**Questions - CompEx 1**

**Name:**  Mark Demore  
  
**Section:**  M1A  
  
**Documentation:** None

**Scenario**

1. (20pts) What's the purpose of the mystery program? Don't repeat what each step does, give me a **concise explanation of purpose**. It should only be one sentence.   
     
   The program simply keeps an old value in memory, then adds the old value to the new value, starting with 1 and 0, 10 times.
2. (10pts) If you wanted to change the program to make it more readable and understandable, what changes would you make? **Describe at least two *specific* changes you would make**.   
     
   I would get rid of the indexed addressing with 0 offset and replace it with just register direct addressing. I would also get rid of all of the lines involving r9, as its value seems to be irrelevant.
3. (10pts) If you were an attacker and wanted to obfuscate (make unclear) the purpose of your code, what changes would you make? **Describe at least two *specific* changes you would make**.   
     
   I would change the "loop" label to something different, perhaps the "forever" label as well, so it was unclear where and why the program was jumping. I would also make better use of the indexed addressing modes so that it was less clear which registers are being used.

**General**

1. (10pts) What's the purpose of a breakpoint? Why are they useful? Give an example of a situation where you'd use a breakpoint.   
     
     
   Breakpoints stop the debugger at a certain line so that variable or register values can be checked at that instruction to ensure the intended procedures are being executed. They are useful in locating the specific line or section of code that is causing the bug. A good use of a breakpoint would be when a program is outputting an appropriate value and then an incorrect value after user input; a breakpoint could be inserted in the section that processes the user input to make sure it is formatted properly before being used.
2. (5pts) Consider the forever jmp forever instruction from the simple program you wrote first. What type of instruction is this? How is the line of code used in your program, and why is it needed?   
     
   This is a jump instruction. This line is used to keep the program running to analyze final values once it is finished executing the main body of code. Without this line, the program would finish executing entirely and stop running, preventing analysis of register values and other forms of "output".
3. (5pts) List the different addressing modes used in the mystery code. List and give an example instruction from the code for each addressing mode used.   
     
   Register Direct – mov.w r12, r13  
   Immediate – mov.w #1, r12  
   Indexed – mov.w r12, 0(r9)
4. (10pts) Are the values in memory stored in little-endian or big-endian format? Using the debugger, how can you tell? If they were stored in the other format, what would it look like? Give me an example from your code. **Be specific on *where* and *how* you find the answer in Code Composer.**   
     
   It is stored in big-endian format, as found in the memory browser in the debugger. For example, the value of R9 is 0x0208,and in the memory browser is displayed as 0x0208 at the location of R9. If it was stored in little-endian format, the value of R9 would appear as 0x0802.
5. (20pts) Write a program that fills RAM (0x0200 - 0x03ff) with words whose *values* start at 0 and count upward by 2. Use CCS to verify that it works, and **explain your verification process.**   
     
   To verify my program, I ran the debugger and looked at RAM in the memory browser. Once I saw that the words increased by 2 at each memory location, the program was verified.
6. (10pts) Go to bitbucket.org and create an account there if you do not have one already. This site will be where you turn in your labs for this class. Create one and only one repository for this class (use a descriptive name, such as ECE382-YourLastName). For each lab, you will create a separate folder under this repository (with a descriptive name, such as Lab1-YourLastName). Your repository must be private, and **you must add your instructor with read permissions to the repository before this assignment is due in order to obtain full points for this portion.**