# CS270 Recitation 7
# "LC-3 Programming Practice"

**Goals**

Improve your understanding of LC-3 assembly programming and to learn a few advanced techniques.

**The Recitation:**

Make a subdirectory called R7 for the recitation and copy the following file into the directory:
http://www.cs.colostate.edu/~cs270/.Fall13/recitations/R7/leftshift.asm

```
;; File:        leftshift.asm
;; Description: ... fill this in
;; Author:      ... fill this in
;; Date:        ... fill this in


               .ORIG     x3000
               BR MAIN

;; Input variables
TOSHIFT   .BLKW         1              ; Input word to left shift
A         .BLKW         1              ; Input variable A, A >= B
B         .BLKW         1              ; Input variable B, B <= A
LS_AMOUNT .BLKW         1              ; Input to subroutine LEFT_SHIFT
;
;; Output variables
SHIFTED   .BLKW         1              ; Result after shifiting left
R_DIFF    .BLKW         1              ; Result computed by COMPUTE_DIFF

;; Function:     main
;; ... fill in description

MAIN           AND R1,R1,#0              ; R1 = 0
               JSR COMPUTE_DIFF          ; Call subroutine COMPUTE_DIFF
               ; ... load value of R_DIFF and store in LS_AMOUNT
               JSR LEFT_SHIFT            ; Call subroutine LEFT_SHIFT
               HALT                      ; Program complete
;
;; Subroutine: COMPUTE_DIFF
;; Input: A, B
COMPUTE_DIFF  ;.... complete rest
              RET                        ; return from subroutine
;
;; Subroutine: LEFT_SHIFT
;; Input:      TOSHIFT, LS_AMOUNT
LEFT_SHIFT    ;.... complete rest
              RET                        ; return from subroutine
;
             .END
```

Given an input word **TOSHIFT**, and two other variables **A** and **B**, the assembly program leftshift.asm attempts to left shift the value in TOSHIFT by the amount A – B and stores the result in **SHIFTED**. However, the assembly program is incomplete, and you will need to implement the subroutines describe below and add a few instructions in MAIN to complete the program.

**Subroutine-1**:
Name: **COMPUTE_DIFF**
Input: **A**, **B**
Output: **R_DIFF**
This subroutine takes two LC-3 2's complement numbers **A** and **B**, computes the difference of these numbers, and stores the result in an output variable called **R_DIFF**. You may assume that A is always greater than or equal to B.

**Subroutine-2:**
Name: LEFT_SHIFT
Input: TOSHIFT, LS_AMOUNT
Output: SHIFTED
This subroutine takes an input variable **TOSHIFT** and another variable **LS_AMOUNT** and left shifts the value in TOSHIFT by the amount LS_AMOUNT. Note that left-shifting a number truncates most significant bits from TOSHIFT. The left shifted number must be stored in variable **SHIFTED**.

**Compiling:**

~cs270/lc3tools/lc3as leftshift.asm

**Testing leftshift.asm**:

The lc3tool chain comes with a text-based simulator called lc3sim. To test your program, assemble the leftshift.asm program, and copy the following series of commands to a file called "cmds.txt":

```
file leftshift.obj
memory TOSHIFT 0x0001
memory A 0x0005
memory B 0x0003
continue
translate R_DIFF
translate SHIFTED
quit
```

Now, run lc3sim with the following command (assuming you have lc3tools inside R9):

```
$ ~cs270/lc3tools/lc3sim -s cmds.txt
```

The above command should output the value of memory location **SHIFTED**.