ICS Lab Report #3

StuID: **Name:**

Problem Setting

Here we are using the LC3 assembly language to implement a *deque*, means that we could push and pop in whether direction of a queue, so it can also have the functionality of a stack.

Algorithm Specification

Here we present a pseudocode representation of left_push, left_pop, right_push, right_pop operations.

```
start <- allocated memory location + 100
end <- start
left_push(x)
  start <- start - 1
  mem[start] <- x
left_pop()
  v_ret <- mem[start]
  start <- start + 1
right_push(x)
  mem[end] <- x
  end <- end + 1
right_pop()
  end <- end - 1
  v_ret <- mem[end]
```

LC3 Implementation

For convenience, we use a output cache as R1 points to. as an example, left_push and right_pop operations are implemented as follows.

```
SW_LPUSH
LD R7, START
```

```
TRAP x20
TRAP x21 ;echo
ADD R7, R7, #-1
STR R0, R7, #0
ST R7, START
BRNZP LOOP ;move start then store

SW_RPOP
LD R7, START
NOT R6, R7
ADD R6, R6, #1
LD R7, END
ADD R6, R6, R7
BRZ POP_FAIL ;check if empty
ADD R7, R7, #-1
LDR R0, R7, #0
STR R0, R1, #0 ;write to output cache
ADD R1, R1, #1
ST R7, END ;renewal end
BRNZP LOOP

POP_FAIL
LD R0, CHARFAIL
STR R0, R1, #0 ;store empty responce
ADD R1, R1, #1
BRNZP LOOP
```

Check Problem

Q: If continuing pushing on left/right direction, to which point will the program not work properly?

A: As my designated start is at x3200, to the left, the pushed data should not cover the last code(at x3049), so at most 151 pushes; to the right, the pushed data should not cover the output cache(at x3400), so at most 200 pushes.