

Problem Domain

Implement a queue using two stacks.

input: [10, 15, 20] } enqueue
output: [5, 10, 15, 20]
input: [5, 10, 15, 20] } dequeue
output: [20]

Big O enqueue

space: $O(1)$
time: $O(1)$

Big O dequeue

space $O(n)$
time $O(1)$

Algorithm

Instantiate 2 stack objects.
Stack one used for enqueue.
Stack two used for dequeue.

The enqueue method will add a value to the rear of the queue. (stack one)

The dequeue method will pop each value from the stack one, pushing each value onto stack two until the front of queue (stack one) is found.

Iterate through input string list, enqueueing each value onto stack one. When dequeue is invoked, pop each value off stack one and push each onto stack two until front of stack one is found. That front value is popped

Code

```

from stacks_and_queues import Stack
from linked_list import LinkedList

class PseudoQueue():
    def __init__(self):
        self._in_stack = Stack()
        self._out_stack = Stack()
        self.front = self._in_stack.top

    def enqueue(self, val):
        self._in_stack.push(val)
        self.front = self._in_stack.top

    def dequeue(self):
        if self._in_stack:
            self._in_stack.pop()
            self.front = self._in_stack.top

```