Operating Systems
CMPT 424

Goals

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

Instructions

1. Add the iProject 2 functional requirements as Issues (optionally labeled as
   enhancements) in GitHub as element of an "iProject 2" milestone.
2. Have a look at cpu.ts in the host directory. This is where you will implement the
   6502a operation codes (see resources, below) so you can execute a single user
   program. Start on this with the LDA and STA op codes.
3. Testing LDA and STA operations is a nice way to test your memory implementation
too.
4. Add the new features as specified in your Issues and iProject 2. Continue
demonstrating programming best practices. (If you have not been demonstrating
programming best practices up to this point then drop this class before it’s too late
and Alan will see you again next year. Go read Code Complete 2nd edition in the mean
time.)
5. Test everything.
6. Test again.
7. Keep testing. Did you think this would change?
8. Read chapter 8.3 in the 8th edition of our text.

Questions

1. Explain the difference between internal and external fragmentation.
2. Given five (5) memory partitions of 100KB, 500KB, 200KB, 300KB, and 600KB (in that
   order), how would optimal, first-fit, best-fit, and worst-fit algorithms place processes
   of 212KB, 417KB, 112KB, and 426KB (in that order)?

Resources

• https://skilldrick.github.io/easy6502/
• http://www.visual6502.org
• http://www.atariarchives.org/mlb/

Grading

Your work on this lab will contribute to your grade for iProject 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.