# Challenge 3: Reversing First k Elements of Queue

Can you reverse first "k" elements in a given queue? A solution is placed in the "solution" section for your help, but we would suggest you solve it on your own first.

**We'll cover the following**



## Problem Statement#

Implement the function `reverseK(queue, k)` which takes a queue and a number “k” as input and reverses the first “k” elements of the queue. An illustration is also provided for your understanding.

### Output#

The queue with first “k” elements reversed. Remember to return the queue itself!

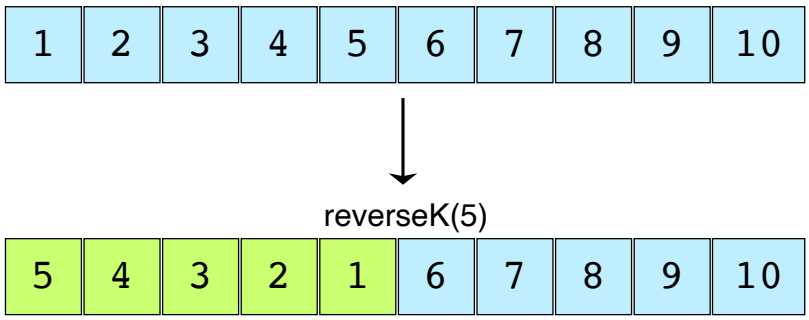
> In case the value of “k” is larger than the size of the queue, a than `0`, or if the queue is empty, simply return `None` instead.

## Sample Input#

```
Queue = [1,2,3,4,5,6,7,8,9,10], k = 5
```

## Sample Output#

```
Queue = [5,4,3,2,1,6,7,8,9,10]
```




# Coding Exercise #

Take a close look and design a step-by-step algorithm first before jumping onto the implementation. This problem is designed for your practice, so try to solve it on your own first. If you get stuck, you can always refer to the solution provided in the solution section. Good Luck!

main.py

DoublyLinkedList.py

Stack.py

Queue.py

```
from Queue import MyQueue
from Stack import MyStack


def reverseK(queue, k):
    # Write your code here
    if queue.is_empty() or k > queue.size() or k < 0:
        return None
    my_stack = MyStack()

    for i in range(k):
        my_stack.push(queue.dequeue())

    for i in range(k):
        queue.enqueue(my_stack.pop())

    for i in range(k):
        queue.enqueue(queue.dequeue())
    return queue
```



Back

Solution Review: Implementing Two S...

Next

Solution Review: Reversing First k Ele...

Completed

Report an Issue