# CPSC 2500
# Computer Organization
# Homework 4 (100 points)
# Due: November 12, 11:59 PM, on Canvas

**NOTE:** Please code your answers to the following questions and submit it on Canvas by Nov 12 (Sunday). This assignment is to be done individually; you can discuss the questions with your classmates, but you should write your answers independently.

### 1. Multiplication (30 points)

Write an ANNA assembly program (mul.ac) to multiply two positive numbers and print their product. Your program should take the numbers for multiplication from user input. If the user enters 10 and 20 as inputs, then the program should print 200 (the product).

### 2. Log2 (30 points)

Write an ANNA assembly program (log2.ac) that asks the user for a number greater than zero and returns the base-2 logarithm of the number (always rounded up). For instance, if the user types in 64, the program should print 6 because 2^6 = 64. If the user types in 900, the program should print 10 because 2^9 (512) < 900 < 2^10 (1024). Print nothing if the user entered a zero or less.

Notes:

- Do not write this program with the various powers of two hard-coded into the problem such as a large chained if/else statement. Assignments that do this will get no more than 12 points on this problem.
- Instead, consider using a shift. A left shift by one bit multiplies a number by 2, a right shift by one bit divides a number by 2 (throwing away any remainder).

**3. Maximum (40 points)**

Write an ANNA assembly program (max.ac) that finds the largest number entered by the user. Initially, the program continually asks the user to enter numbers. As soon as a negative value is entered, compute which number is the largest. For instance, if the user entered 2, 6, 7, 6, 6, 7, 6, -1; the program should print 7 (largest number). If the user enters a negative number at the beginning, print 0.

Notes:

- The goal of this problem is to exercise storing, retrieving, and scanning the numbers stored in memory. Therefore the straightforward solution where you keep track of the max number in a register as the numbers are entered is not acceptable. Such a solution will receive a maximum of 18 points for this problem.
- You will not be able to keep track of everything in the input loop. Therefore, you will need to store all numbers entered by the user in memory.
- It suffices to print out only one number if there is a tie for the largest number.
- Store all numbers entered into a growing array.
- The array should be the last item in your data section so it can grow as large as necessary.
- You may assume there is enough memory to hold all numbers entered by the user.

Grading Requirements:

The assignment will be graded on functionality, though students are encouraged to follow recommended programming style as well. For information on proper style, refer to section 5 of the ANNA guide. Programs that fail to assemble will receive a zero.

Functionality will be based primarily on a suite of tests. A program that fails to successfully pass many of the tests will receive a very low score. Testing and debugging are important part of the programming process. Students who fail to test and debug their programs are handing in incomplete work and their grade will suffer as a result.

Submitting your Program:

Programs will be submitted electronically on Canvas. Please follow the steps carefully:

1. Ensure the assembly code files have the proper names:

- Program 1: mul.ac
- Program 2: log2.ac
- Program 3: max.ac

2. Submit the three assembly files on Canvas. Please submit all three files in a single zip file.

- You may resubmit your assignment. If you do, please resubmit all three files (even if you changed only one). Only the last submission will be graded.
- Canvas will only accept .zip files.