# Homework 2

**Due Date: March 7**

All written homeworks are due at the beginning of class on the due date. Write your answers on plain or lined paper or type them up using a word processor. There is no provision for homework turned in late. Points are scaled relative to the "time" value given in the book or below (the number in square brackets next to the problem). Show your work for maximum partial credit. Be as neat as possible; if I can't read it, it's wrong. If your homework is longer than 1 page, please staple or paper-clip the pages together.

It is not guaranteed that all problems will be graded. We will either cover answers in class and/or a solution sheet will be posted.

At the end of your homework, please write and sign the Honor Code pledge:

**I have abided by the Wheaton College Honor Code in this work.**

Do the following problems from the text; note that if the question asks for an answer using C, the answer will be the same in C++. Note, also, that in any problem that references MIPS, you must use **true** MIPS assembly instructions, which means **no pseudo-instructions.** In the text, SPIM refers to the assembly language that includes pseudo-instructions and MIPS refers to the true language.

2.3, 2.4[1], 2.5, 2.10.1–2.10.4, 2.19, 2.20

In addition (be sure to look up each instruction's format first, where applicable), do the following problems. In those that show MIPS code, do not assume that variables are contained in certain registers; rather, you have to move data into or out of memory.

1. [10] Show the single MIPS instruction or minimal sequence of instructions for the following C++ statement. Assume that `x` is a 32-bit integer, `A`, and `B` are arrays of 32-bit integers, and all variables have already been defined/allocated in memory.

   ```
   x = A[B[x]];
   ```

2. [10] Show the single MIPS instruction or minimal sequence of instructions (no pseudoinstructions) for the following C++ statement:

   ```
   A[4] = A[5] + x;
   ```

   Assume that `x` corresponds to register `$t3` and the array `A` has a base address of $10,518,543_{10}$.[2]

3. [5] Show the single MIPS instruction or minimal sequence of instructions for this C++ statement, where `a` and `b` are integer variables:

   ```
   b = 45 & a;
   ```

4. [10] Give the binary instruction equivalent of the MIPS instruction: `lw $s1, number($t2)`, where the base address of `number` is 112. Show all of the fields (i.e., don't just write the 32-bit binary value, but show the result as a "box" as done in class).

5. [10] Write the binary instruction equivalent of the MIPS instruction: `sllv $t1, $s4, $t7`.

6. [10] Find a reduced sum-of-products version of the following expression:

   $AB + BC + A\overline{B}\ \overline{C} + \overline{A}BC + \overline{A}\ \overline{B}C$.

   The use of a Karnaugh map is suggested but not required.

7. [5] Draw the gates required for the expression found in the previous problem.

8. [5] DeMorgan's Laws indicate that $\overline{BC} = \overline{B} + \overline{C}$. Prove that the expression $\overline{B}\ \overline{C} = \overline{B} + \overline{C}$ is **not** true.

[1] They *really* mean **one** C++ instruction.

[2] Hint: This is not as straightforward as you might think. The size of the addressing portion of the instruction format is important! Also remember that `li` is a pseudoinstruction, and therefore not available in MIPS.