

Systems Internals and Assembly Language: Assignment 5, Fall 2006

For your final assignment, due in the last week of class at your presentation time (see below) you may choose from several options. You have two copies of p. 1 of this handout so that you can return one by **Wednesday, November 15, 2006.**

1. Student name _____

You may work individually or in a team. If your previous teammate wants to work individually, and you want to work in a team, I will match up people who would like teams. Just let me know. I also need to know because team projects need a team folder in Q:\StudentFiles, so a new team needs a new folder.

2. I plan to work G individually
 G as a team with _____

3. The topic that I would like to work on is—

4. Dr. Chase's signature approving of the topic _____(date)_____

If you choose a, b, c, or d below, the topic is automatically approved. Use an additional sheet of paper to tell me what your project will be if is not described somewhere in a–f below.

a. Any of the three topics on my third assignment that you didn't do for your third assignment report, **(1) Sort, (2) Bit-oriented programming, or (3) Display of floating point representation.**

b. Single-instruction computer (SIC). See your text, page IMD-2.20-10, Problems 2.54 and 2.55. Former students James Atlas and Kevin Young did as their project for this course a Java compiler that converts single-instruction computer assembly language programs into MIPS assembly language. So you should test your solutions to the two problems assigned by writing them in SIC and then compiling them to MIPS and running the MIPS to see if you get the result that you predict. Their Java program is found at Q:\InstructorFiles\Chase_Gene\Assembly\singleInstructionComputer. Notice that since a SIC only has one instruction, Atlas and Young decided not to write the instruction name `sbm` at all! Your report should include your solutions, summarize your findings, document your input and output, and address limitations—in other words, be like any other lab assignment.

c. Computer worm. Former student Jonathan Corbin did this as his project, and the sample MIPS code is at Q:\InstructorFiles\Chase_Gene\Assembly\worm.s. Write another one, documenting what you did and how you did it, and describe its runtime behavior.

d. Raw i/o. Former student Ben Cassell liked Appendix A.8 so much that his project was to write a MIPS program to demonstrate its ability to do what I'll call "raw" input and output. That is to say, that appendix section teaches you how to read from the keyboard and write to the console window without a syscall, therefore effectively showing you what a syscall does in part. Write a MIPS program that uses raw input and output, and capture the output in your report, describing what you did.

e. **A creative presentation of one of the topics of our course.** Former student Chad Blank did for his final project an animated demo of the single cycle model of a CPU, and even one multi-cycle model instruction. See the link to his work at November 27 on our course's web page, <http://www.messiah.edu/~chase/csc/281/index.html>

f. A **paper on** and a presentation of one of the **other topics in our textbook that I omitted for lack of time.** For example, the verilog hardware description language (Appendix B.4 and end of B.5, where there is a typographical correction that will help understanding; check textbook's website); or the combinational logic inside the ALU control (Appendix C.2); or finite state machines (B.10). Or write a small program in an Intel assembler, to show that you know how to write and assemble such a program. One year, former student Ed Bond actually did tic-tac-toe in Intel assembler for our course. Whew!

g. Another topic of your choice. I regard creativity in project topic as a good sign. You are welcome to discuss other topics with me.

Presentation schedule

To allow plenty of time for planning, here is the tentative presentation schedule. Each person or team should be prepared to spend 8–10 minutes describing their project to the class. This is tentative only because I don't know who the teams will be yet, so obviously there'll be some collapsing of positions, allowing some folks to present later than shown here.

Dec 4 Rosenshein, Steven

Dec 4 Pennington, Kyle

Dec 4 Blossey, Nick

Dec 4 Ashinghurst, Jeremy

Dec 6 Paul, Gavin

Dec 6 Patrick, Jonathan

Dec 6 Barcroft, Justin

Dec 6 Green, David

Dec 6 Fenstermacher, Adam

Dec 8 Sonne, Josiah

Dec 8 Thiebault, Chris

Dec 8 Hess, Cameron

Dec 8 Leslie, Ian

Dec 8 Mwangi, Gerald

o:\data\courses\assembly\a5_06.wpd