# CS 252 Introduction to Computer Engineering, Fall '18
# Homework 6

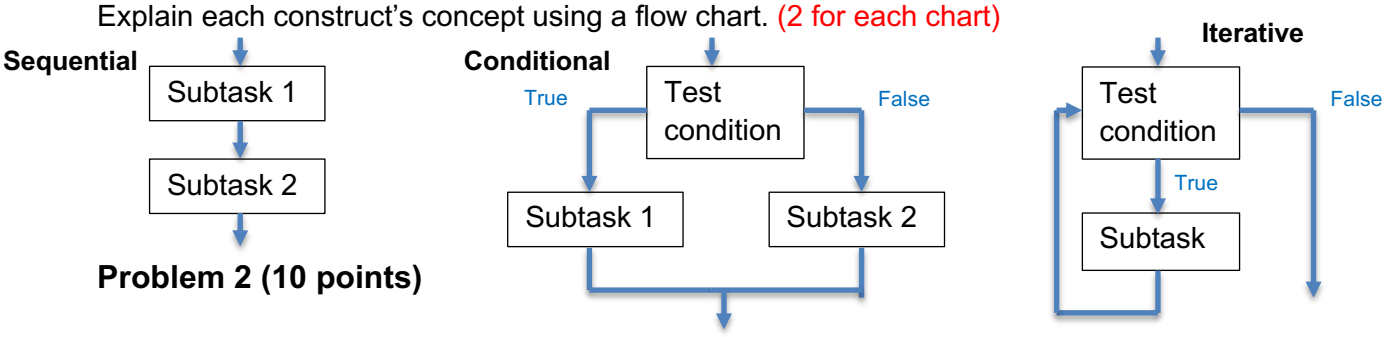
**Student Details:**


Student Name: Mo Felonda
Section: 005
Net ID: felonda@wisc.edu


**Course Details:**
Sections 2 and 5
Instructor: **Prof. Adil Ibrahim**
Primary Contact for this homework: **Yutian Tao (ytao37@wisc.edu)**
Total points: **32**

## Problem 1 (6 points)

There are three basic constructs to decompose a task: sequential, conditional, and iterative. Explain each construct's concept using a flow chart. (2 for each chart)




## Problem 2 (10 points)

The following LC-3 program increments each of the numbers stored in memory location A through memory location B. Assume these locations have already been initialized with meaningful numbers. Then the program can always finish execution. The addresses A and B can be found in memory locations x30ff and x3100.

a. Fill in the missing instructions of the code (6: 0.5 for each row)

| | | |
|---|---|---|
| x3000 | 0010 0000 1111 1110 | R0 <- M[x30ff] |
| x3001 | 0010 0010 1111 1110 | R1 <- M[x3100] |
| x3002 | 0001 0010 0110 0001 | R1 <- R1 + 1 |
| x3003 | 1001 0010 0111 1111 | R1 <- NOT R1 |
| x3004 | 0001 0010 0110 0001 | R1 <- R1 + 1 |
| x3005 | 0001 0110 0000 0001 | R3 <- R0 + R1 |
| x3006 | 0000 0100 0000 0101 | BRz PC + x005 (i.e. If Z, go to x300c) |
| x3007 | 0110 0100 0000 0000 | R2 <- M[R0] (hint: using LDR) |
| x3008 | 0001 0100 1010 0001 | R2 <- R2+ 1 |
| x3009 | 0111 0100 0000 0000 | M[R0] <- R2 (hint: using STR) |
| x300a | 0001 0000 0010 0001 | R0 <- R0+ 1 |
| x300b | 0000 1111 1111 1001 | BRnzp PC - x007 (i.e. If N/Z/P, go to x3005) |
| x300c | 1111 0000 0010 0101 | HALT |

**Student Details:**


Student Name: Mo Felonda
Section: 005
Net ID: felonda@wisc.edu


b. After the above program finishes execution,
(i) what's the value in R3? (1)

R3 = 0

(ii) Will R0 save the value as the address B? (1)

No, the address is saved after B.

c. If the instruction at x3002 is removed, will the program still get correct results? If not, what's wrong? (2)

No, the program will not get correct results, as it does not increment the value in address B.

## Problem 3 (4 points)

The tables below show the contents of memory and registers before and after an LC-3 instruction at location x3001 is executed. Identify the instruction located at x3001 and give its comment given the information below.

| | Before | After |
|---|---|---|
| R0 | x2100 | x2100 |
| R1 | x2279 | x2279 |
| R2 | x4A41 | x4A41 |
| R3 | x1532 | x1532 |
| R4 | xEFFF | xEFFF |
| R5 | x0244 | x0D12 |
| R6 | x350A | x350A |
| R7 | x533C | x533C |
| x3500 | x5671 | x5671 |
| x3501 | x0D12 | x0D12 |
| x3502 | x1743 | x1743 |
| x3503 | x53A3 | x53A3 |

| LC-3 Instruction (2) | Comment (2) |
|---|---|
| 0110 1011 1011 0111 | R5 <- M[R6 – 9] |

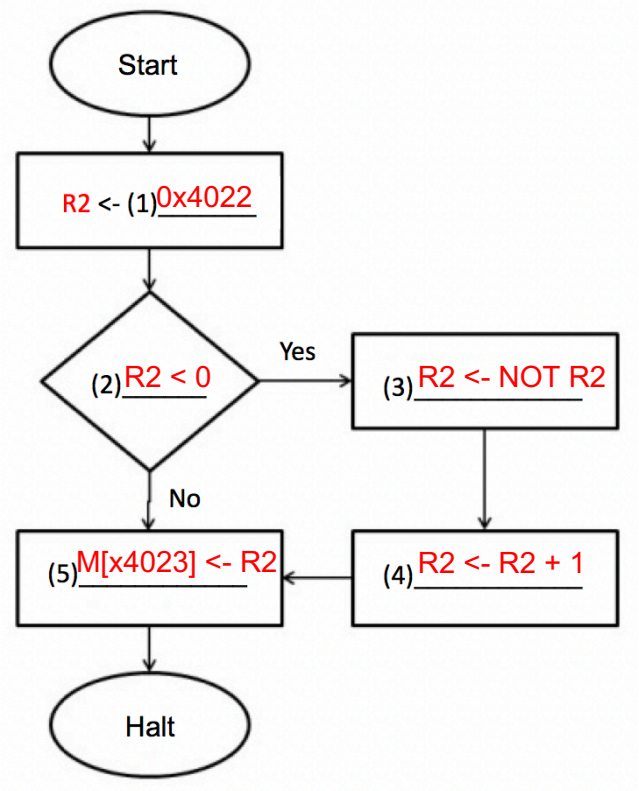
**Student Details:**


Student Name: Mo Felonda
Section: 005
Net ID: felonda@wisc.edu


## Problem 4 (12 points)

Consider an algorithm which takes the absolute value of a 2's complement number stored in memory location 0x4022 and writes the result to the memory location 0x4023. (Note: The example, the absolute value of -5 is 5, and the absolute value of 15 is 15 itself.)

a. We can represent the algorithm as the flowchart below by decomposing it into its basic constructs. Fill in the missing instructions for each block. (5)




b. Convert the above algorithm to an LC-3 program. Write the program in LC-3 binary code with comment. The program should start at memory address x3000. (Hint: use LDI and STI to access memory locations and you can store 0x4022 and 0x4023 in memory 0x3006 and 0x3007 to use) (7)

```
0x3000    ; 1010 0100 0000 0101    ; R2 <- M[M[x3006]]
0x3001    ; 0000 0110 0000 0010    ; BRzp PC + 2 (if Z,P goto 0x3004)
0x3002    ; 1001 0100 1011 1111    ; R2 <- NOT R2
0x3003    ; 0001 0100 1010 0001    ; R2 <- R2 + 1
0x3004    ; 1011 0100 0000 0010    ; M[M[x3007]] <- R2
0x3005    ; 1111 0000 0010 0101    ; HALT
```