με tab.

Για παράδειγμα, παρατίθεται το αρχείο `aep.ascii` με τις μετρήσεις της αύξησης του ΑΕΠ:

1.92	2.14	2.21	2.06	2.08	2.14	2.03	2.12	1.80	1.73	1.79	1.72	1.72
0.68	0.85	3.49	4.53	4.62	5.51	5.81	5.95	5.61	4.69	4.81	4.44	3.55
0.20	0.68	0.83	0.72	0.67	0.58	0.73	1.28	1.16	1.11	1.54	1.74	2.10
3.38	3.47	3.32	3.23	3.05	2.75	2.82	2.63	1.83	1.29	1.03	0.72	0.59
-1.42	-1.43	-1.90	-2.76	-3.54	-3.87	-2.99	-3.29	-3.56	-4.01	-2.52	-2.89	-1.66

Θεωρούμε γνωστό ότι έχουμε 5 χώρες και 13 χρονιές,. Παρ' όλ' αυτά, όλοι οι πίνακες πλην των συμβολοσειρών θα κατασκευαστούν με `malloc`.

Βασικές λειτουργίες

Η `main()` θα φορτώνει τα περιεχόμενα του αρχείου σε ένα πίνακα από struct `statisticalData` και θα υπολογίζει όποια πεδία χρήζουν υπολογισμού (μέσες, μέγιστες και ελάχιστες τιμές). Μετά, θα καλείται ένα μενού για να επιτρέπει στον χρήστη να εργαστεί με τα δεδομένα.

Σε κάθε σημείο που χρειάζεται, θα πρέπει ΥΠΟΧΡΕΩΤΙΚΑ να υπάρχουν οι σχετικοί έλεγχοι καθώς και τα σχετικά διαγνωστικά μηνύματα για το αν κάτι πήγε στραβά.

1. Δέσμευση μνήμης για τον κεντρικό πίνακα από struct `statisticalData`, καθώς και για τους πίνακες από `float` για κάθε κελί του κεντρικού πίνακα, με κατάλληλες εντολές `malloc`.
2. Εισαγωγή δεδομένων από τα σχετικά αρχεία (αν η εισαγωγή γίνει επιτυχώς, μπορείτε να κλείσετε τη σύνδεση με τα αρχεία). Για το σκοπό της εισαγωγής

δεδομένων, θα καλέσετε μια συνάρτηση `int loadData(... συμπληρώστε ...)` η οποία θα επιστρέφει 0 αν όλα πήγαν καλά, αλλιώς θα εξέρχεται του προγράμματος τυπώνοντας ένα διαγνωστικό μήνυμα.

3. Υπολογισμός μέσων, μεγίστων και ελαχίστων τιμών: Θα υλοποιήσετε μια συνάρτηση `calcStats(...)` που θα παίρνει όρισμα ένα πίνακα από `struct statisticalData`, καθώς και όποιο άλλο όρισμα χρειάζεται, και θα υπολογίζει τις μέγιστες, ελάχιστες και μέσες ποσότητες για κάθε χώρα.

Στη συνέχεια, η `main()` θα καλεί ένα μενού, μέσω του οποίου, ο χρήστης μπορεί να επιλέξει κάτι από τα παρακάτω:

4. Εκτύπωση μέσων, μεγίστων και ελαχίστων τιμών ανά χώρα. Κατασκευάστε μια συνάρτηση `void printXML(...)`, η οποία θα παίρνει όρισμα ένα πίνακα από `struct statisticalData`, καθώς και όποιο άλλο όρισμα χρειάζεται, και θα εκτυπώνει σε ένα αρχείο `summaryData.xml` τα στατιστικά των χωρών (μέσες, μέγιστες ελάχιστες τιμές) για πληθωρισμό, ΑΕΠ, επενδύσεις, με τον εξής τρόπο:

```
<stats>
  <country> όνομα χώρας
    <inflation>
      <avg_inflation> μέση τιμή για τη χώρα </avg_inflation>
      <min_inflation> ελάχιστη τιμή για τη χώρα </min_inflation>
      <max_inflation> μέγιστη τιμή για τη χώρα </max_inflation>
    </inflation>
    ... ομοίως και για τα άλλα τρία μεγέθη ...
  </country>
  ... ομοίως και για τα άλλα 4 κράτη ...
</stats>
```

Ότι είναι με πλάγια γράμματα είναι τιμή που πρέπει να συμπληρώσετε. Τα υπόλοιπα είναι σταθερά, τούτέστιν, τα γράφετε ακριβώς όπως τα βλέπετε (και στοιχισμένα).

/ Δοκιμάστε να ανοίξετε το αρχείο xml που φτιάξατε με κάποιο πρόγραμμα πλοήγησης Διαδικτύου. */*

5. Υπολογισμός συμμεταβλητότητας: Το πρόγραμμα θα ζητάει από το χρήστη το όνομα μιας χώρας. Στη συνέχεια θα καλεί μια συνάρτηση `allInOne(...)`, η οποία
 - (α) καλεί μια βοηθητική συνάρτηση `findCountry(...)` που αναζητά στον πίνακα `all_data` αν υπάρχει η χώρα αυτή και επιστρέφει τη θέση της, αν υπάρχει, ή -1 αν δεν υπάρχει
 - (β) καλεί μια βοηθητική συνάρτηση που θα υπολογίζει τη συμμεταβλητότητα για τα εξής στοιχεία:
 - i. inflation vs. aep_growth

- ii. inflation vs. investments
- iii. aep_growth vs. investments

και ενημερώνει με πέρασμα κατ' αναφορά τρεις σχετικές μεταβλητές. Για να το δυσκολέψουμε κι άλλο: η εκτύπωση γίνεται από τη main, άρα πρέπει να ενημερωθεί και η main για τις τρεις αυτές τιμές.

6. Έξοδος

Άσκηση 1

Υποθέστε ότι το πρόγραμμά σας για την προηγούμενη άσκηση, φτιάχνεται από διαφορετικούς προγραμματιστές. Το παρακάτω πινακάκι δείχνει ποιος φτιάχνει τι σε ποιο αρχείο.

Θαλής (συντονιστής)	main() και ότι άλλο χρειάζεται	makeMyDay.c + εγκρίνει ότι αρχείο .h χρειάζεται + το makefile
Αναξιμένης	loadData(...)	loadData.c
Αναξίμανδρος	calcStats(...)	calcStats.c
Ηράκλειτος	printXML(...)	printXML.c
Παρμενίδης	Όλες οι ρουτίνες της συμμεταβλητότητας	convol.c

Χωρίς να πειράξετε τον κώδικά σας (εκτός αν πραγματικά υπάρχει κάποια ανάγκη), σπάστε τον σε αρχεία .c και .h και χρησιμοποιήστε ένα **makefile** για να μεταφράσετε το πρόγραμμά σας. Αποφασίστε τι αρχεία **.c και (κυρίως) .h** χρειάζονται με βάση το παραπάνω σχήμα.

Κάντε turnin όλα τα .c, .h, αρχεία, καθώς και το makefile.