Lab Assignment 1: Performance Measurements

Work Breakdown:
Question 1 : Harshita Huria (1000980398)
Question 2 : Angel Serah (1000491773)

1) State the % performance drop versus an ideal pipeline with CPI of 1.0 for the two questions. Briefly describe the mathematical derivations used to arrive at these answers

We calculate the CPI w/hazard = 1 + 1*sim_num_one_cycle_stall/sim_num_insn
+2*sim_num_two_cycle_stall/sim_num_insn

-sim_num_insn is the total number of instructions
-sim_num_one_cycle_stall is the number of instructions with one cycle stall
-sim_num_one_cycle_stall is the number of instructions with two cycle stalls

Q1) % Slowdown = 1/Speedup = (CPI w/hazard - ideal CPI)/ideal CPI
= (1.6642-1)/1
= **66.42% drop**
Q2) % Slowdown = 1/Speedup = (CPI w/hazard - ideal CPI)/ideal CPI
= (1.3903-1)/1
= **39.03% drop**

2) Briefly explain how your microbenchmark collected statistics validate the correctness of your code for the first problem statement. Feel free to refer to comments within the mbq1.c file, as needed. Specify which compilation flags you used

The microbenchmark was compiled using the command:
/cad2/ece552f/compiler/bin/ssbig-na-sstrix-gcc mbq1.c -O0 -o mbq1

We manually calculated how many stalls both questions would have and they were a match with the statistical values. In the comments within the microbenchmark, we have remarked on the number of stalls per instruction, for both questions. We used the values given below to verify the correctness of our code:

sim_num_one_cycle_stall_q1 1000079 # total number of 1 cycle stall instructions for question 1
sim_num_two_cycle_stall_q1 10000854 # total number of 2 cycle stall instructions for question 1
sim_num_one_cycle_stall_q2 8000770 # total number of 1 cycle stall instructions for question 2
sim_num_two_cycle_stall_q2 2000091 # total number of 2 cycle stall instructions for question 2

Note: Our microbenchmark consisted of a set of inline assembly instructions executed in a loop of 10^6. When our microbenchmark did not have any hazards,(it consisted of 1 ASM instruction : addi $1, $0, 20) the following were the baseline 1 cycle and 2 cycle RAW hazards for both questions:

sim_num_RAW_hazard_q1 6000933 # total number of RAW hazards (q1)
sim_num_RAW_hazard_q2 5000861 # total number of RAW hazards (q2)
CPI_from_RAW_hazard_q1 2.0904 # CPI from RAW hazard (q1)
CPI_from_RAW_hazard_q2 1.5452 # CPI from RAW hazard (q2)

These hazards were purely a result of the for loop and faults in libraries etc.
We verified that there were additional RAW hazards by dumping the .s file and examining the assembly instructions. Attached are a few examples of RAW hazards within the loop, which were not added by us:

```
        lw      $2,16($fp)
        li      $3,0x000f423f           # 999999
(RAW)   slt     $2,$3,$2
(RAW)   beq     $2,$0,$L5
```