Daniel Bielejeski and Ricky Rodriguez

CISC 340

Project 1

Simulator Overview

Simulator takes as input a file containing machine code. We assumed the machine code is valid and properly assembled. We read through the file and then for each line we get the op code, using bit shifting. Follow that, we get the regA, regB, and the destination register also using bit shifting. Based on the opcode, we get the appropriate parameters e.g. add needs RegA, RegB and Destination, whereas halt does not need any other information.

Tracks the memory addresses for each line of code. Prints out memory addresses and contents of each register. Prints out all the changes made in the simulated code.

One difficulty encountered included any command that would modify the program counter. In the current simulator, the program counter is incremented after a command is executed. This is contrary to how the computer increments the counter. But with the same result. This was done for greater readability of code. The difficulty arose when dealing with shifting the program counter since we know after the shift it would be incremented again. To solve this we subtracted one when necessary. We had to subtract for jump-and-link register and branch on equals.

Another major difficulty was using the struct concept in C to track program state. Through much hard work and study we figured it out.