CSCI-21 Lab #9, due data changed to 5/2/16

As usual, please work in groups of two.

Using memory-mapped I/O and polling, not syscall, write a MIPS program that prompts the user for two integers, and then prints the numbers and their sum and difference (first – second).

Make a subroutine to prompt for the number that takes the address of the prompt string and returns the number the user entered. Make a second subroutine to print the value that takes a string (description of what is printing) and the number to print, and return no value. Use the full "real-world" stack-frame calling convention. You may not use syscall for any I/O for this lab. You may make second-level subroutines to print strings or integers, or just get the values.

Here's an algorithm to get a value from the keyboard into an integer (assuming no leading whitespace):

```
t = 0   // total
s = '+' // an indicator for positive or negative
get character c
if( c is '-' )
    s = '-'
    get character c
endif
while( c is a digit )
    t *= 10
    t += c - '0'
    get character c
end loop
if( s == '-' )
    t = -t;
endif
```

To print an integer n:

```
if( n < 0 )
    print '-'
    n = -n
endif
have an array of characters available, initially all '\0'
do
    q = n / 10
    r = n % 10 (these can be done with one division)
    put r + '0' into the array and advance to next position
    n = q
while( n > 0 )
print characters in reverse order, then print a '\n'
```