CS/COE 0447 Fall 2009
Lab 5: More Functions
Due Date: October 15, 2009

To get started on this lab, attend recitation on 10/9. Each of you should submit your own solution, according to these instructions: http://www.cs.pitt.edu/~sab104/teaching/cs447/submission.html. You may collaborate with your partner, but each person must turn in their own copy of the lab, with the name of their partner. The lab is due on 10/15 at 11:59pm.

In this lab, we will write two functions that use an activation record (a.k.a. activation frame or stack frame) to save registers, local variables and the return address. To get started writing the functions, use the following template of a safe function: http://www.cs.pitt.edu/~sab104/teaching/cs447/labs/func.txt. Notice that this is one (safe) way of writing functions and that there are other (possibly more efficient) ways of doing it.

1. Translate the following function to MIPS assembly. You must use the template presented above:

```
int addition( int x, int y ) {
     int z;
     z = x + y;
     return z;
}
```

To test your function, use the following main program:

```
     .text
     li $a0, 4       #Load first parameter
     li $a1, 5       #Load second parameter
     jal addition    #Make call
     move $a0, $v0   #Print return value
     li $v0, 1
     syscall
     li   $v0, 10    #Exit
     syscall
```

2. Translate the following Fibonacci function to MIPS assembly. You don't have to use the template, but doing so will make things much easier.

```
int fib(int n) {
    if (n <= 1) {
        return n;
    } else {
        return fib(n-1)+fib(n-2);
    }
}
```

To test your function, use the following main program:

```
.text
li $a0, 5      #Load first parameter
jal fib        #Make call
move $a0, $v0  #Print return value
li $v0, 1
syscall
li   $v0, 10   #Exit
syscall
```