

6ο Εργαστήριο Αρχιτεκτονικής Η/Υ:

Διοχέτευση Risc-V,

A. Ευθυμίου

Παραδοτέο: Βλ. ecourse, 23:59

Με αυτή την εργαστηριακή άσκηση θα χρησιμοποιήσετε τον Ripes για να τρέξετε ένα πρόγραμμα και να δείτε πως επηρεάζεται η ταχύτητα εκτέλεσης του από τους κινδύνους διοχέτευσης.

1 Παραλαβή των αρχείων της άσκησης

Ακολουθήστε τον σύνδεσμο που δίνεται στο ecourse για να δημιουργηθεί το δικό σας αποθετήριο της άσκησης και κλωνοποιήστε το.

Στον κατάλογο που θα δημιουργηθεί, lab06-ghUsername, θα βρείτε το αρχείο lab06.s και ένα αρχείο απλού κειμένου quizLab06 με ερωτήσεις που θα πρέπει να απαντήσετε. Δεν υπάρχει python αρχείο για έλεγχο σε αυτή την άσκηση.

2 Περιγραφή

Φορτώστε το πρόγραμμα του lab06.s στον Ripes με την επιλογή 5-stage processor(RV32I, M). Προσοχή πρέπει να είναι ενεργοποιημένες οι προωθήσεις και αναβολές, οπότε **μην επιλέξετε** τον επεξεργαστή με 5 στάδια και την κατάληξη “w/o ...”.

Για την άσκηση αυτή θα τρέξετε το πρόγραμμα, θα δείτε πως αντιδρά ο διεχτετευμένος επεξεργαστής και θα απαντήσετε στις ερωτήσεις που βρίσκονται στο quizLab06.

Στο lab06.s οι εντολές μέσα στο βρόχο έχουν ένα label της μορφής l και αριθμός. Αυτό τον αριθμό θα δίνετε σε απάντηση όπου ζητείται να υποδείξετε κάποια εντολή (π.χ. 3 αν η εντολή έχει label l3). Σε όσες ερωτήσεις ζητούν αριθμό καταχωρητή γράφετε τον αριθμό καταχωρητή χωρίς το x μπροστά. Π.χ. για x5, γράφετε 5.

Χρησιμοποιήστε τη λειτουργία Show pipeline diagram του Ripes που εξηγήθηκε στο προηγούμενο εργαστήριο. Στο διάγραμμα αυτό, με παύλα συμβολίζεται η αναβολή (stall), ενώ εντολές που σταματάνε απότομα σε κάποιο ενδιάμεσο στάδιο, ακυρώνονται (εκκένωση - flush μετά από διακλάδωση). Στην επάνω σειρά αυτού του πίνακα υπάρχει ο αριθμός κύκλου από την αρχή της εκτέλεσης.

3 Παραδοτέο

Παραδώστε το αλλαγμένο αρχείο απαντήσεων στο αποθετήριο σας στο github.