CS 201 Data Structures II – Spring 2024

Instructor: Dr. Muhammad Mobeen Movania

Quiz 1 - Solution

Name: _____ **Date:** _____

Regn. No. : _____

There are two questions in this quiz. Each question carries 5 marks.

Q1) Show how you may implement the deque using

a) the list interface? (2.5 marks)

```
addFirst(x) -> add(0,x)
removeFirst() -> remove(0)
addLast() -> add(size(), x)
removeLast() -> remove(size()-1)
```

b) using 2 stacks? (2.5 marks)

```
class DequeUsingTwoStacks:
    def __init__(self):
        self.front_stack = []
        self.rear_stack = []

    def addFirst(self, item):
        self.front_stack.append(item)

    def removeFirst(self):
        if not self.front_stack:
            # If front stack is empty, transfer elements from rear stack
            while self.rear_stack:
                self.front_stack.append(self.rear_stack.pop())
        if not self.front_stack:
            # If both stacks are empty, deque is empty
            return None
        return self.front_stack.pop()

    def addLast(self, item):
        self.rear_stack.append(item)

    def removeLast(self):
        if not self.rear_stack:
            # If rear stack is empty, transfer elements from front stack
            while self.front_stack:
                self.rear_stack.append(self.front_stack.pop())
        if not self.rear_stack:
```

```
            # If both stacks are empty, deque is empty
            return None
        return self.rear_stack.pop()

    def is_empty(self):
        return not self.front_stack and not self.rear_stack

    def size(self):
        return len(self.front_stack) + len(self.rear_stack)
```

Q2) a) Consider an ArrayQueue with a backing array of length 5. The current state of the ArrayQueue is as follows: (4 marks)

j=1, n=3

	A	P	P	

Show the state of the ArrayQueue after each of the following operations? The values of j and n as well as the backing array should be given after each function call. Do resize the backing array if there is a need and show the changed size in your output?

Operation	**State of ArrayQueue**
remove()	| | | P | P | | j=2, n=2
add('L')	| | | P | P | L | j=2,n=3
add('E')	| E | | P | P | L | j=2,n=4
add('T')	| E | T | P | P | L | j=2,n=5
add('S')	| E | T | P | P | L | | P | P | L | E | T | S | | | | | j=0, n=6
remove()	| | P | L | E | T | S | | | | | j=1,n=5
remove()	| | | L | E | T | S | | | | | j=2,n=4
remove()	| | | | E | T | S | | | | | j=3,n=3 | E | T | S | | | | j=0,n=3

b) Show how you can implement the intersection function in USet interface? (1 mark)

```
def intersection(self, other_set):
    result_set = USet()
    for element in self.elements:
        if other_set.find(element) != null:
            result_set.add(element)
    return result_set
```

-----X-----