**Lab3 questions**

Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Section: \_\_\_\_

Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Section: \_\_\_\_

**Read and perform the lab guide** posted on the course website. Answer the questions in the lab guide as you get to them in the spaces below. The numbers below refer to steps in the lab guide.

* [3.5] What output do you expect when you run p7-loop.asm?
* [3.6] Is the actual output what you expected?
* [3.8] What output do you expect when you run p7-loop.asm?
* [3.9] Is the actual output what you expected?
* [3.11] If SwapMaxWithLast needed to return a value how would that be accomplished?
* [3.12] What changes would you need to make if SwapMaxWithLast needed more than 8 arguments?
* [3.13] What changes would you need to make if SwapMaxWithLast called another procedure?
* [3.14] The code in p7-loop.asm works almost like a procedure. Describe the changes you would make to convert the p7-loop.asm code into a sort procedure which is called and which in turn calls SwapMaxWithLast.
* [4.2] fib in fib.asm uses s registers to store both n and fib(n -1). Could t registers be used? Which would you choose?
* [4.4] While calculating fib(4), how many times is fib called?
* [4.5] In your fib code, you should have two internal calls to fib. Set a breakpoint at the line AFTER the first internal call to fib. Then, while calculating fib(4), observe the state of the stack when the breakpoint is hit (e.g. when recursion using the first jal fib is complete). Then, draw the state of your stack on the next page. Draw the stack as a rectangle, indicate each stack frame, and what register and value is present in each word.
* [4.5] What happens if fib does not restore the return address before using jr?

Draw your stack here