Peter Woodman
0127119
CSE 410, Sp. 05 - H. Levy
Programming Assignment 1

1. `li` is replaced by `ori`, and `move` is replaced by `addu`. The address of newline is converted to an immediate and the call is changed to `lui`.

2. `$ra` (return address) is changed to `0x00400018`, indicating the return address to jump back to (note that this is one instruction past the `jal` instruction).

3. Twelve times.

4. `$t0` = `0x0000000f`
   `$s3` = `0x0000000f`
   `$s0` = `0x000000e9`
   `$s1` = `0x00000179`

5. By removing the `beq` at the top and changing the `b` at the bottom to a `bne`. Code attached at the end.

## CODE SNIPPETS

*-fibonacci.s-*

```
# First SPIM Assignment
    # Program 1
    # Name: Peter Woodman
    # Class: CSE410
    # Date: 04072005
# Start with data declarations
#
    .data
str1: .asciiz "Here are the first 15 Fibonacci numbers:\n"
newline: .asciiz "\n"

.align 2  #align next value on word boundary

.globl main
    .text   #begin the text segment

main:   la $a0, str1
    li $v0, 4
    syscall

    li $s0, 0   #s0 = F_(n-2)
    li $s1, 1   #s1 = F_(n-1)
    li $s3, 2   #s3 = n

    #print the first two numbers of the sequence manually

    move $a0, $s0
    li $v0, 1
    syscall

    la $a0, newline
    li $v0, 4
    syscall

    move $a0, $s1
    li $v0, 1
    syscall

    la $a0, newline
    li $v0, 4
    syscall

    li $t0, 15

loop:   add $t1, $s0, $s1   #MODIFIED
    move $s0, $s1
    move $s1, $t1

    move $a0, $s1
    li $v0, 1
    syscall

    la $a0, newline
```

```
    li $v0, 4
    syscall

    addi $s3, $s3, 1

    bne $t0, $s3, loop  #MODIFIED


end:    li $v0, 10
    syscall
```

*-SPIM console output-*

```
Here are the first 15 Fibonacci numbers:
0
1
1
2
3
5
8
13
21
34
55
89
144
233
377
```