

## 2ο Εργαστήριο Αρχιτεκτονικής Η/Υ: MIPS assembly: Μέτρηση συχνότητας εμφάνισης χαρακτήρων σε string

A. Ευθυμίου

**Παραδοτέο: Τετάρτη 23 Οκτώβρη, 23:59**

Το αντικείμενο αυτής της άσκησης είναι ένα πρόγραμμα που διατρέχει τα στοιχεία ενός αλφαριθμητικού (string) και καταγράφει πόσες φορές εμφανίζεται κάθε γράμμα (και λοιποί χαρακτήρες) μέσα σε αυτό.

Θα πρέπει να έχετε μελετήσει τα πρώτα 2 μαθήματα για τη γλώσσα assembly του MIPS που αντιστοιχούν μέχρι την ενότητα 2.7 και, επίσης, την ενότητα 2.9 του βιβλίου.

Για να ξεκινήσετε, ακολουθήστε τον σύνδεσμο <https://classroom.github.com/a/nKVIRC0Q>. Κάνοντας κλικ στον σύνδεσμο, δημιουργείται έναν νέο αποθετήριο στον οργανισμό του μαθήματος. Μπορείτε να δείτε το νέο αποθετήριο είτε αμέσως μετά το κλικ στον σύνδεσμο. Το URL του αποθετηρίου θα έχει τη μορφή <https://github.com/UoI-CSE-MYY505/lab02-ghUsername>, όπου ghUsername το όνομα χρήστη που έχετε στο GitHub.

Κλωνοποιήστε το με την εντολή:

```
git clone https://github.com/UoI-CSE-MYY505/lab02-ghUsername.git
```

Για να πάρετε τα αρχεία της εργαστηριακής άσκησης, μεταβείτε στον κατάλογο που θα δημιουργηθεί από το παραπάνω βήμα και θα έχει το ίδιο όνομα με το αποθετήριο (αλλάζετε το ghUsername με το όνομα χρήστη).

### 1 Η άσκηση

Ένα αλφαριθμητικό (string) είναι ένας πίνακας χαρακτήρων αποθηκευμένος στη μνήμη σε διαδοχικές θέσεις. Οι χαρακτήρες είναι bytes και χρησιμοποιούν την κωδικοποίηση ASCII που αντιστοιχεί αριθμούς σε χαρακτήρες. Το τέλος του αλφαριθμητικού παριστάνεται με ένα byte που έχει την τιμή 0 και, για παράδειγμα, το μικρό a είναι το 97 ενώ το κεφαλαίο A είναι το 65. Η διεύθυνση του string είναι η διεύθυνση του πρώτου στοιχείου του, δηλαδή αυτού που βρίσκεται στην μικρότερη διεύθυνση.

Για να μετρηθεί πόσες φορές εμφανίζεται κάθε χαρακτήρας στο string, θα χρειαστεί μία μεταβλητή-μετρητής για κάθε είδος χαρακτήρα, με κεφαλαία - μικρά γράμματα να μετριοούνται ξεχωριστά. Επειδή στην κωδικοποίηση ASCII υπάρχουν ως 256 διαφορετικοί χαρακτήρες, θα χρησιμοποιηθεί ένας πίνακας ακεραίων 256 θέσεων για τους μετρητές. (Αν και οι πρώτοι 31 είναι ειδικοί «αόρατοι» χαρακτήρες και οι τελευταίοι 127 εξαρτώνται από την επιλεγμένη γλώσσα, οπότε δεν θα εμφανίζονται στην άσκηση, θα χρησιμοποιήσουμε και τις 256 πιθανές θέσεις για απλούστευση). Παρόμοια με το string, ένας πίνακας ακεραίων, 32-bit λέξεων στην περιπτωσή μας, αποθηκεύεται ως διαδοχικές λέξεις με αύξουσα διεύθυνση στη μνήμη και η διεύθυνση του πίνακα ορίζεται ως η διεύθυνση του πρώτου στοιχείου.

Στο αρχείο lab02.asm θα βρείτε έναν μικρό σκελετό του κώδικα όπου στον καταχωρητή a0 ανατίθεται η διεύθυνση του string (string) και στον a1, η διεύθυνση του πίνακα των μετρητών εμφάνισης χαρακτήρων (counters). Θα δείτε παρακάτω, με σχόλια, σε ποιο σημείο θα πρέπει να συμπληρώσετε κώδικα, αμέσως μετά το label countChar. Ο κώδικάς σας θα πρέπει να διατρέχει το string, να βρίσκει τον μετρητή που αντιστοιχεί στον εκάστοτε χαρακτήρα και να τον αυξάνει.

Το label countChar είναι απαραίτητο για τον αυτόματο έλεγχο της άσκησης (με το munit.jar), όπως επίσης και τα label string και counters που αντιστοιχούν στο string και στον πίνακα των μετρητών αντίστοιχα. Μην αλλάξετε τον κώδικα πριν από το countChar. Επειδή όταν υπάρχει το label main, το munit ίσως να μην λειτουργεί σωστά, μην το προσθέσετε στον κώδικα.

Το αρχείο Lab02TestCountChar.java περιέχει ένα τεστ με μερικές τιμές εισόδου που ελέγχει αν το lab02.asm λειτουργεί σωστά. Για να το τρέξετε, δώστε τις εντολές:

```
javac -cp munit.jar Lab02TestCountChar.java  
java -jar munit.jar lab02.asm Lab02TestCountChar.class
```

## 2 Παραδοτέο και κριτήρια αξιολόγησης

Το παραδοτέο της άσκησης είναι το αρχείο lab02.asm που περιέχει το πρόγραμμά σας. Προαιρετικά αλλάξτε και το README.md για να βλέπετε το badge του Travis CI.

Πρέπει να κάνετε commit τις αλλαγές σας και να τις στείλετε (push) στο αποθετήριό σας στο GitHub για να βαθμολογηθούν πριν από την καταληκτική ημερομηνία!

Τα προγράμματά σας θα βαθμολογηθούν για την ορθότητά τους, την ποιότητα σχολίων και τη ταχύτητα εκτέλεσής τους. Το τελευταίο σημαίνει ότι πρέπει να είναι σύντομα και ο αριθμός εντολών, ειδικά μέσα σε βρόγχο, να είναι όσο γίνεται μικρότερος.