Project Details and Dates

- Due on April 4th by 2pm on USB (for copying).
- Work either as a solo or as a team of 2
- Design and code a C program (10%) which will convert a perfectly programmed assembly program (read in as text) into an binary program for the LC3 machine.
- Your task is to write a well documented and formatted C program which reads in a properly (and perfectly) formatted .asm file (as detailed below) for the LC3 machine and translates it to a binary file for the same machine.
- You can assume your .asm code does not contain information past Chapter 7 of our book, hence only the commands ADD, AND, BR, LD, LDI, LDR, LEA, NOT, ST, STI, STR, and TRAP are to be considered.
- We will not utilize user input or output (no GETC, PUTS, IN, etc).

Your .asm file can be assumed to be perfectly formatted.

- The file contains no tabs (all blank space is the "space" character)
- The first 6 characters of every line is a label. All labels are 6 character words and there are 2 blank spaces after every label. A line with no label starts with 8 blank spaces.
- The next 7 characters are for the opcode (LD, ST, etc) or a pre-processor directive. The "." **will be** the 9th character in the line when there.
- The rest of the line are the operands, which are comma separated with each comma touching the operand but followed by a single space character.

and your binary file should

- have the same name as the .asm file but .bin
- Contains an initial header effectively identical to the one from the .asm file
- Include in the first 8 spaces the hexadecimal line number of the instruction line followed by 2 spaces
- Clearly state the 16 bit instruction separated into components. For example (0001 001 001 0 00 001) vs (0010 010 000111111)
- The line of code that was translated should be an IN LINE comment after the binary instruction.

Finally, create a video (like my game video from last year CS 1083) which will explain, demonstrate, and present your program. Use software (like OBS) to a nice video of your team (on camera) overlaying on the clear presenting of your code and demonstrating it (5%). Two test programs will be provided to run on your program - one early on in the course and one the week before the project Due date. More details will be provided in D2L.