ΕΡΓΑΣΤΗΡΙΟ 5

Θέμα: Πίνακες

Απαιτήσεις/Διευκρινίσεις:

- Απαγορεύονται **αυστηρά** η χρήση goto και η χρήση πινάκων μεταβλητού μεγέθους.
- Χρησιμοποιήστε την καταλληλότερη δομή επανάληψης για κάθε περίπτωση.
- Τα προγράμματά σας πρέπει να είναι σωστά στοιχισμένα, να περιέχουν κατάλληλα σχόλια και να έχουν περιγραφικά ονόματα μεταβλητών και σταθερών. Ακολουθήστε τις υποδείξεις των εκφωνήσεων lab1-lab3 και του φυλλαδίου με τίτλο "**Οδηγός γραφής ευανάγνωστου κώδικα**" που βρίσκεται στο eclass, στην ενότητα Έγγραφα->Φυλλάδια/Εγχειρίδια.
- Η έξοδος πρέπει να συμμορφώνεται πλήρως με τις προδιαγραφές. Σημειώστε πως στα μηνύματα κειμένου που εκτυπώνονται υπάρχει ακριβώς ένα κενό (space) ανάμεσα σε διαδοχικές λέξεις και πάντα χαρακτήρας αλλαγής γραμμής στο τέλος.
- Διαβάστε τις υποδείξεις/διευκρινίσεις κάθε άσκησης πριν ξεκινήσετε να γράφετε κώδικα!

Αρχικά αρχεία

Σας δίνουμε το αρχείο **`lab5.h`** στο οποίο περιλαμβάνονται ενδεικτικοί ορισμοί των σταθερών που πρέπει να χρησιμοποιήσετε και στα δύο προγράμματα ως μεγέθη πινάκων. Οι σταθερές είναι:

- `WORDSIZE`: μέγεθος πίνακα στον οποίο αποθηκεύεται μια λέξη ως συμβολοσειρά (άσκ. 1.)
- `MATRIXSIZE`: μέγεθος τετραγωνικού πίνακα (άσκηση 2).

Προσθέστε τη γραμμή **`#include"lab5.h"`** στην αρχή καθενός από τα `.c` αρχεία που θα γράψετε ώστε να συμπεριλάβετε αυτούς τους ορισμούς στα προγράμματά σας. Βεβαιωθείτε πως το `lab5.h` βρίσκεται πάντα στον ίδιο κατάλογο με τα αρχεία `lab5a.c, lab5b.c`.

Υποβολή ασκήσεων

Πριν υποβάλετε:

- Κατασκευάσετε εκ νέου την ομάδα σας.
- Βεβαιωθείτε ότι έχετε προσθέσει στα αρχεία C την εντολή #include "lab5.h" και ΔΕΝ έχετε προσθέσει δικές σας εντολές #define για τα WORDSIZE και MATRIXSIZE.
- Πακετάρετε τα αρχεία lab5a.c, lab5b.c και μόνο αυτά σε ένα zip με όνομα lab5.zip .
- Υποβάλετε το lab5.zip στο autolab.

ΠΡΟΣΟΧΗ: Έχετε όριο 10 υποβολών. Χρησιμοποιήστε τα αρχεία ελέγχου ώστε να επιβεβαιώσετε ότι το πρόγραμμα σας λειτουργεί σωστά ΠΡΙΝ υποβάλετε στο autolab.

Άσκηση 1

Αποθηκεύστε το πρόγραμμα σε αρχείο με όνομα `lab5a.c`.

Ζητούμενο: Γράψτε ένα πρόγραμμα που χρησιμοποιεί την απόσταση λέξεων ως κριτήριο για να προτείνει μια διόρθωση για μια τυχόν λανθασμένη λέξη. Το πρόγραμμα επιλέγει ως διόρθωση τη λέξη ενός λεξικού που έχει τη μικρότερη δυνατή απόσταση από τη λανθασμένη.

Ορισμός: Η απόσταση ανάμεσα σε δύο λέξεις ορίζεται ως το πλήθος χαρακτήρων που βρίσκονται σε αντίστοιχες θέσεις και είναι ένας προς έναν διαφορετικοί.

Στα παρακάτω παραδείγματα οι χαρακτήρες που συμμετέχουν στον υπολογισμό της απόστασης ανάμεσα στις λέξεις 1 και 2 εμφανίζονται με κόκκινο χρώμα.

Λέξη 1	baseball	baseball	baseball	baseball
Λέξη 2	soccer	hand-glider	ballpark	baseballer
Απόσταση	8	9	5	2

Το πρόγραμμα λειτουργεί ως εξής:

Εκτυπώνει το μήνυμα `"Check word:\n"` και διαβάζει την πιθανώς λανθασμένη λέξη που θέλουμε να ελέγξουμε.

Εκτυπώνει το μήνυμα `"Dictionary words:\n"` και, σε επανάληψη μέχρι να δοθεί απάντηση `'n'` στο βήμα 4:

1. Διαβάζει μία λέξη λεξικού και υπολογίζει την απόσταση της από τη "λανθασμένη" λέξη.
2. Αν η απόσταση είναι μηδέν, τότε η "λανθασμένη" λέξη βρέθηκε στο λεξικό (άρα δεν ήταν λανθασμένη!) και το πρόγραμμα εκτυπώνει το μήνυμα `"X is correct!\n"` όπου `X` η λέξη και τερματίζει.
3. Διαφορετικά, εκτυπώνει το μήνυμα `"X [D]\n"` όπου `X` η λέξη λεξικού και `D` η απόσταση ανάμεσα σε αυτή και στη λανθασμένη λέξη.
4. Εκτυπώνει το μήνυμα `"More (y/n)?\n"` και διαβάζει ένα χαρακτήρα. Όσο αυτός δεν είναι ούτε `'y'` ούτε `'n'`, επαναλαμβάνεται το βήμα 4.

Όταν ολοκληρωθεί η επανάληψη (των βημάτων 1-4), το πρόγραμμα εκτυπώνει πρώτα το μήνυμα `"Suggestion:\n##\n"` και στη συνέχεια το μήνυμα `"Y->X [D]\n"` όπου `Y` η λανθασμένη λέξη, `X` η λέξη λεξικού που έχει την ελάχιστη απόσταση (σε σχέση με όλες τις άλλες λέξεις λεξικού) από τη λανθασμένη λέξη και `D` αυτή η απόσταση.

Υποδείξεις/Διευκρινίσεις:

- Όλες οι λέξεις αποθηκεύονται ως συμβολοσειρές σε πίνακες μεγέθους `WORDSIZE`.
- Μπορείτε να υποθέσετε πώς όλες οι λέξεις που δίνονται χωράνε σε πίνακα μεγέθους `WORDSIZE`.

- Χρησιμοποιήστε τις λέξεις όπως δίνονται. Δε χρειάζεται να κάνετε έλεγχο για το αν τα γράμματα είναι μικρά ή κεφαλαία, ούτε κάποια μετατροπή. Οι λέξεις μπορεί να περιέχουν χαρακτήρες που δεν είναι γράμματα (π.χ. παύλες). Αυτό δεν επηρεάζει κάτι.
- Αν δύο λέξεις έχουν διαφορετικό μέγεθος, οι χαρακτήρες της μεγαλύτερης λέξης για τους οποίους δεν υπάρχει αντίστοιχος χαρακτήρας στην μικρότερη λέξη απλά αυξάνουν την απόσταση ανάμεσα στις δύο λέξεις.
- Το πλήθος των λέξεων λεξικού είναι άγνωστο. Κατά την ανάγνωση, χρησιμοποιήστε ένα πίνακα μεγέθους `WORDSIZE` στον οποίο αποθηκεύετε τη νέα λέξη λεξικού σε κάθε επανάληψη ("πάνω" στην προηγούμενη λέξη).
- Χρησιμοποιήστε μια ακέραια μεταβλητή για να αποθηκεύσετε την ελάχιστη απόσταση που έχει βρεθεί ανάμεσα στη λανθασμένη λέξη και μία λέξη λεξικού, κι έναν πίνακα μεγέθους `WORDSIZE` για να αποθηκεύσετε τη λέξη λεξικού που έχει αυτή την ελάχιστη απόσταση.
  Κάθε φορά που υπολογίζετε μια απόσταση μικρότερη από την ελάχιστη που έχει βρεθεί μέχρι στιγμής, ανανεώστε κατάλληλα την τιμή της ακέραιας μεταβλητής και τα περιεχόμενα του πίνακα.
- Αν υπάρχουν πολλαπλές λέξεις λεξικού με την ίδια ελάχιστη απόσταση, επιλέγεται πάντα η πρώτη που βρέθηκε.
- Δεν επιτρέπεται η χρήση συναρτήσεων από το string.h

Άσκηση 2

Αποθηκεύστε το πρόγραμμα σε αρχείο με όνομα `lab5b.c` .

Ζητούμενο: Γράψτε ένα πρόγραμμα C το οποίο ανιχνεύει αν ένας πίνακας είναι συμμετρικός ή όχι.

Το πρόγραμμα λειτουργεί ως εξής:

1. Ορίζει ένα δισδιάστατο πίνακα ακεραίων μεγέθους `MATRIXSIZE` × `MATRIXSIZE`.
2. Διαβάζει από το πληκτρολόγιο μια σειρά ακεραίων και γεμίζει τον πίνακα ανά γραμμή.
3. Αφού ολοκληρωθεί η εισαγωγή των στοιχείων του πίνακα, ελέγχει αν ο πίνακας είναι συμμετρικός ως προς την κύρια διαγώνιο.
4. Αφού ολοκληρωθεί ο έλεγχος,
    a. Αν ο πίνακας είναι συμμετρικός, εκτυπώνει το μήνυμα `"YES\n"` και στη συνέχεια εκτυπώνει το τριγωνικό κομμάτι του πίνακα κάτω από τη κύρια διαγώνιο, όπως περιγράφεται στις υποδείξεις.
    b. Διαφορετικά εκτυπώνει το μήνυμα `"NO\n"` και στη συνέχεια εκτυπώνει όλο τον πίνακα, όπως περιγράφεται στις υποδείξεις.
5. Στο τέλος, χρησιμοποιεί την εντολή `fprintf(stderr, "\n%d\n", comparisons);` όπου `comparisons` είναι μια ακέραια μεταβλητή που έχετε ορίσει εσείς και που χρησιμοποιείτε για να μετρήσετε το συνολικό πλήθος συγκρίσεων που έγιναν ανάμεσα σε στοιχεία του πίνακα μέχρις ότου να βρεθεί αν ο πίνακας είναι συμμετρικός ή όχι.

Υποδείξεις/Διευκρινίσεις:

- Ολοκληρώστε την άσκηση χωρίς χρήση επιπλέον πινάκων.
- Το πρόγραμμα σας πρέπει να περιέχει τρεις διακριτές ενότητες: Εισαγωγή δεδομένων, έλεγχος συμμετρίας, εκτύπωση αποτελεσμάτων.
- Προσπαθήστε να αναπτύξτε τον αλγόριθμό σας ώστε να εντοπίζει αν ο πίνακας είναι συμμετρικός κάνοντας όσο το δυνατό λιγότερες συγκρίσεις ανάμεσα σε στοιχεία του πίνακα.
- Η εντολή fprintf(stderr, ...) εκτυπώνει στην έξοδο σφαλμάτων. Θα δείτε το αποτέλεσμα στην οθόνη σας ακόμη κι αν κάνετε ανακατεύθυνση της συμβατικής εξόδου σε αρχείου.
- Για τις εκτυπώσεις πινάκων:
    - Η εκτύπωση γίνεται ανά γραμμή και στο τέλος κάθε γραμμής βρίσκεται χαρακτήρας αλλαγής γραμμής.
    - Πριν από κάθε στοιχείο του πίνακα εκτυπώνεται ένα κενό (space).
    - Κάθε στοιχείο εκτυπώνεται με πλάτος 4.