**1\**

**OUT PUT:**

1. **5**
2. **9**
3. **16**
4. **4**
5. **2**

**2\**

**OUT PUT:**

1. **10**
2. **20**
3. **40**
4. **20**
5. **5**
6. **3**
7. **NULL**
8. **NULL**

**3\**

**OUT PUT:**

0

14

8

Stack Element =

8

14

22

64

Queue Element =

14

35

19

Queue Element =

22

32

7

Queue Element =

64

11

13

Stack Element =

13

30

-999

Stack Elements:

30

13

14

8

Queue Elements:

11

32

7

**4\**

1. queueFront**=50 ,** queueRear**=0**
2. queueFront**=50 ,** queueRear**=0**

**after** queueFront**=51 ,** queueRear**=0**

**5\**

1. queueFront**=99 ,** queueRear**=25**

**after** queueFront**=99 ,** queueRear**=26**

1. queueFront**=99 ,** queueRear**=26**

**after** queueFront**=0 ,** queueRear**=26**

**6\**

1. queueFront**=25 ,** queueRear**=75**

**after** queueFront**=25 ,** queueRear**=76**

1. queueFront**=25 ,** queueRear**=76**

**after** queueFront**=26 ,** queueRear**=76**

**7\**

1. queueFront**=99 ,** queueRear**=99**

**after** queueFront**=99 ,** queueRear**=0**

1. queueFront**=99 ,** queueRear**=0**

**after** queueFront**=0 ,** queueRear**=0**

**8\**

**import java.util.Queue;**

**import java.util.Stack;**

**public class QueueReversal {**

**public static <T> void reverseQueue(Queue<T> queue) {**

**Stack<T> stack = new Stack<>();**

**// Push all elements from the queue to the stack**

**while (!queue.isEmpty()) {**

**stack.push(queue.poll());**

**}**

**// Pop elements from the stack and enqueue them back to the queue**

**while (!stack.isEmpty()) {**

**queue.offer(stack.pop());**

**}**

**}**

**}**

**9\**

**Curent size of q= 32-(15-5)=32-10=22**

**10\**

**1. addFirst(3): الdeque يصبح [3]. لا يتم إرجاع أي قيمة.**

**2. addLast(8): الdeque يصبح [3, 8]. لا يتم إرجاع أي قيمة.**

**3. addLast(9): الdeque يصبح [3, 8, 9]. لا يتم إرجاع أي قيمة.**

**4. addFirst(1): الdeque يصبح [1, 3, 8, 9]. لا يتم إرجاع أي قيمة.**

**5. last(): يعيد 9 (آخر عنصر في الdeque).**

**6. isEmpty(): يعيد false (نظرًا لأن الdeque ليست فارغة).**

**7. addFirst(2): الdeque يصبح [2, 1, 3, 8, 9]. لا يتم إرجاع أي قيمة.**

**8. removeLast(): يزيل ويعيد 9. الdeque يصبح [2, 1, 3, 8].**

**9. addLast(7): الdeque يصبح [2, 1, 3, 8, 7]. لا يتم إرجاع أي قيمة.**

**10. first(): يعيد 2 (أول عنصر في الdeque).**

**11. last(): يعيد 7 (آخر عنصر في الdeque).**

**12. addLast(4): الdeque يصبح [2, 1, 3, 8, 7, 4]. لا يتم إرجاع أي قيمة.**

**13. size(): يعيد 6 (عدد العناصر في الdeque).**

**14. removeFirst(): يزيل ويعيد 2. الdeque يصبح [1, 3, 8, 7, 4].**

**15. removeFirst(): يزيل ويعيد 1. الdeque يصبح [3, 8, 7, 4].**

**لذا، القيم المُرجعة خلال تسلسل عمليات deque هي كالتالي:**

**- last(): 9**

**- isEmpty(): false**

**- removeLast(): 9**

**- first(): 2**

**- last(): 7**

**- size(): 6**

**- removeFirst(): 2**

**- removeFirst(): 1**