CS061 - Programming Assignment 1

Objective

The purpose of this assignment is to familiarize you with the basics of LC-3 assembly language programming, the SIMPL emulator, and rudimentary debugging.

Your Tasks

Implement the LC-3 program from the image below.

Note: This is similar to, but not *quite* the same as, the program you worked on in Lab 1: it uses a DO-WHILE loop to multiply two numbers together.

First (as with all your labs and assignments) go to the Lab and Assignment folder in Piazza, and download the assignment 1 zip file to your assignments/assn1 folder in your cs account, and unzip it in place.

Type the code from the image into your assn1.asm file *(again, the registers are assigned differently from your Lab 1 exercise, so **don't** copy that!)*:

```
gedit assn1.asm &
```

and then launch & run simpl:

```
simpl assn1.asm &
```

***Remember**: whenever you run simpl, you must ALWAYS keep the Text Window open so you can see warnings & error messages from the emulator!*

When the emulator opens, place a *breakpoint* at the beginning of the DO-WHILE loop by right-clicking that line of code and selecting “Mark as Breakpoint”.

Lastly, create a table to record the register contents for *each* register (R0 through R7) as follows:

- Before entering the loop (i.e. the first time the program halts at the breakpoint)
- After each iteration of the loop (each subsequent breakpoint halt)
- Note that since your breakpoint is at the start of the loop, the register values after the *last* iteration will just be the values when the program has ended, as suggested by the naming of the last row in the image below.

How many rows do you think you will end up with in your table?

Record the table as a block of comments beneath your header and above the actual LC-3 code, just like the table in the image below *(but obviously with the actual values from ALL your registers for ALL iterations of the loop!)*

The following program performs the action:

`R3 <-- 6 * 12  ; (i.e. multiply 6 by 12 and write the result into Register 3)`

using the equivalent of a DO-WHILE loop:

```
;---------------------------------------------------
; Name: Hayao Miyazaki
; Login: hayam
; Email address: hayam@cs.ucr.edu
; Assignment: assn1
; Lab Section: <021 or 022>
; TA: Sean Foley
;
; I attest that this code was totally given
; to me and that I didn't come up with
; any of it  =P
;---------------------------------------------------

; ------------------------------------------------------------------------------
; REG VALUES      R0  R1  R2  R3  R4  R5  R6  R7
; ------------------------------------------------------------------------------
; Pre-loop        x   x   x   x   x   x   x   x
; Iteration 01    x   x   x   x   x   x   x   x
; Iteration 02    x   x   x   x   x   x   x   x
; ...
; Iteration n     x   x   x   x   x   x   x   x
; End of program  x   x   x   x   x   x   x   x
; ------------------------------------------------------------------------------

.ORIG x3000                     ; Program begins here
;-----------------
; Instructions
;-----------------
LD   R1,  DEC_6                 ; R1 <-- 6
LD   R2,  DEC_12                ; R2 <-- 12
LD   R3,  DEC_0                 ; R3 <-- 0

DO_WHILE    ADD R3,  R3,  R2    ; R3 <-- R3 + R2
            ADD R1,  R1,  #-1   ; R1 <-- R1 - 1
            BRp DO_WHILE        ; if ( LMR > 0 ) goto DO_WHILE

HALT                            ; Terminate the program
;---------
; Data
;---------
DEC_0   .FILL    #0             ; Put the value 0 into memory here
DEC_6   .FILL    #6             ; Put the value 6 into memory here
DEC_12  .FILL    #12            ; Put the value 12 into memory here

.END
```

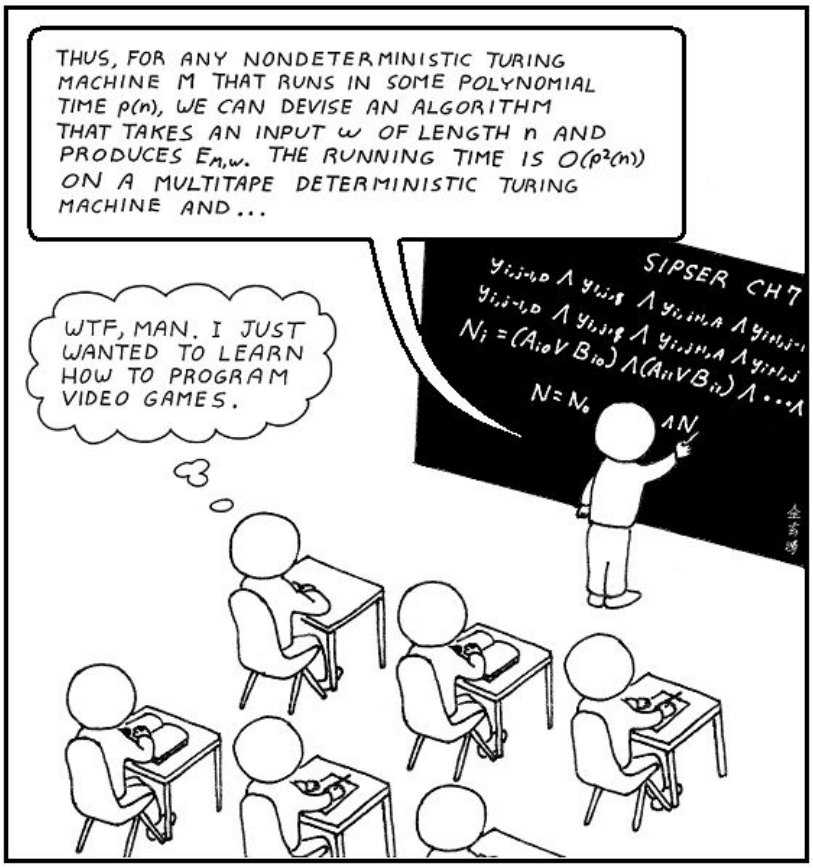
Submission Instructions

Submit to the Programming Assignment 1 folder in Gradescope.
Note that there is no autograding for assn1 - you will be notified when it has been graded manually.

Rubric

- There is nothing for you to actually get wrong in this assignment (we give you the code!) - so the only way to mess up and lose points would be to fail to follow instructions!
  So get into the habit of reading the instructions fully & carefully :)
- This assignment has to be graded manually (i.e. we have to look at your code formatting and your register table), so your results file (results.html) will be pushed to your repo only once, after the deadline - so get it right the first time. See the point above!
- For this and all assignments, use of the template given to you in the GitHub repo is **required**.

Comics??!Sweet!!!



Source: https://xkcd.com/