(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.

\[ 81_{(10)} = 0000 0000 0101 0001 \]

Invert the binary bit pattern: \(1111 1111 1010 1110\)

Add one to the bit pattern: \(1111 1111 1010 1111\)
(3) Which numbering system solves the two problems in the “sign magnitude integers”?

Two’s complement number does.

(4) “li $t0, (1024)” is an illegal instruction (if you try to assemble that instruction using PC-SPIM simulator, that instruction will cause a syntax error). What’s wrong?

“(1024)” in the instruction means “the contents of the memory address at 1024”. Since “li” instruction does NOT access memory address, but it takes a constant, instead, “li $t0, (1024)” should be “li $t0, 1024”.

(5) 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 first instruction, “li $a0, 1024” encodes “1024” based on two’s complement integer.
- The second instruction, “la $a0, 1024” encodes “1024” based on unsigned integer.