Processor Assignment

As part of this assignment, you'll be writing codes in assembly language, using the op-codes of the simple processor discussed in class.

Instructions

- You've been provided with a Simulator directory and an Instruction-Set, that summarises all the op-codes used in the design of our simple processor.
- The simulator directory contains the executable sim that simulates the working of the processor. Open up this directory and run the executable.

```
cd Simulator
./sim
```

And voila! The processor is now ready to execute code!

- First, it will ask you to load the program file. Enter the name of your code file, for instance,add.txt . Ensure all your code files are present in the Simulator directory and are in (.txt) format. The code must contain only the opcodes and no comments.
- Next, if you need to pre-load the memory, enter the name of your file, like m1.txt . We have provided you with the memory files for the assignment.
- The format of the memory file is i j: a1...aj where i refers to the address of the first element (a1). j is the number of elements and a1..aj are the numbers that we load into the memory.
- You can print the initial content of the registers and memory, and after the program finishes execution, you can print the contents again, to comprehend the changes.

MacOS users

The executable doesn't seem to run on MacOS. You would have to use a virtual environment like Vagrant. Vagrant basically allows you to use a guest operating system like Ubuntu on your MacOS, but you can only use it through command line. It doesn't generate the GUI itself. (which is not needed anyways).

Follow these steps to install vagrant on MacOS: You need to first install Vagrant (https://www.vagrantup.com/downloads.html) and VirtualBox(https://www.virtualbox.org/). Once that's complete, initialise vagrant and run the environment.

```
vagrant init hashicorp/precise64
vagrant up
```

Next, ssh into the server to start using the guest OS.

```
vagrant ssh
```

You'll see your terminal window read `Ubuntu 12.XX` , meaning you're now using a Linux environment! Make sure you install and update vim in your vagrant environment.

```
sudo apt-get update
sudo apt-get install vim
```

Now COME BACK to your MacOS by using the `exit` command:

```
exit
```

Once you're back to MacOS, then you can use vagrant scp to move the simulator folder from your local machine into vagrant. Install vagrant scp, and then move the folder to your vagrant virtual machine.

```
 vagrant plugin install vagrant-scp
vagrant scp /Users/abhinav/assignment4 :~
```

The second command moves the assignment4 folder into vagrant. `:~` refers to home directory in vagrant. Now you can ssh into vagrant and you'll see the assignment waiting for you there!