

Εργαστήριο #8

Απαιτήσεις/Διευκρινίσεις:

- Απαγορεύονται η χρήση goto, η χρήση gets και η χρήση καθολικών ή static μεταβλητών.
- Το πρόγραμμά σας πρέπει να είναι γραμμένο ώστε να λειτουργεί σωστά για οποιαδήποτε θετική τιμή των ορισμένων μεγεθών πινάκων.
- Για την αντιγραφή, σύγκριση, εύρεση και αντικατάσταση συμβολοσειρών ή μέρους αυτών, πρέπει υποχρεωτικά να χρησιμοποιηθούν συναρτήσεις από το string.h.

Αρχεία: Δίνονται τα αρχεία lab8.c και lab8.h. Το lab8.h περιέχει τους ορισμούς των MAX_WORDS και WORD_WIDTH, και περιλαμβάνεται στο .c αρχείο μέσω μιας εντολής #include. Μην ορίσετε αυτές τις σταθερές εκ νέου στο lab8.c.

Υποβολή: Υποβάλετε στο autolab μόνο το αρχείο lab8.c στο οποίο έχετε προσθέσει τον κώδικά σας.

Ζητούμενα: Γράψτε ένα πρόγραμμα που παίρνει μια σειρά από ορίσματα, επιλέγει κάποια από αυτά, τα αποθηκεύει σε δισδιάστατο πίνακα χαρακτήρων ως συμβολοσειρές και στη συνέχεια αντικαθιστά κάποια κομμάτια αυτών.

1. **Δομές:** Ορίστε ένα δισδιάστατο πίνακα χαρακτήρων με MAX_WORDS γραμμές και WORD_WIDTH στήλες.
2. **Επεξεργασία ορισμάτων προγράμματος και γέμισμα πίνακα:** Καλέστε μια συνάρτηση η οποία διαχειρίζεται τα ορίσματα του προγράμματος και αποθηκεύει δεδομένα στον δισδιάστατο πίνακα χαρακτήρων.

Η συνάρτηση επιστρέφει ακέραιο και παίρνει ως παραμέτρους (α) έναν ακέραιο που δηλώνει το πλήθος ορισμάτων του προγράμματος, (β) ένα πίνακα από δείκτες σε χαρακτήρα που περιέχει τα ορίσματα του προγράμματος, (γ) τον δισδιάστατο πίνακα χαρακτήρων που δημιουργήσατε νωρίτερα (δ) ένα δείκτη σε ακέραιο. Η συνάρτηση πρέπει να κάνει τα εξής:

a. Ελέγχει τα ορίσματα του προγράμματος:

- Αν ένα όρισμα είναι "-word" τότε το αμέσως επόμενο όρισμα είναι συμβολοσειρά που πρέπει να αποθηκευτεί στον δισδιάστατο πίνακα, στην επόμενη διαθέσιμη γραμμή του. Εάν αυτή δε χωράει στη γραμμή, τότε αποθηκεύεται το μέγιστο τμήμα της που χωράει. Εάν δοθούν περισσότερα από MAX_WORDS ορίσματα λέξεων, τα επιπλέον ορίσματα αγνοούνται.
- Οποιοδήποτε άλλο όρισμα θεωρείται λάθος και σε αυτή την περίπτωση η συνάρτηση επιστρέφει -1.

b. Αποθηκεύει στη διεύθυνση που δείχνει η τέταρτη παράμετρος το πλήθος συμβολοσειρών που εισήχθησαν στον δισδιάστατο πίνακα.

c. Επιστρέφει 0 εφόσον δεν παρουσιάστηκε λάθος.

3. **Διαχείριση σφάλματος:** Εάν η παραπάνω συνάρτηση επέστρεψε -1, εκτυπώστε το μήνυμα **Incorrect arguments.\n** και τερματίστε το πρόγραμμα.

4. **Αντικατάσταση κομματιών συμβολοσειρών:**

- a. Εκτυπώστε το μήνυμα **To replace:\n** και διαβάστε μία συμβολοσειρά σε πίνακα μεγέθους WORD_WIDTH.
- b. Εκτυπώστε το μήνυμα **With:\n** και διαβάστε ένα χαρακτήρα.
- c. Κατασκευάστε μια νέα συμβολοσειρά, σε πίνακα μεγέθους WORD_WIDTH, ώστε να έχει ίδιο μήκος με την παραπάνω (βήμα 4a) αλλά να αποτελείται αποκλειστικά από τον χαρακτήρα που διαβάσατε.
- d. Καλέστε μια συνάρτηση που, μέσω μίας ακόμη βοηθητικής συνάρτησης, αντικαθιστά όλες τις εμφανίσεις της

πρώτης συμβολοσειράς στον δισδιάστατο πίνακα με τα περιεχόμενα της δεύτερης:

Η συνάρτηση επιστρέφει ακέραιο και παίρνει ως παραμέτρους (α) τον δισδιάστατο πίνακα χαρακτήρων, (β) το πλήθος συμβολοσειρών που είναι αποθηκευμένες σε αυτόν (γ) τη συμβολοσειρά που διαβάστηκε στο βήμα 4a, έστω `toReplace` και (δ) τη συμβολοσειρά που κατασκευάστηκε στο βήμα 4c, έστω `replacement`. Η συνάρτηση πρέπει να κάνει τα εξής:

- a. Για κάθε μία συμβολοσειρά που βρίσκεται σε μία γραμμή του δισδιάστατου πίνακα, καλεί την παρακάτω βοηθητική συνάρτηση που αντικαθιστά όλες τις εμφανίσεις της `toReplace` μέσα στη συμβολοσειρά με την `replacement`.

- Η βοηθητική συνάρτηση παίρνει ως παραμέτρους τρεις συμβολοσειρές: την αρχική `str`, την `toReplace` και τη `replacement`. Αντικαθιστά όλες τις εμφανίσεις της `toReplace` στην `str` με τη `replacement` κι επιστρέφει το πλήθος (υπο)συμβολοσειρών που αντικαταστάθηκαν.

- b. Επιστρέφει το συνολικό πλήθος αντικαταστάσεων που έγιναν στον δισδιάστατο πίνακα.

5. Εκτύπωση αποτελεσμάτων:

- a. Εκτυπώστε το μήνυμα `Replaced: R\n` όπου R το συνολικό πλήθος αντικαταστάσεων που έγιναν.
- b. Καλέστε μία συνάρτηση που εκτυπώνει τις συμβολοσειρές που περιέχει ο δισδιάστατος πίνακας χαρακτήρων, μία ανά γραμμή.

Παραδείγματα εκτέλεσης (με κόκκινο η εντολή εκτέλεσης και η είσοδος από το πληκτρολόγιο):

```
./lab8 -word plane -word apple -word tin -word 1234
To replace:
pl
With:
*
Replaced: 2
**ane
ap**e
tin
1234
```

```
./lab8 -word aaa -word area
To replace:
a
With:
o
Replaced: 5
ooo
oreo
```

```
./lab8 -word hello -words goodbye
Incorrect arguments.
```

**Η ΠΑΡΑΚΑΤΩ ΕΠΕΚΤΑΣΗ ΤΟΥ lab8 ΕΙΝΑΙ ΠΡΟΑΙΡΕΤΙΚΗ
ΜΠΟΡΕΙΤΕ ΝΑ ΤΗ ΣΥΜΠΕΡΙΛΑΒΕΤΕ ΣΤΟ lab8.c ΠΟΥ ΘΑ ΥΠΟΒΑΛΕΤΕ ΣΤΟ AUTOLAB ΑΛΛΑ Η ΑΞΙΟΛΟΓΗΣΗ ΤΟΥ
ΕΡΓΑΣΤΗΡΙΟΥ ΘΑ ΒΑΣΙΣΤΕΙ ΣΤΗΝ ΚΥΡΙΑ ΑΣΚΗΣΗ.**

Σε αυτή την επέκταση, το πρόγραμμα σας θα διαχειρίζεται επιπλέον ορίσματα. Διατηρήστε τον κώδικα που έχετε ήδη γράψει, και κάνετε τις προσθήκες που περιγράφονται παρακάτω:

1. Προσθέστε στο lab8.h τους ορισμούς `#define MAX_NUMBERS 10` και `#define EXTENSION 1`
2. Ορίστε στη main ένα πίνακα ακεραίων μεγέθους `MAX_NUMBERS`
3. Η συνάρτηση επεξεργασίας των ορισμάτων θα πρέπει να συνεχίσει να λειτουργεί όπως έχει περιγραφεί παραπάνω για τα ορίσματα που ακολουθούν τα "-word". Επιπλέον, κάντε τις παρακάτω αλλαγές σε αυτή:
 - a. Δέχεται δύο επιπλέον παραμέτρους, τον πίνακα ακεραίων που ορίστηκε στο (1) και ένα δείκτη σε ακέραιο.
 - b. Προσθέστε την εντολή `#ifdef EXTENSION` ακριβώς πριν το σημείο που ξεκινά αυτό το βήμα και την εντολή `#endif` ακριβώς μετά το τέλος του βήματος (ρωτήστε μας στο εργαστήριο τι κάνουν αυτές οι εντολές): Αν ένα όρισμα είναι ίσο με "-number" τότε το αμέσως επόμενο όρισμα είναι ακέραιος που θα πρέπει να αποθηκευτεί στον πίνακα ακεραίων, στην επόμενη διαθέσιμη θέση του. Εάν δοθούν περισσότερα ορίσματα ακεραίων από όσο είναι το `MAX_NUMBERS`, τα επιπλέον ορίσματα ακεραίων αγνοούνται
 - c. Εφόσον δεν έχουν ανιχνευθεί λάθος ορίσματα, αποθηκεύει στη διεύθυνση που δείχνει η τελευταία παράμετρος το πλήθος ακεραίων που εισήχθησαν στον αντίστοιχο πίνακα.
4. Στη main, μετά την κλήση της συνάρτησης που εκτυπώνει τα περιεχόμενα του πίνακα συμβολοσειρών, καλέστε μια νέα συνάρτηση που παίρνει ως παραμέτρους (α) τον πίνακα ακεραίων και (β) το πλήθος ακεραίων, και αναδιατάσσει τους ακεραίους έτσι ώστε να εμφανίζονται πρώτα όσοι είναι μικρότεροι ή ίσοι του πρώτου και ακολούθως όσοι είναι μεγαλύτεροι αυτού. Η αναδιάταξη πρέπει να γίνει με τον τρόπο που περιγράφεται παρακάτω.
5. Στη main, μετά την κλήση της συνάρτησης αναδιάταξης, γράψτε την εντολή `#ifdef EXTENSION` και στη συνέχεια καλέστε μια νέα συνάρτηση που παίρνει ως παραμέτρους (α) τον πίνακα ακεραίων και (β) το πλήθος ακεραίων και τους εκτυπώνει σε μία σειρά με ένα χαρακτήρα ' ' (space) πριν από κάθε ακέραιο. Στο τέλος, εκτυπώνει χαρακτήρα αλλαγής γραμμής. Μετά την κλήση της συνάρτησης γράψτε την εντολή `#endif`

Αναδιάταξη αριθμών

Η αναδιάταξη των αριθμών πρέπει να γίνει με μία διάτρεξη του πίνακα αριθμών ως εξής: Θα διατρέξετε τον πίνακα από αριστερά προς τα δεξιά και από δεξιά προς αριστερά αναζητώντας στοιχεία που είναι "εκτός σειράς" (δηλαδή κάποιο μικρό στοιχείο να είναι πιο δεξιά ενώ κάποιο μεγάλο να είναι πιο αριστερά). Κάθε φορά που βρίσκετε δύο τέτοια στοιχεία, τα αντιμεταθέτετε.

Χρησιμοποιήστε δύο ακέραιες μεταβλητές, έστω left και right για να κρατάτε τη θέση που βρίσκεστε στον πίνακα. Η θέση left ξεκινάει από την αρχή του πίνακα και κινείται δεξιά προς το τέλος του ενώ η θέση right ξεκινάει από το τέλος του πίνακα και κινείται αριστερά προς την αρχή του. Το left πρέπει να βρίσκεται πάντα αριστερά του right.

Ξεκινώντας από το τέλος του πίνακα και κινούμενοι προς την αρχή, συγκρίνετε ένα-ένα τα στοιχεία που συναντάτε με το πρώτο. Όσο βρίσκετε μεγαλύτερα στοιχεία, δε χρειάζεται να κάνετε κάτι. Όταν βρείτε κάποιο μικρότερο ή ίσο, σταματάτε προσωρινά στη θέση που το βρήκατε (η θέση αυτή είναι πια η right).

Ξεκινώντας από την αρχή του πίνακα και κινούμενοι προς το τέλος, συγκρίνετε ένα-ένα τα στοιχεία που συναντάτε με το πρώτο. Όσο βρίσκετε μικρότερα ή ίσα στοιχεία, δε χρειάζεται να κάνετε κάτι. Όταν βρείτε κάποιο μεγαλύτερο, σταματάτε προσωρινά στη θέση που το βρήκατε (η θέση αυτή είναι πια η left).

Σε αυτό το σημείο, εφόσον το left εξακολουθεί να είναι "αριστερά" (μικρότερο) του right, αντιμεταθέτετε τα στοιχεία στις θέσεις left και right, και επαναλαμβάνετε την παραπάνω διαδικασία. Διαφορετικά τερματίζετε.

Παράδειγμα: Ας υποθέσουμε ότι τα περιεχόμενα του πίνακα είναι 10 9 20 16 2 1 15 3.

1η επανάληψη:

Το left προχωράει δεξιά μέχρι να βρει κάτι μεγαλύτερο του 10, άρα σταματάει όταν γίνει 2 (θέση του 20).

Το right προχωράει αριστερά μέχρι να βρει κάτι μικρότερο ή ίσο του 10, άρα σταματάει άμεσα στη θέση 7 (θέση του 3).

Αντιμετατίθενται τα 20 και 3 και ο πίνακας γίνεται 10 9 3 16 2 1 15 20.

2η επανάληψη:

Το left προχωράει δεξιά μέχρι να βρει κάτι μεγαλύτερο του 10, άρα σταματάει όταν γίνει 3 (θέση του 16).

Το right προχωράει αριστερά μέχρι να βρει κάτι μικρότερο ή ίσο του 10, άρα σταματάει όταν γίνει 5 (θέση του 1).

Αντιμετατίθενται τα 16 και 1 και ο πίνακας γίνεται 10 9 3 1 2 16 15 20.

3η επανάληψη:

Το left προχωράει δεξιά μέχρι να βρει κάτι μεγαλύτερο του 10, άρα σταματάει όταν γίνει 5 (θέση του 16).

Το right προχωράει αριστερά μέχρι να βρει κάτι μικρότερο ή ίσο του 10, άρα σταματάει όταν γίνει 4 (θέση του 2).

Σε αυτό το σημείο, το right έχει μετακινηθεί πιο αριστερά από το left επομένως η επανάληψη σταματά.

Παραδείγματα εκτέλεσης

```
./lab8 -word apple -number 5 -word tin -number 21 -number 8 -number 3
```

```
To replace:
```

```
pl
```

```
With:
```

```
*
```

```
Replaced: 1
```

```
ap**e
```

```
tin
```

```
##
```

```
5 3 8 21
```

```
./lab8 -number 10 -number 9 -number 20 -number 3 -number 1 -number 15 -number 4
```

```
To replace:
```

```
number
```

```
With:
```

```
x
```

```
Replaced: 0
```

```
##
```

```
10 9 4 3 1 15 20
```

```
> ./lab8 -number 10 -word mississippi -number 8 -number 17 -word spin
```

```
To replace:
```

```
ss
```

```
With:
```

```
$
```

```
Replaced: 2
```

```
mi$$i$$$ppi
```

```
spin
```

```
##
```

```
10 8
```

[ΓΙΑ MAX_NUMBERS ίσο με 2]