# Assembly Language Programming

## Lab 5:

Purpose

Learn to how to develop programs using memory-mapped I/O

What to Hand In

Completed PrintString.asm, ReadString.asm

1. Using memory-mapped I/O, write a function that will perform the same task as the “Print String” System function. Write a main program to test this function. Make sure you activate the memory-mapped I/O feature of SPIM in the settings window before you load your program.
2. Using memory-mapped I/O, write a function that will perform the same task as the “Read String” System function. This function will get characters from the keyboard and send them to the terminal as well as placing them in a buffer pointed to by register $a0. The contents of register $a1 specifies the length “n” of the buffer. It reads up to n–1 characters into the buffer and terminates the string with a null byte. If fewer than n-1 characters are on the current line, it reads up to and including the newline (Enter Key) and again null-terminates the string. Write a main program to test this function. Make sure you activate the memory-mapped I/O feature of SPIM in the settings window before you load this program.