

4ο Εργαστήριο Αρχιτεκτονικής Η/Υ: MIPS assembly: Ευκλείδειος αλγόριθμος υπολογισμού μέγιστου κοινού διαιρέτη

A. Ευθυμίου

Παραδοτέο: Παρασκευή 27 Μάρτη, 23:00

Το αντικείμενο αυτής της άσκησης είναι ένα πρόγραμμα που υπολογίζει τον μέγιστο κοινό διαιρέτη δύο αριθμών. Σε αυτή την εργαστηριακή άσκηση θα γράψετε ένα προγράμμα assembly που χρησιμοποιεί υπορουτίνες και αναδρομή. Θα πρέπει να έχετε μελετήσει τα μαθήματα για τη γλώσσα assembly του MIPS που αντιστοιχούν μέχρι και την ενότητα 2.8 του βιβλίου.

Έχετε 2 εβδομάδες διαθέσιμες γι'αυτή την άσκηση. Μη την αφήσετε όμως για την τελευταία στιγμή! Η χρήση της στοίβας για υπορουτίνες είναι αρκετά περίπλοκη και θέλει σκέψη και πιθανόν πολλές δοκιμές για να υλοποιήσετε σωστά αναδρομικές συναρτήσεις.

Για να παρατηρήσετε τί συμβαίνει στην περιοχή της μνήμης που αντιστοιχεί στη στοίβα, αλλάξτε την επιλογή στο κάτω μέρος του παραθύρου Data Segment σε current \$sp.

1 Η άσκηση

Ένας από τους πιο γνωστούς αλγόριθμους για τον υπολογισμό του μέγιστου κοινού διαιρέτη δύο φυσικών αριθμών είναι αυτός του Ευκλείδη. Αν δεν τον θυμάστε, υπάρχει πολύ αναλυτική περιγραφή του στην Wikipedia στο σύνδεσμο: http://en.wikipedia.org/wiki/Euclidean_algorithm. Δείτε και τον ψευτοκώδικα που ακολουθεί.

Σε αυτή την άσκηση θα υλοποιήσετε τον αλγόριθμο σε μία υπορουτίνα assembly. **Η ρουτίνα θα πρέπει να είναι αναδρομική (recursive) και να ακολουθεί τις συμβάσεις των υπορουτινών MIPS που αναφέρθηκαν στον μάθημα.** Επιπλέον, επειδή δεν χρησιμοποιούμε τις εντολές πολλαπλασιασμού και διαίρεσης του MIPS, θα χρειαστεί να υλοποιήσετε την συνάρτηση υπολογισμού του υπολοίπου μιας διαίρεσης ως ξεχωριστή υπορουτίνα (με όνομα mod).

Απαγορεύεται να χρησιμοποιήσετε οποιαδήποτε εντολή του MIPS που είναι σχετική με πολλαπλασιασμό και διαίρεση, συμπεριλαμβανομένης της rem. Όπως πάντα οι μόνες εντολές διακλάδοσης που επιτρέπονται είναι οι beq, bne.

Συνιστάται να γράψετε και να ελέγξετε πρώτα την υπορουτίνα που υπολογίζει το υπόλοιπο και μετά αυτή του μέγιστου κοινού διαιρέτη.

Για να πάρετε τα αρχεία της εργαστηριακής άσκησης, μεταβείτε στον κατάλογο εργασίας που είχατε κλωνοποιήσει από το αποθετήριό σας στο GitHub. (cd <όνομα χρήστη GitHub>-labs). Μετά, κάνετε τα παρακάτω βήματα:

```
git remote add lab04_starter https://github.com/UoI-CSE-MYY402/lab04_starter.git
git fetch lab04_starter
git merge lab04_starter/master -m "Fetched lab04 starter files"
```

Στον κατάλογο lab04 σας δίνεται το αρχείο gcd.asm που καλεί την υπορουτίνα gcd με αριθμούς που διαβάξει από τη μνήμη. Μην αλλάξετε τον κώδικα του κύριου προγράμματος που καλεί την gcd ούτε τις θέσεις ή την σειρά των ετικετών δεδομένων, γιατί χρησιμοποιούνται από τον αυτόματο έλεγχο. Αν χρειαστεί να προσθέσετε άλλες μεταβλητές, κάντε το μετά από τις υπάρχουσες. Δοκιμάστε το πρόγραμμα με άλλους αριθμούς αν θέλετε, αλλά στο τέλος παραδώστε το με τους αρχικούς αριθμούς (και με την ίδια σειρά).

2 Ψευτοκώδικας

Ο ψευτοκώδικας της υπορουτίνας υπολογισμού του μέγιστου κοινού διαιρέτη των a και b δίνεται παρακάτω. Παρατηρήστε ότι στην αρχή ελέγχεται αν ο $b > a$ και οι τιμές τους ανταλλάσσονται ώστε στη συνέχεια να είναι ο $a \geq b$. Η ανταλλαγή θα πρέπει να γίνει επιτόπου, δεν χρειάζεται να καλέσετε κάποια υπορουτίνα.

```
gcd (a,b)
    if b > a
        swap(a,b)
    if b == 0
        return a
    else
        return gcd(b, mod(a,b))
```

Ο ψευτοκώδικας της υπορουτίνας υπολογισμού του υπόλοιπου της διαίρεσης του a με τον b δίνεται παρακάτω. Υποθέτουμε ότι ο $a \geq b > 0$ και δεν χρειάζεται να το ελέγχετε στον κώδικα που θα γράψετε. Η υπορουτίνα gcd εξασφαλίζει ότι αυτοί οι περιορισμοί ισχύουν πάντα.

```
mod (a,b)
    while a >= b
        a = a - b
    return a
```

3 Παραδοτέο

Το παραδοτέο της άσκησης είναι το αρχείο lab04.asm που περιέχει το πρόγραμμά σας και η αναμενόμενη τιμή για τον μέγιστο κοινό διαιρέτη στο αρχείο test/lab04_tester.

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

Το πρόγραμμά σας θα βαθμολογηθεί για την ορθότητά του, την ποιότητα σχολίων και την ταχύτητα εκτέλεσής του.