

CS 1428 Honors

Lab 7

Jared Wallace

Overview

Today we will be extending our assembler even further. We are going to modify it so that the program can read in the instructions from a file, rather than forcing the user to input the instructions line by line.

Questions

1. (15 pts) Since we will be reading in several lines of instructions from a file, we need somewhere to store these instructions. Assuming we set a hard limit of 512 different instructions, each of which consists of four integers, how can we store these instructions? Write your answer in valid C++ code.
2. (15 pts) How would you then write the loop to read the instructions and store them into the place you created in question one? Again, write your answer as valid C++ code.
3. (20 pts) Modify the switch statement below to properly handle the instructions. In other words, modify the statement to reflect the fact that we now get our instructions from "memory", and not the console.

```

switch(instruction)
{
    case 0:
        memory[data0] = memory[data1] + memory[data2];
}

```

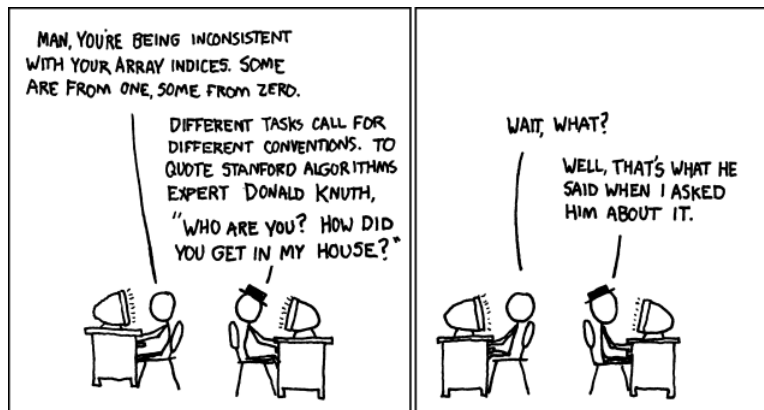
(You only need to change the case listed, not all possible cases)

4. (50 pts) Using the answers from questions one through three, modify your assembler to read it's instructions from a file (prompt the user for the name). Your program should comply with the following requirements:

- Read all instructions into memory at one time.
- Process the instructions in order, one line at a time.
- Properly handle the same sample programs from last time (supplied)

Deliverables

Hard copy of the source code you wrote (assembler.cpp). Soft copy (upload to homework upload) of your source code. As always, make sure you make a commit and push your work.



His books were kinda intimidating; rappelling down through his sky-light seemed like the best option.