(1) Complete the following figure by specifying number systems we discussed in the classroom.

(2) Transform the following decimal number to the two’s complement binary number (using the 16-bit format: your processor is a 16-bit architecture processor): \(-81_{(10)}\). Show all your work.

**Construct the bit pattern for positive 81**: \(81_{(10)} = 101 \ 1110_{(2)}\)

**Translate the pattern to a 16-bit pattern**: \(81_{(10)} = 0000 \ 0000 \ 0101 \ 1110_{(2)}\)

**Invert the bit pattern**: \(1111 \ 1111 \ 1010 \ 0001\)

**Add binary 1**: \(1111 \ 1111 \ 1010 \ 0001\)
(3) What do “lw” instruction does (performs)?

The “lw” instruction accesses a memory address, grabs the 64 bits there, and copy the 64 bits into a register.

(4) What is the difference between “li $a0, 1024” and “la $a0, 1024” instructions? Assume that this computer system is a 32-bit system (i.e., all the registers are 32-bit registers and its ALU can deal with up to 32-bit inputs and outputs).

- The “li” instruction translates a constant integer as a two’s complement integer to a 64 bit pattern, and load the 64 bits to a register
- The “la” instruction translates a constant integer as an unsigned integer to a 64 bit pattern, and load the 64 bits to a register

(5) Translate the “if-then-else” program structure using MIPS instructions (in the following MIPS assembly program structure (by showing all the necessary missing instructions there).