ECE 3504: Fundamentals of Computer Architecture

**Project 1**

You must work on this assignment **individually**. Your submission must be your own original work. Please follow the submission instructions posted on Canvas exactly.

**Late Policy:**Projects should be submitted via Canvas by 11:59 PM on the day they are due. Any time, even 1 second, late the syllabus late penalty is applied. Late submissions are subject to a penalty of 10% per each 24-hour time period.

**Note 1**: For this project, we may use Moss, a tool developed by Stanford University to detect similarities in code structure. Moss cannot be defeated by changes to variable and function names, function ordering, formatting changes, and comments. Any strong similarities flagged by Moss will be carefully examined and possibly submitted to the Office of Undergraduate Academic Integrity.

**Note 2**: You are responsible to ensure GTA can easily compile and run your source code, you must include basic instructions for running your code in your submission.

You will submit a zip file containing two files: your MIPS assembly solution (myCollatz.s or myCollatz.asm) and a PDF report. This assignment requires you to implement a MIPS assembly program. Your program must be tested in the QtSpim MIPS simulator.

Your PDF report must contain a short (1-2 paragraph) description of your program and screenshots (of the QtSpim console and/or register file) depicting the result of your program. Points will be deducted if the pdf report is missing or if the code is not commented sufficiently.

Your assembly code must run in QtSpim without any errors. Should your code produce errors/exceptions, please describe those errors in your pdf report.

**MIPS Programming Assignment**

The Collatz conjecture is an unsolved mathematical problem that can be understood by elementary school students. Starting from any give positive integer, it checks if repeating the following arithmetic operations will always converge to 1: for an odd number, multiply it by 3 before adding 1; for an even number, divide it by 2. See the following page for more details <https://mathworld.wolfram.com/CollatzProblem.html>

Write a MIPS assembly program that helps checking the Collatz conjecture. Your calculator will read in a positive integer from the console, and perform a single-step operation in the conjecture:

* If the number is non-positive, report an error message;
* Else
  + If the number is even, divide it by two and report the result;
  + If the number is odd, triple it and add one and report the result;

For example, if the given input is 7, then your program should report 22 as the result of the next step.

You are given a file project1.s as your starter code. It contains example code for printing messages and reading in inputs from the console.

**Bonus Question**

For up to **5% Extra Credit**, you may implement a program finding the minimum number of operations that lead to 1. Specifically, your program should report, for a positive integer,

* the sequence of results (called hailstone numbers) that eventually ends with 1;
* the number of steps to reach 1.

For example, if the given input is 6, then your program should report 3, 10, 5, 16, 8, 4, 2, 1 as the resulting sequence, and state that the number of steps is 8.

Additional information on MIPS assembly directives (*e.g.* .text, .data, .asciiz) that you may encounter in the file are given in Appendix A.2 and A.9 of the textbook. Please refer to Figure A.9.1 in the textbook for more information on reading/printing integers and characters from the console.

**Program documentation:**

* Use inline-comments following each instruction to explain its purpose.
* In addition, the header at the beginning of your program, it should contain the following documentation:

# Your name  
# Date: <last modified date>

# A description of what the program does in **pseudo-code**.

# A list describing use of each register used and its purpose in your assembly program