**[100 points]**

**Teams:** Form groups of 4 each.

**Task:**

Using the ARM simulator (ARM, write code that computes addition, subtraction, and multiplication of floating point numbers without using ARM floating point operations or registers but still using IEEE-754. Given 2 numbers in the decimal number format diagrammed below, convert to IEEE-754, compute the 3 results, and compare your results against ARM’s embedded floating point functions. Also, compute the CPI of each function.

Format: 1 sign bit, 15 binary bits for integer part, 16 binary bits for fraction part = 32 bits

(not two’s complement)

Range: +32727.65535 to -32727.65535

+100.000 = 0 000 0000 0110 0100 . 0000 0000 0000 0000

-100.000 = 1 000 0000 0110 0100 . 0000 0000 0000 0000

-32727.65535 = 1 111 1111 1111 1111 . 1111 1111 1111 1111

+32727.65535 = 0 111 1111 1111 1111 . 1111 1111 1111 1111

0 = 0 000 0000 0000 0000 . 0000 0000 0000 0000

**Proposed steps:**

1. Read 2 numbers in format above from memory
2. Convert numbers to IEEE-745 and store in memory
3. Do ADD, SUB, and MUL on the numbers
4. Estimate CPI of your code
5. Use the numbers from step 2 and compute FP solution of ADD, SUB, and MUL using ARM’s FP operations.
6. Estimate CPI of the ARM code

**Schedule:**

Sunday, 4 Dec, 11.59pm: Assembly code due on Blackboard.

Monday, 5 Dec, 11.59pm: No changes allowed after submission.

Documentation due on Blackboard as PDF

Tuesday-Friday 6-9 Dec: 10 min demonstration, in Charlie’s office

**Grading:**

Source code: 30%

* Commented

Oral presentation & demonstration: 30%

* 10 minutes
  + Approach, Code discussion, Issues and solutions, Results, and Demonstration

Final report: 30%

* CPI computations
* All implemented algorithms must be described.
* Show sample input and output data
* Show ARM floating point results and explain the difference.

Peer review: 10%