

Khoa Nguyen

CS 256 - Lab 6 Draft & Design Paper

1+2+3) Stack myDataStructures

class Stack:

push(), top(), pop()

class Queue:

enqueue(), dequeue(), front() (~top)

class Dequeue:

Extra credit

4) a) open the file and read lines, then

for loop:

insertEfficient or Append

or Push

b) Create two lists ↘ original

reverse version

→ Compare

c) Create a reverse list of items.

Then populate the stack using the newly-created list.

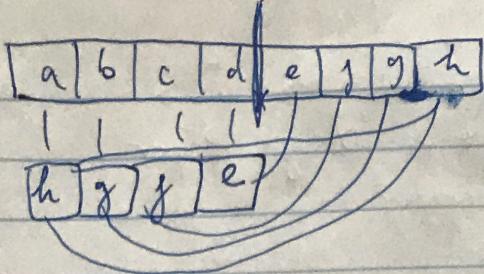
d) Pop until for loop:

Pop to count & sum until is-empty is True

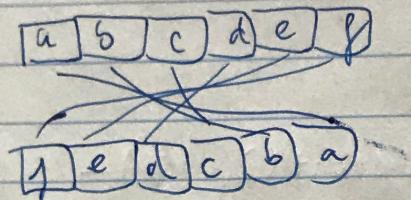
Return Count

Sum

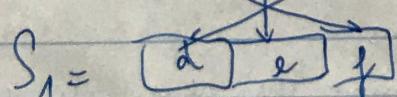
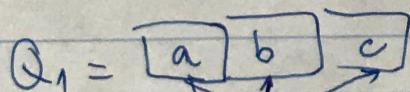
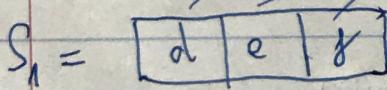
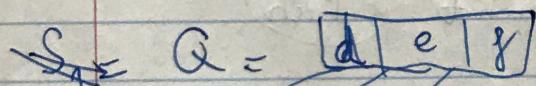
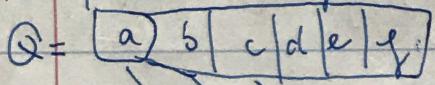
is - palindrome for Stack :



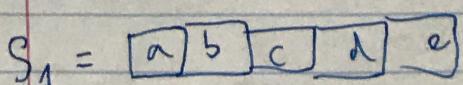
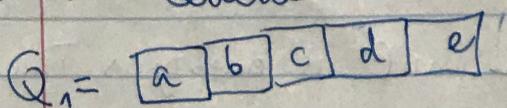
Reverse - Stack



is - palindrome for Queue



Reverse Queue:



Reverse - Q = e d c b a

* Q ~ Queue

S ~ Stack