Question 1

a) Consider the following Circular Queue and draw the queue frames after executing each statement given below.

		P	Q	
4	3	2	1	0

i) insert('R');

	R	P	Q	
4	3	2	1	0

ii) peekFront();

Q

iii) insert('S');

S	R	P	Q	
4	3	2	1	0

iv) insert('T');

S	R	P	Q	T
4	3	2	1	0

v) remove();

S	R	P		T
4	3	2	1	0

b) What will happen if the above queue is a linear queue?

Inset T would have ended up in an error.

\

Question 2

i) Assume that a queue class has already been implemented to store double values. Write an application to insert 5 numbers from the keyboard to a queue object created from the class.

```
 class queueApp
{
    public static void main(String []args) {
        QueueX q = new QueueX(5);

        for (int i = 1; i <= 5; i++) {
            double val = 0;
            Scanner sc = new Scanner(System.in);

            System.out.println("Enter value");
            val = sc.nextDouble();
            q.insert(val);
        }
    }
}
```

ii) Modify the application to retrieve values from the queue and print them.

```
class queueApp
{
    public static void main(String []args) {
        QueueX q = new QueueX(5);

        for (int i = 1; i <= 5; i++) {
            double val = 0;
            Scanner sc = new Scanner(System.in);

            System.out.println("Enter value");
            val = sc.nextDouble();
            q.insert(val);
        }
        while (!q.isEmpty()) {
            System.out.println(q.remove());
        }
    }
}
```

iii) Comment on the order of insertion to the queue and the order of retrieval from the queue.

First in first out order.

Question 3

Write a program to reverse the first k elements of a given circular queue. Assume the queue class is available with insert(), remove() , peek() and size() (how many locations are full) methods. Hint: Use a stack

// anytime you hear the word reverse the stack should come to mind

```
class queueApp
{
    public static void main(String []args) {
        StackX s = new StackX(k);

        for (int i = 0; i < k; i++) {
            int val = 0;
            val = q.remove();
            s.push(val);
        }

        for (int i = 0; i < k; i++) {
            int val = 0;
            val = s.pop();
            q.insert(val);
        }
        //or you can do it until the stack is empty
        while (!S.isEmpty()) {
            int val = 0;
            val = s.pop();
            q.insert(val);
        }

        //then inorder to move the front and rear pointers back to the
original position

        for (int i = 0; i < q.size() - k; i++) {
            q.insert(q.remove());
        }
    }
}
```

Lets look at this using an example

10	20	30	40	50	60	70	80	90	100
0	1	2	3	4	5	6	7	8	9
Front									Rear

And lets say we want to change the first 4 elements

```
StackX s = new StackX(4);

        for (int i = 0; i < 4; i++) {
            int val = 0;
            val = q.remove();
            s.push(val);
        }
```

				50	60	70	80	90	100
0	1	2	3	4	5	6	7	8	9
				Front					Rear

40
30
20
10

```
while (!S.isEmpty()) {
            int val = 0;
            val = s.pop();
            q.insert(val);
        }
```

40	30	20	10	50	60	70	80	90	100
0	1	2	3	4	5	6	7	8	9
			Rear	Front					

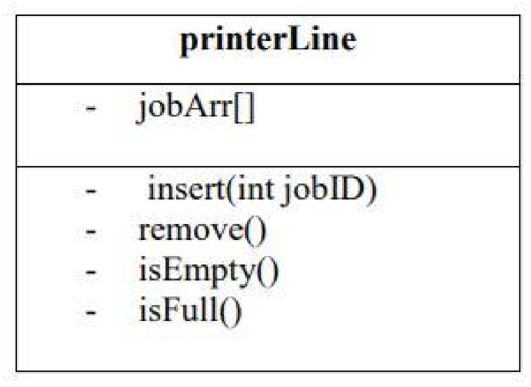
```
        for (int i = 0; i < q.size() - k; i++) {
            q.insert(q.remove());
        }
```

40	30	20	10	50	60	70	80	90	100
0	1	2	3	4	5	6	7	8	9
Front									Rear

Additional Exercises:

Question 1

i) You are required to build the following class in your program.



A printer processes the jobs that are sent to be printed in the same order as it receives. You are required to create an application which will do the same function that a printer does.(printer queue)
Hint : In addition to the attributes and methods given above you can use the necessary attributes and methods that are related with the data structure you have selected

a) Implement the constructor printerLine(int size);

b) Implement insert(), remove(), isEmpty() and isFull() methods of the class.

c) Write a main Program to create an object with 5 elements of the printerLine class.

d) Allow the user to input 5 JobIDs from the keyboard.

```
Enter JobID : 101
Enter JobID : 22
Enter JobID : 180
Enter JobID : 111
Enter JobID : 50
```

e) You are required to send JobIDs to separate PCs: Jobs sent to PC1 are even numbers and jobs sent to PC2 are odd numbers.

(Eg: JobID 22 is send to PC1 and JobID 111 is send to PC2)

f) Write the code to remove the jobs and display the result as follows.

```
JobID 22 (PC1)
```

JobID 111 (PC2)

Answer is in the folder in the hard drive in the sliit folder