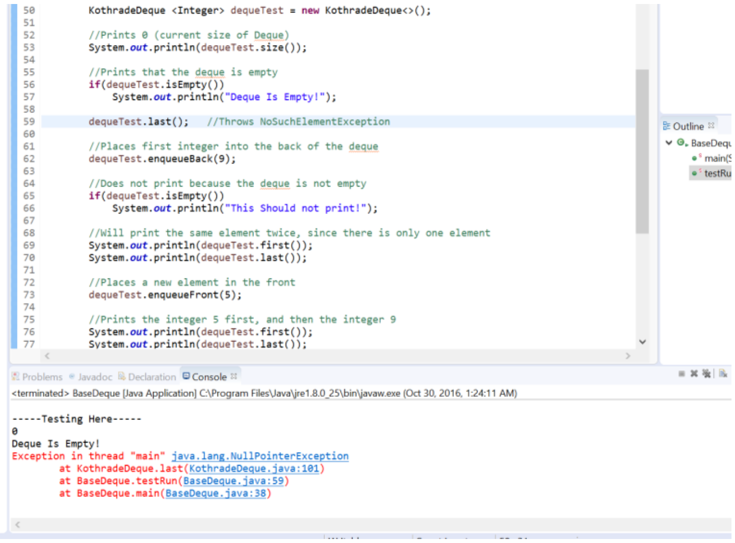
The program written is designed to create a double ended queue (deque). I have used sample input and expected outputs to test the design of the KothradeDeque class, which implements the deque interface.

To start, I call the size() method, which prints a "0" as there are no elements in the deque yet. Following this up, I use an if statement that references the Boolean isEmpty() method to print out a message declaring that the deque does not contain any elements. While the deque is still empty, I call the last() method, which throws an error, since there was no element to return. (see pic)

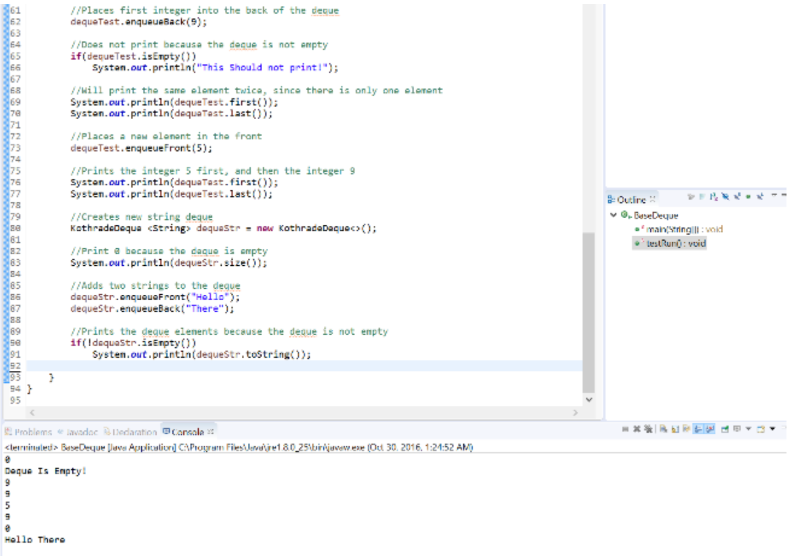


```
        KothradeDeque <Integer> dequeTest = new KothradeDeque<>();

        //Prints 0 (current size of Deque)
        System.out.println(dequeTest.size());

        //Prints that the deque is empty
        if(dequeTest.isEmpty())
            System.out.println("Deque Is Empty!");

        dequeTest.last();   //Throws NoSuchElementException

        //Places first integer into the back of the deque
        dequeTest.enqueueBack(9);

        //Does not print because the deque is not empty
        if(dequeTest.isEmpty())
            System.out.println("This Should not print!");

        //Will print the same element twice, since there is only one element
        System.out.println(dequeTest.first());
        System.out.println(dequeTest.last());

        //Places a new element in the front
        dequeTest.enqueueFront(5);

        //Prints the integer 5 first, and then the integer 9
        System.out.println(dequeTest.first());
        System.out.println(dequeTest.last());
```

```
<terminated> BaseDeque [Java Application] C:\Program Files\Java\jre1.8.0_25\bin\javaw.exe (Oct 30, 2016, 1:24:11 AM)

-----Testing Here-----
0
Deque Is Empty!
Exception in thread "main" java.lang.NullPointerException
	at KothradeDeque.last(KothradeDeque.java:101)
	at BaseDeque.testRun(BaseDeque.java:59)
	at BaseDeque.main(BaseDeque.java:38)
```

To begin filling the deque, I place the integer 9 using the enqueueBack() method. Now, using the first() and last() methods, I check to see that both my head and tail nodes are working as intended by printing the #9 twice. To test the first() and last() methods again, I add the integer 5 to the front using the enqueueFront() method, and then proceed to print out both elements.

Finally, I begin to close up my testing with the creation of a new deque which contains String objects. I print the size() method again to ensure the deque is indeed empty, and then use the enqueue methods to enter "Hello" and "There" to the deque. To finish I utitlize an if statement that prints the toString() method as long as the deque is not empty.



```
        //Places first integer into the back of the deque
        dequeTest.enqueueBack(9);

        //Does not print because the deque is not empty
        if(dequeTest.isEmpty())
            System.out.println("This Should not print!");

        //Will print the same element twice, since there is only one element
        System.out.println(dequeTest.first());
        System.out.println(dequeTest.last());

        //Places a new element in the front
        dequeTest.enqueueFront(5);

        //Prints the integer 5 first, and then the integer 9
        System.out.println(dequeTest.first());
        System.out.println(dequeTest.last());

        //Creates new string deque
        KothradeDeque <String> dequeStr = new KothradeDeque<>();

        //Print 0 because the deque is empty
        System.out.println(dequeStr.size());

        //Adds two strings to the deque
        dequeStr.enqueueFront("Hello");
        dequeStr.enqueueBack("There");

        //Prints the deque elements because the deque is not empty
        if(!dequeStr.isEmpty())
            System.out.println(dequeStr.toString());

    }
}
```

```
<terminated> BaseDeque [Java Application] C:\Program Files\Java\jre1.8.0_25\bin\javaw.exe (Oct 30, 2016, 1:24:52 AM)
0
Deque Is Empty!
9
9
5
9
0
Hello There
```