**Homework 3**

Trevor Lund, CprE 381

1. Summary
   1. Saved registers ($s0 - $s7), Stack pointer ($sp, $fp, $gp), Return address ($ra)
   2. Register addressing (add), Base addressing (ls), Immediate addressing (addi), PC-relative addressing (beq), pseudo-direct addressing (j)
   3. Formats and Fields:
      1. R: OP code, rs, rt, rd, shamt, funct
      2. I: OP code, rs, rt, imm
      3. J: OP code, target
2. Code Translation
   1. D, because if the first equal command is true, it loops, making an or condition, and the “or” operand in C is “||”.
   2. Addi $s0, $0, 4400000

Lw $s1, 16($s0)

Addi $t0, $0, 0

Addi $s2, $0, 0

Beq $t0, $t3, Skip

Loop: add $s2, $s2, $s1

Addi $t0, $t0, 1

Bne $t0, $t3 Loop

Skip: sw $s2, 4($s0)

* 1. Addi $t0, $0, 0

Lw $t1, 0($a3)

Loop: sll $t2, $t0, 2

Add $t3, $t2, $a0

Add $t4, $t2, $a1

Lw $s1, 0($t4)

Add $s1, $s1, $s0

Sw $s1, 0($t3)

Beq $t0, $t1 Exit

Addi $t0, $t0, 1

J Loop

Exit:

11 instructions were used, and 3 memory references were made.

* 1. F: addiu $sp, $sp, -32

Sw $ra, 28($sp)

Sw $fp, 24($sp)

Addiu $fp, $sp, 28

Addiu $a0, $a0, $0

Addiu $a1, $a1, $0

Addiu $a2, $a2, $0

Jal func

Addiu $a0, $v0, $0

Addiu $a1, $a2, $0

Lw $fp, 24($sp)

Lw $ra, 28($sp)

Addiu $sp, $sp, 32

Jr $ra

Func: addiu $v0, $a0, $0

Addiu $v1, $a1, $0

Jr $ra