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

(2) How will \(-1_{(10)}\) be represented using \(n\)-bits two’s complement number?

“1” for all of the \(n\) bits (e.g., “1111 1111 1111 1111” for 16-bit processors)

(3) Given \(N\) bits for 2’s complement integers, what is the largest positive integer (in decimal) a processor can have and what is the smallest negative integer (in decimal) a processor can have?

\(2^{(N-1)} - 1\) (e.g., if \(N = 16\), it’s \((2^{(16-1)} - 1) = 32,767\))

(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 former (li) encodes a given integer immediate based on the 2’s complement integer, while the latter (la) encodes a given integer immediate based on the unsigned integer.
(5) There is a logic error in the following if-else structure implemented in MIPS R3000 assembly language. (a) What is the error (describe the problem). (b) How can we fix the problem?

```
.text
.globl main
main:
    beq $t0, $t1, if_equal
    instruction A
    ...
if_equal:
    instruction B
    ...
end_if_else:
    instruction C
    ...
    jr $31
```

We need a “j (unconditional jump)” instruction at the end of the group of the instructions for “instruction A”. Otherwise a processor will execute the instructions for both of the two cases (if the condition is satisfied and if the condition is not satisfied – instead of either of them).