

Lab 1

Due: By 11:59 pm PDT on Monday, 10/2/2017.

Name:

Points Possible: 40

Prerequisite: Install the MARS MIPS simulator on your computer. It can be downloaded here:

<http://courses.missouristate.edu/KenVollmar/mars/download.htm>

Submission Instructions:

- Your code should be in an individual .asm file. Naming convention details provided within each part of this lab. "Part" refers to the two parts of this lab (1 and 2).
- Within each asm file, include your name (as a comment).
- Each part's code should be in its own file. So, for this lab, there should be two asm files.
- As a comment in your code, indicate the test cases used to further prove that your code works (in case I see it doesn't work for me, I can refer to your comment to see what went wrong).
- All files should then be compressed into a single zip or rar file. Compressed file name should indicate your full name, the Course Name, and the Lab #. E.g. "John Doe - CECS 341 - Lab 1.rar".

Failure to follow the above instructions will result in deduction of points, up to and including zero credit for your work.

Grading Guidelines:

- Does the program meet the requested requirements/criteria?
- Are the submission instructions followed?
- Is your code properly commented?
- Does your code compile?
- Does your code pass the test cases?
- Any closing of execution other than "-- program is finished running --" will result in deduction of points. This includes the "Dropped off bottom" message.
- MARS MIPS will be the compiler of choice when grading this exam. So your program must be able to be compiled using that.

Part 1:

(20 points)

Convert the below summation to MIPS:

$$\sum_{i=1}^N y \times i$$

Where N and y are inputs. N is the number of integers and y is the multiplier. You can interpret “y × i” as “y multiplied by i” or “y times i”.

Your code should be in a file with the following naming convention: **<Your Name>-Lab 1-Part 1.asm**

Sample console showing the input and output:

Number of integers (N)? 6

Multiplier (y)? 4

Sum = 84

-- program is finished running --

Part 2:

(20 points)

Create a MIPS program that will take a string input and output it in opposite order. Your program should handle a minimum of 50 characters in the string.

Your code should be in a file with the following naming convention: **<Your Name>-Lab 1-Part 2.asm**

Sample console showing the input and output:

Input a string: I love computer architecture!

!erutcetihcra retupmoc evol I

-- program is finished running --