1 Overview

Your assignment is to write a MIPS assembly language program that simulates the simple CPU we designed in lecture. The CPU has one register, and supports four operations: addition, subtraction, multiplication, and division. The program processes two arrays of bytes, one containing the inputs, and the other the operations. For each input the CPU performs the specified operation on the input and the contents of the register and puts the result back into the register. Each input byte is a 2’s-complement number (so it might be negative), and each operation byte is

- 0 for add, 1 for subtract, 2 for multiply, 3 for divide, or -1 for exit.

The initial value of the register is 0. If an attempt is made to divide by zero (0) then the simulator should print the string ‘Illegal operation: Division by 0.’ and exit.

spim and xspim can be found in /home/cs340/spim6.2.

2 Assignment

Write a MIPS assembly language program that simulates the CPU as described. The file /home/cs340/prog1/sim.s contains a skeleton program to get you started. The inputs array contains the input values and the operations array contains the operating codes. When your program finishes the value in register should be the result of the computation.

3 Makefile

A sample makefile for your use can be found in file /home/cs340/prog1/Makefile. SPIM will run sim.s directly, but the Makefile makes it easy for you to turn in your assignment when you’re done.

4 Turnin

When you have completed the program, submit your sim.s file by typing ‘make turnin’. Or, to do it manually type ‘turnin cs340prog1 sim.s’. You may turn in your sim.s file as many times as you want; turnin will always replace the previously turned-in version with the new version. Leave inputs, operation, and count set to the values in the skeleton sim.s file. We will test other values during grading.