Part 1

Implement the Queue interface with a class that is based on the LinkedDeque and DLinkedList classes (wrapper). See code fragments 3.23, 5.21, 5.22, and 5.23 and table 5.4 (below). Do not use the STL. Highlight your Deque and Queue classes.

|  |  |
| --- | --- |
| Queue method | Deque Implementation |
| size() | size() |
| empty() | empty() |
| front() | front() |
| enqueue() | insertBack() |
| dequeue () | eraseFront() |

Test all the member functions (size(), empty(), front(), enqueue, and dequeue) of the Queue class with the following data.

String string1 = "A man, a plan, a canal, Panama”;

String string2 = “Was it a car or a cat I saw?";

String string3 = “Sit on a potato pan, Otis";

String string4 = “No lemon, no melon";

Part 2

Implement the vector ADT by means of an extendable array (expands dynamically) used in a circular fashion, so that insertions and deletions at the beginning and end run in constant time. Print the circular array before and after each insertion and deletion. Insertions and deletion that are not at the beginning or end will run in O(n) time. Test the vector ADT methods (inserts, deletes, size, isEmpty, etc.)

Do not use the <vector> STL.

Thoroughly test **all** the methods of your vector ADT.

Due September 20th