University of California, San Diego
Dept. of Computer Science and Engineering
CSE 30 – Computer Organization and Systems Programming
Problem Set #4

1. Redo any problems that you missed on Exam #1

2. **(10 points)** Translate the following code into ARM assembly:

   A[i] = B[i-1] + B[i] + B[i+1]

   Assume that *A* and *B* are byte arrays. Both arrays are stored in memory. Register r1 contains the initial address for array *A*; register r2 contains the initial address for array *B*; and register r3 contains the value of *i*. You should not overwrite the values of A, B and i.

   Assume that load and store instructions take 3 cycles each and all other instructions take 1 cycle.

   How many cycles does your code take? How many additional registers does your code require?

3. **(5 points)** Replace the C code for

```
if (x < y)
     x = y + 2;
else
     y = x - 4;
```

   by a set of equivalent ARM instructions. Use as few instructions and additional registers as possible. Assume x and y are in registers r0 and r1, respectively. Comment your code.

4. **(10 points)** Implement the following C code in ARM:

```
for (int i=0; i != 10; i++) {
     a[2i] = b[i] + i;
}
```

   Assume that the base address at a[] is stored in r0, and the base address of b[] is stored in r1, and that a and b are int arrays. Do not overwrite the values of a, b and i. Comment your code.

   How many static instructions do you have? How many dynamic instructions will you execute? How many cycles will it take to execute your program? Assume that multiply takes 3 cycles, load and store take 5 cycles and all other operations take 1 cycle.

5. **(10 points)** Here is some assembly code. Disassemble it (i.e., write the equivalent C code) and describe, in English, the function of this code.

```
            MOV   r0, #0
Loop:       CMP   r1, #0
            BEQ   End
            AND   r2, r1, #1
            ADD   r0, r0, r2
            MOV   r1, r1, LSR #1
            B     Loop
End:
```