CS061 – Programming Assignment 02

Objective	The purpose of this assignment is to further familiarize you with the basic LC3 instructions, some simple Two's Complement procedures taught in lecture, and basic Input/Output.

High Level Description	Subtract one number from another and output it to the console.

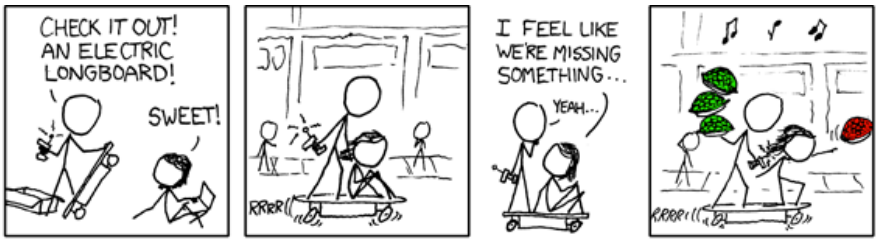
Atomic-Level Breakdown	This assignment is comprised of five tasks: 1. Read two numeric characters *(i.e. '0' ... '9')* from the user using Trap x20, storing them in separate registers 2. Convert the numeric characters into the actual numbers they represent (i.e. convert '7' into 7) 3. Subtract the second number from the first 4. Determine the sign (+/-) of the result - if negative, determine the magnitude of the result 5. Convert resulting number back to a printable format and print it, together with minus sign if necessary

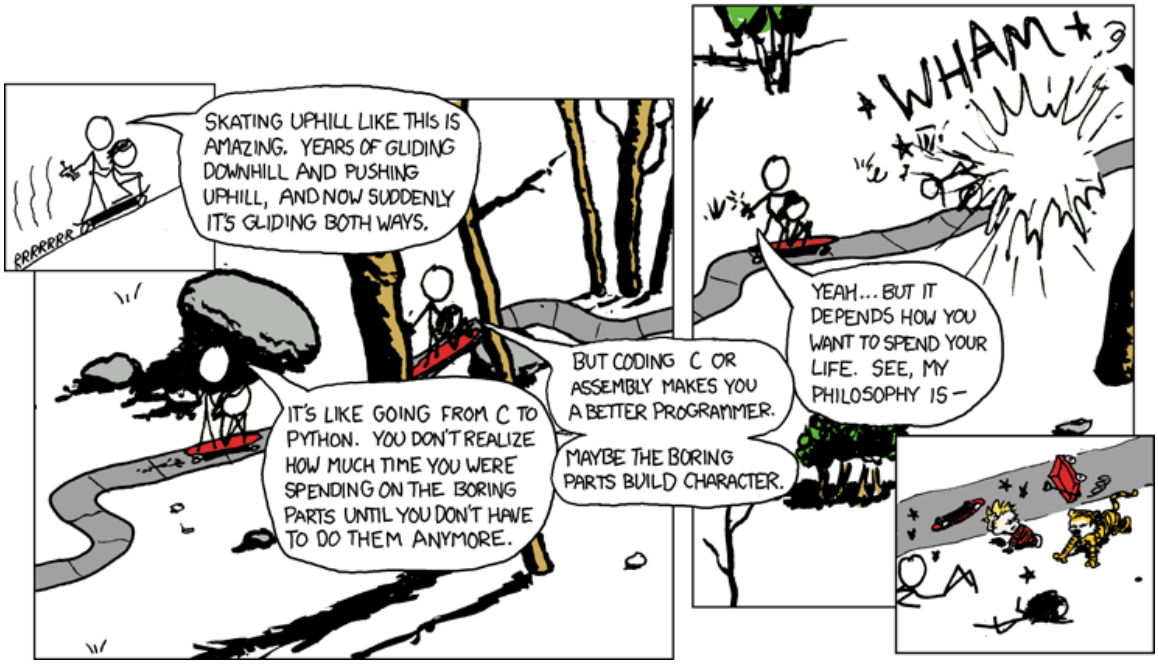
Example	i. Program starts: user enters '7' Program converts '7' into 7 and stores it in a register ii. User enters '5' Program converts '5' into 5 and stores it in another register iii. Program performs (7-5) and stores the result, 2, in a register iv. Program converts 2 into '2', and stores it in a register Program outputs '2'

Uh…help?	• The instruction "Trap x20" will *always* store the input character into R0. You cannot specify any other register. • The instruction "Trap x21" will *always* print whatever ASCII code is stored in R0. You cannot specify any other register. • If the user enters '7', the value stored into R0 is b0000 0000 0011 0111 (= x0037 = #55), ***not*** b0000 0000 0000 0111 (= #7). Go to www.asciitable.com and see why (conversion between a *character* and the *number it represents* will be used repeatedly in this course, so make sure you understand how to do it now!!)

	• To take the Two's Complement of a number (i.e. to make a positive number negative or vice versa): - Invert the bits *(what assembly instruction does this?)* - Add one • A neat trick in LC3 to copy the value of one register directly to another: ADD R5, R6, #0 ; R5 ← (R6) + 0, i.e. R5 ← (R6) • If the result is negative, remember that you will have to print <u>*two*</u> characters, not one (there is no ASCII character for '-1', yah?) • If you are struggling with writing LC3 code from scratch, try writing the program out in pseudo-code or even C++ first. Then, your only task is to convert the logic/code into LC3. Historically, many students have found this to be very helpful.

Comics??! Sweet!!!





Source: http://xkcd.com/409/

Rubric 	• Code does not assemble: -10 points (no reshow) • Assignments with no header: -5 points (if we can figure out who you are!) • Well commented code: +2 points • Correct use of TRAPs: +2 points • Correct ASCII to representative number conversion: +2 points • Correct Two's Complement conversion: +2 points • Correct output: +2 points