CSE220 Midterm Practice Sheet

1. Given a **circular array** of integers, do the following operations stepwise:

Value	25	a + 15	52	25	0	0	b + 25	25	5	19	5 + a	5	6 + b
Index	0	1	2	3	4	5	6	7	8	9	10	11	12

Where **"a"** is the **last two digits of your BRACU student id** and **"b"** is **first two digits of your BRACU student id**

The **start** of the array is **index 6**. [Note: if the capacity of the array is full, resize it by its previous capacity + 3].

1. Remove 25 by right-shifting.
2. Insert b%67 at position 5.
3. Insert studentId%13 at position 8.
4. Insert birthyear%61 at position 3.
5. Remove 5 by left-shifting.
6. Remove 52 by left-shifting.
7. Right rotate the array 3 times.
8. Left rotate the array by 4 times.

You have to show the simulation of each operation separately, no coding is required.

2. Draw a **singly linked list** where each node contains a letter of your full name without considering any space. **[You must show proper locations and indexes.]**

Do the following operations (stepwise):

a. Reverse the list (**in-place** – draw all the steps)
b. Insert 'P' in the first position
c. Insert 'A' in position 2
d. Left rotate the list 4 times
e. Delete the second element of the list
f. Insert 'G' in the last position
g. Right rotate the list 3 times
h. Sort the list in alphabetical order (show all the steps)

used string+ extra link list

3. You are given two **dummy-headed singly linked** lists, and write a method to sum the integers represented in two different lists. The input lists will have single digits in each node. The digits in each node concat to form an integer.

Sample Input	Sample Output
List 1 : x → 4 → 5 → 3 List 2 : x → 9 → 5 → 2	x → 1 → 4 → 0 → 5

Explanation: List 1 represents the integer 453. List 2 represents the integer 952. 453+952=1405. Hence, List 3 contains 1 → 4 → 0 → 5.

used array+ new linked list

4. Given **two dummy-headed sorted singly linked lists**, merge the list in one dummy-headed sorted singly linked list.

Sample Input	Sample Output
X -> 1 -> 5 -> 7 -> 9 and X -> 2 -> 3 -> 5 -> 9	X -> 1 -> 2 -> 3 -> 5 -> 5 -> 7 -> 9 -> 9
X -> 1 -> 1 -> 2 -> 9 and X -> 2 -> 4 -> 5 -> 7	X -> 1 -> 1 -> 2 -> 2 -> 4 -> 5 -> 7 -> 9

```
/*
* @param head1 is the reference of first dummy-headed singly  linked list
* @param head2 is the reference of second dummy-headed singly linked list
* @returns the head of the new list
*/

public Node mergeList (Node head1, Node head2){

        //to do

}

def mergeList (head1, head2):
        pass
```

5. Write down the insertBefore method which inserts a new element in the list before the node containing the given element.The method takes as parameters a **dummy headed doubly linked circular list**, the element existing in the list and new element to be added.

```
public void insertBefore (Node head, Object elem, Object newElement){
    //to do
}
```

OR

```
def insertBefore (head, elem, newElement):
pass
```

insertBefore (head, 3, 50)

Sample Input	Sample Output
↻ x ⇄1 ⇄ 2 ⇄ 3 ⇄ 4 ↺	↻ x ⇄1 ⇄ 2 ⇄ 50 ⇄ 3 ⇄ 4 ↺

6. Evaluate the following expression using Stack:

(2 + 4) * (9 – 8 + 5 * 2) / 2 % 5

7. Given the expressions, do the following conversions using Stack:

a. Infix to Postfix

3 + [5 / 7 - { 5 % (1 + 3 * 1) – 0 } + 1] - 1

b. Postfix to Infix

2 5 9 % + 2 - 6 2 - + 2 \ 3 6 * 8 - 1+ +

[You have to show all steps while using Stack]