

Homework 1Due: Sep 8th, 2025

Please write legibly or type. Show your work to the extent possible.

1. (40 points) Translate the high-level language code below into assembly instructions. The variables A, B, C, D, E and F are located in the memory and can be accessed by their label (e.g., LOAD R1, A will load A from the memory into R1). Minimize the number of instructions in the assembly code that you write.

$$F = (A-B)*(C+D)/(E-D)$$

- a) Write the code for an accumulator architecture
- b) Write the code for a stack architecture. Assume that the division (subtraction) operation divides (subtracts) the topmost value in the stack by the second topmost value.
- c) Write the code for a register-memory architecture
- d) Write the code for a load-store architecture
- e) Compare and count the number of instructions and memory accesses between the different ISAs in the previous parts of the questions (a, b, c and d).

2. (30 points) Some architectures support the ‘memory indirect’ addressing mode. Below is an example. In this case, the register R2 contains a pointer to a pointer. Two memory accesses are required to load the data.

ADD R3, @(R2)

The MIPS CPU doesn’t support this addressing mode. Write a MIPS code that’s equivalent to the instruction above. The pointer-to-pointer is in register \$t1. The other data is in register \$t4.

3. (30 points) Memory Alignment, Big Endian vs. Little Endian: Write C language program to show how your computer stores the 32- bit integer 0x12131415 and the float 34.73. Your program should print byte per line (i.e. byte 0 = 0xAA, byte 1 = BB ... etc.)