

## Παρατηρήσεις για το lab7

**Μεταγλώττιση:** Ο κώδικας θα πρέπει να μεταγλωττίζεται χωρίς **warnings**. Χαρακτηρισμός “μη ικανοποιητικό” σημαίνει ότι είχατε τουλάχιστον ένα warning. Κώδικας που παράγει λάθη (errors) κατά τη μεταγλώττιση βαθμολογείται με FAIL.

**Στοιχίση/Κενά:** Ο κώδικας πρέπει να είναι στοιχισμένος σωστά, με ιδιαίτερη προσοχή στη στοιχίση εμφωλευμένων for/while/if. Δεν πρέπει να έχει περισσότερες από μία διαδοχικές κενές γραμμές, αλλά ούτε και να είναι όλα “κολλημένα”.

Χαρακτηρισμός “μέτρια” γενικά σημαίνει ότι έχετε κατά το πλείστον καλή στοιχίση και χρήση κενών γραμμών. Χαρακτηρισμός “μη ικανοποιητική” σημαίνει ότι δεν έχετε στοιχίσει τον κώδικά σας ή η στοιχίση είναι πολύ ασυνεπής ειδικά όσον αφορά τις δομές ελέγχου/επανάληψης.

**Σταθερές:** Τα μεγέθη πινάκων ορίζονται με **#define**, και πρέπει να χρησιμοποιείται το όνομα της σταθεράς σε όλο το πρόγραμμα (στη δεύτερη άσκηση, θα πρέπει να χρησιμοποιηθεί και στην αρχική printf).

**Είσοδος:** Και στις δύο ασκήσεις πρέπει με χρήση της **sprintf** να δημιουργήσετε ένα format string με βάση το SIZE. Το format string θα χρησιμοποιηθεί στη συνέχεια στις κλήσεις της **scanf** για να δηλώσει το μέγιστο μέγεθος συμβολοσειράς που το πρόγραμμα σας επιτρέπεται να διαβάσει. Προσοχή στην τιμή που πάει στο %d: πρέπει να είναι ένα λιγότερο από το μέγεθος του string, δηλαδή `sprintf(format, "%%ds", SIZE-1)`;

Χαρακτηρισμός “μέτριο” σημαίνει ότι το format string δεν είναι μεταβλητό με βάση το SIZE, αλλά τουλάχιστον έχει προσδιοριστεί κατάλληλο πλάτος (για παράδειγμα `scanf("%9s", str)`;

Χαρακτηρισμός “μη ικανοποιητικό” σημαίνει ότι δεν υπάρχει πλάτος στη scanf (για παράδειγμα `scanf("%s", str)`; ή έχει λάθος πλάτος (για παράδειγμα `scanf("%10s", str)`; για str με μέγεθος 10)

**Έλεγχοι:** Όπως και στα προηγούμενα δύο εργαστήρια, πρέπει να γίνει έλεγχος εγκυρότητας της ακέραιας ποσότητας που διαβάστηκε και επανάληψη εφόσον αυτή δεν είναι σωστή.

**ΠΡΟΣΟΧΗ:** Η συνθήκη `(number <= 0 && number >= 10)` δεν είναι ΠΟΤΕ αληθής γιατί δεν είναι δυνατό ένας ακέραιος να είναι ταυτόχρονα αρνητικός και μεγαλύτερος του 10. Να σκέφτεστε πάντα προσεκτικά ποιο είναι το εύρος τιμών που κάνει αληθή ή ψευδή μια συνθήκη που γράψατε και όταν σας γίνεται μια παρατήρηση σε κάποιο lab για την μη-ορθότητα των ελέγχων σας, να τους επανεξετάζετε ή να ζητάτε διευκρινίσεις αν δεν καταλαβαίνετε το λάθος! Στην προκειμένη περίπτωση, πρέπει να βάλουμε `||` ώστε η συνθήκη να είναι αληθής αν το number είναι αρνητικό ή μεγαλύτερο του 10 και ψευδής αν το number είναι ανάμεσα στο 0 και 10 (μη-συμπεριλαμβανομένων).

**Πίνακες:** Οι πίνακες χαρακτήρων, ειδικά αυτοί που αποθηκεύουν συμβολοσειρές μέγιστου μήκους, θα πρέπει να είναι αρχικοποιημένοι έχοντας τον χαρακτήρα '\0' τουλάχιστον στην μηδενική τους θέση. Η αρχικοποίηση αυτή ισοδυναμεί με κενή συμβολοσειρά, κι ο πιο εύκολος τρόπος να γίνει είναι με τη σύνταξη `char str[SIZE] = {'\0'};`

Επίσης, δώστε προσοχή στα μεγέθη. Στην άσκηση 1 της Τετάρτης, οι πίνακες των δύο πρώτων βημάτων πρέπει να έχουν μέγεθος SIZE ενώ αυτός του τρίτου βήματος `2*SIZE-1` (δεκτό και `2*SIZE`)

### Αλγόριθμος :

Πρώτα μια γενική παρατήρηση: Αρκετοί αποθηκεύσατε το αποτέλεσμα της `strcmp` σε μια μεταβλητή και χρησιμοποιήσατε τη μεταβλητή στη συνθήκη της `if` που ακολουθούσε. Αυτό δεν είναι λάθος, αλλά έχει το μειονέκτημα ότι καταλήγει σε λιγότερο κατανοητό κώδικα, ειδικά δεδομένου ότι οι μεταβλητές που επιλέγατε είχαν ονόματα όπως `result1`, `result2`, `result3`.

Τι είναι πιο κατανοητό? Το `if (result1 > 0)` ή το `if (strcmp(max, word) > 0)` ?

Επειδή κάποιοι ίσως νομίζετε ότι το δεύτερο δεν είναι έγκυρο, σας διαβεβαιώνουμε ότι είναι. Το `strcmp(max, word)` είναι κλήση σε συνάρτηση η οποία επιστρέφει τιμή, επομένως το `strcmp(max, word)` είναι μια έκφραση και μπορεί να χρησιμοποιηθεί οπουδήποτε είναι έγκυρο να χρησιμοποιηθεί έκφραση.

**[Άσκηση 1 Τετάρτη]** Αρχικοποιούμε τους πίνακες για τις μέγιστες συμβολοσειρές με '\0' (αρκεί στην πρώτη θέση) ώστε να γίνει σωστά η σύγκριση στο πρώτο βήμα. Σε κάθε βήμα, πρώτα ελέγχουμε αν η συμβολοσειρά που διαβάσαμε είναι μεγαλύτερη από οποιαδήποτε από τις δύο, κι αν ναι, αντικαθιστούμε τη μικρότερη από αυτές με τη νέα. Με άλλα λόγια:

```
if (strcmp(word, max1) > 0 || strcmp(word, max2) > 0) {
    if (strcmp(max1, max2) > 0) {
        strcpy(max2, word);
    }
    else {
        strcpy(max1, word);
    }
}
```

Δεν είναι σωστό να κάνουμε "τυχαία" την αντικατάσταση. Για παράδειγμα αν max1="alpha", max2="gamma" και διαβάσουμε "delta", τότε το σωστό είναι να αντικαταστήσουμε το "alpha" με το "delta" κι όχι το "gamma" με το "delta".

Στο τέλος, επικολλούνται τα δύο strings στο τελικό με σειρά ανάλογα με το μήκος τους. Αν το τελικό string έχει αρχικοποιηθεί σε "" τότε μπορείτε να χρησιμοποιήσετε δύο strcat. Διαφορετικά, θα πρέπει να χρησιμοποιήσετε πρώτα strcpy για να αντιγράψετε το ένα μέγιστο string στο τελικό και μετά strcat για να κολλήσετε το δεύτερο στο τέλος. Μπορείτε να σκεφτείτε γιατί η αρχικοποίηση κάνει διαφορά?

**[Άσκηση 2]** Η συνάρτηση για τον έλεγχο αν ένας χαρακτήρας είναι σημείο στίξης είναι η **ispunct**. Προσοχή στην κατασκευή του τελικού πίνακα. Θα πρέπει να φροντίσετε να τερματίζει σωστά με '\0'. Επίσης, επειδή θέλουμε να αλλαχθεί ο αρχικός πίνακας, αν χρησιμοποιήσατε βοηθητικό τότε στο τέλος πρέπει να αντιγράψετε τον βοηθητικό στον αρχικό.

**[Άσκηση 1 Πέμπτης]** Χρειαζόμαστε τρεις βασικούς πίνακες: Έναν που θα περιέχει το τελικό αποτέλεσμα (final), έναν για να αποθηκεύουμε τη λέξη που διαβάσαμε κάθε φορά (current) κι έναν που θα περιέχει την προηγούμενη λέξη που διαβάστηκε (previous). Αρχικά, η προηγούμενη λέξη είναι "", δηλαδή αρχικοποιούμε τον πίνακα ώστε να περιέχει το '\0' τουλάχιστον στην πρώτη θέση του πίνακα. Αυτό εγγυάται πως οτιδήποτε διαβάσουμε στο πρώτο βήμα θα είναι λεξικογραφικά μεγαλύτερο. Επίσης, είναι σημαντικό να αρχικοποιήσουμε τον πίνακα final σε "" γιατί η strcat λειτουργεί ψάχνοντας να βρει το τέλος της αριστερής παραμέτρου για να κολλήσει εκεί τη δεξιά παράμετρο. Το τέλος μιας συμβολοσειράς είναι το πρώτο '\0' που εμφανίζεται στον πίνακα (άρα αρχικά, το πρώτο '\0' πρέπει να είναι στην πρώτη θέση).

Η λογική είναι:

Αρχικοποίησε την previous και τη final σε ""

Σε επανάληψη:

Διάβασε την current

Αν η current είναι μεγαλύτερη λεξικογραφικά της previous και το μήκος της μεταξύ 2 και 9,

Επικόλησε με strcat την current στη final

**Όρια πινάκων:** Η πρώτη θέση ενός πίνακα μεγέθους SIZE είναι 0 και η τελευταία SIZE-1. Προσπέλαση του table[i] με i>=SIZE είναι εκτός των ορίων του πίνακα. Λάθος μέγεθος στην κατασκευή του format string μπορεί να επιτρέψει προσπέλαση εκτός ορίων.

**Έξοδος:** Η έξοδος του προγράμματος πρέπει να είναι ΑΚΡΙΒΩΣ ίδια με την εκφώνηση.