Questions 1 through 7 refer to the completed single-cycle datapath design, reproduced below, which supports execution of the following MIPS instructions: add, sub, and, or, slt, lw, sw, beq and j.

For the following questions, we would say that:

- A signal must be set in a certain way for a given instruction if and only if the instruction could not be guaranteed to yield correct results unless the signal was set in that way, no matter how the other signals are set.
- It does not matter how a signal is set for a given instruction if and only if that instruction would always be guaranteed to yield correct results, no matter how the signal was set, so long as all the other signals are set appropriately.

1. [10 points] For which of the supported instruction(s) must the control line Jump be set to 0? Explain why.
2. [10 points] For which of the supported instruction(s) must the control line MemRead be set to 0? Explain why.
3. [10 points] For which of the supported instruction(s) must the control line RegWrite be set to 0? Explain why.
4. [10 points] For which of the supported instruction(s) must the control line MemtoReg be set to 0? Explain why.
5. [10 points] For which of the supported instruction(s) must the control line ALUSrc be set to 1? Explain why.
6. [15 points] For which of the supported instruction(s) does it not matter how the control signal MemWrite is set? Justify your answer precisely.

7. [15 points] Suppose that an implementation of the datapath is created in which a defect causes the MemtoReg signal to always be 0. Which of the supported instructions would be affected, and how?

8. The designers of the MIPS32 instruction set made the decision to include and exclude the following possible bitwise logical instructions from the basic instruction set:

    # include these                      Effect for $i = 0:31
    or $rd, $rs, $rt       # GPR[ rd[i] ] = GPR[ rs[i] ] OR GPR[ rt[i] ]

    # exclude these
    not $rd, $rs            # GPR[ rd[i] ] = NOT GPR[ rs[i] ]

Recall your Boolean algebra from Discrete Mathematics. Also, recall that NAND means NOT AND, and that XNOR means NOT XOR.

Now, most high-level programming languages, including C, will include one or more of the operations that will be excluded from the MIPS machine language, so a compiler must be able to translate all of the excluded instructions into an equivalent form using only the basic instructions.

a) [5 points] Which of the bitwise logical instructions that are not supported in the MIPS design are included in C?

b) [15 points] For each instruction you identified above, give an example of C code, and show an equivalent translation into basic MIPS machine/assembly instructions.