

Name: Meng Yit Koh

Class: F1703701

ID: 517030990022

PRIOR THOUGHTS

Solve the problem by few blocks, each block is doing separate function.

Separate instruction by player1 and player2

Use a register to keep track of which player's turn

Use some constant values to change ASCII to integer to know how many rocks to print

IMPORTANT REGISTERS' VALUES

R6 = Store player's turn

R1 = Store number of stones in row A

R2 = Store number of stones in row B

R3 = Store number of stones in row C

TO USE

Run the program using LC3 simulator. Follow the instruction on the console. You **do not** need to press Enter to allow the computer to capture your decision. The program captures your decision every 2 characters typed by players(ie: A3). The one who gets all the stones finish loses.

ALGORITHM

I have described the algorithm in my program. Please take a look at **nim.asm** file.

In short, I have used many jumps to complete this program. Each turn, I will store the number of stones in each row in R1, R2, R3 respectively. And if there is no problem, the program will restart from the beginning and print the required stones as the values stored in R1, R2, R3.

If there is a problem, prompt user to input new value. Every turn, I will check if total addition of R1, R2, R3 is 0. If $(R1 + R2 + R3 == 0)$: One of the players(described in R6) wins because there is no more stone.