Operating Systems
CMPT 424

─Lab 4 ─────────────────────────────────

Goals

Executing one program

This approximately one-hour active learning exercise will you help you make progress on the practical aspects of developing your operating system.

Instructions

1. Now that you have a program loaded in memory and a CPU on which to execute the op codes, write the routines necessary for your OS to execute the program by implementing the classic fetch - decode - execute cycle.
   a. Fetch the next instruction from memory.
   b. Decode it. This is to say, determine what CPU routine to call based on the op code.
   c. Execute it. Meaning, call that routine.
2. Be sure to synchronize your CPU "tick" with the system clock. In other words, you can only execute one instruction per clock tick. This usually trips up some students, so think carefully about it.
3. Add the new features as specified in your Issues and *i*Project 2. I'm not even mentioning best practices any more. If that was an issue, you'd be gone by now.
4. Test everything.
5. Test again.
6. Keep testing. Seriously. Did you think this would change?
7. Read chapters 2.3 and 2.4 in the 8th edition of our text.

Questions

1. What is the relationship between a guest operating system and a host operating system in a system like VMware? What factors need to be considered in choosing the host operating system?

Resources

- http://www.labouseur.com/commondocs/6502alan-instruction-set.pdf
- https://skilldrick.github.io/easy6502/

- Chapter 6 in http://pages.cs.wisc.edu/%7Eremzi/OSTEP/

Grading

Your work on this lab will contribute to your grade for *i*Project 2.

Submitting

Commit your work to your **private** GitHub account in an appropriately-named folder. Make sure to tag your commit messages with the Issue number they address.