Q3

GIT Department of Computer Engineering CSE 222/505 - Spring 2020 Homework 4

Djuro Radusinovic

171044095

Problem solution approach

I implemented a deque interface here. Inside I made a private class that has LinkedList functionality. Also, there is another class inside called Node which is used in both iterator and linked list implementation. Iterator class and Descending Iterator classes are not static here as they access its outer class's elements. Also Descending Iterator extends Iterator class and overrides its next method and has a separate constructor for it. Everything is written to be as simple as possible. My deque keeps two linked lists of which one is elements(current elements in the list) and removed elements which keeps the nodes that have been removed and can be reused. Each time an element is removed from 'elements' list that element is added to 'removed elements' list. Every time an element is added to 'elements' list it is first checked if there is an element in 'removed elements' list that can be used to avoid constructing a new node and garbage collection when an element is removed from 'elements' list. There is a Main class in which there is main method which is used for testing the code. Output is shown in terminal of the console running the code.

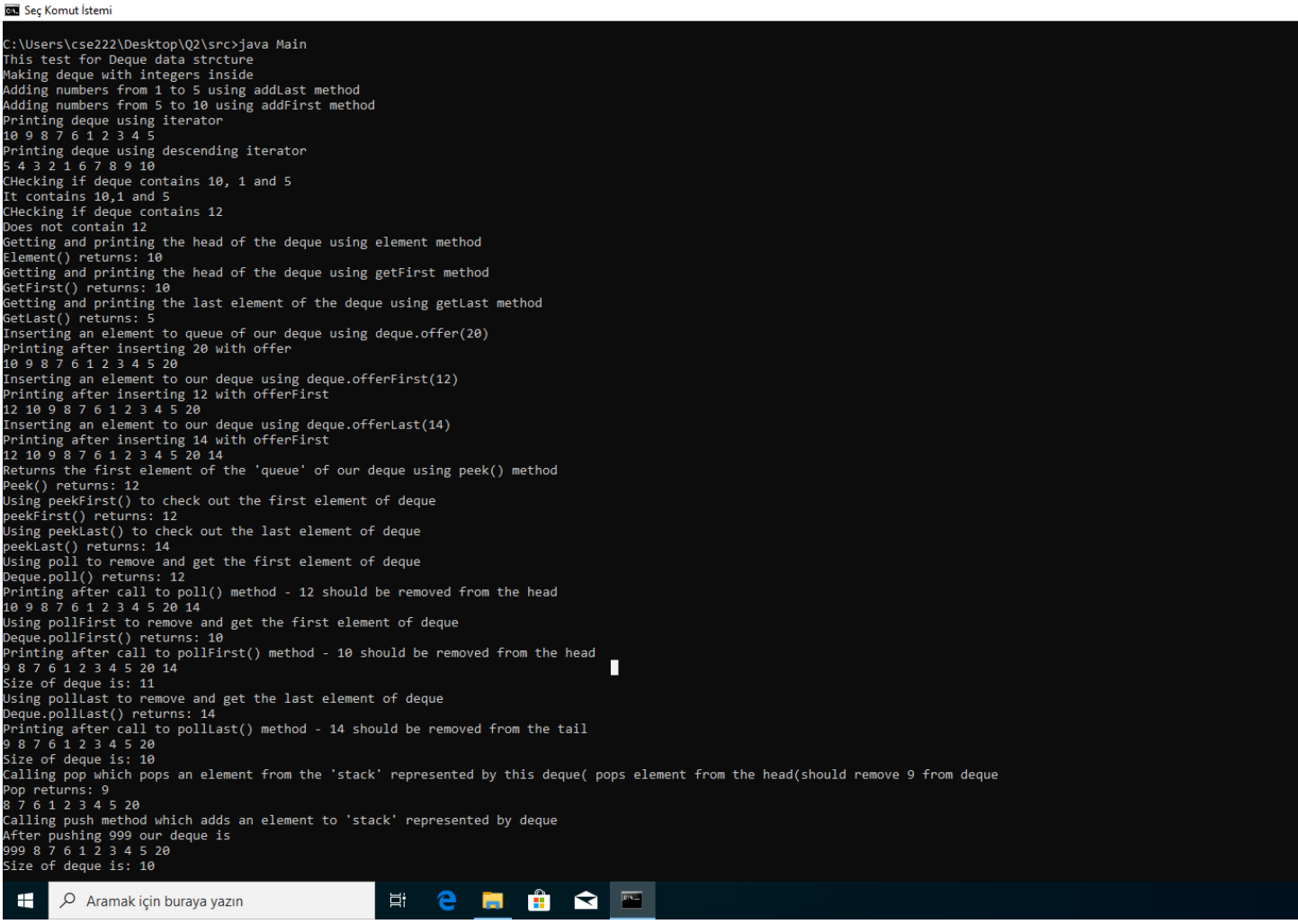
Test cases

Test Scenario	Expected Results	Actual Results
Adding numbers from 1 to 5 using addLast method	Should be added	As expected
Adding numbers from 5 to 10 using addFirst method	Should be added	As expected
Printing deque using iterator	Should be printed from head to tail	As expected
Printing deque using descending iterator	Should be printed from tail to head	As expected
Checking if deque contains 10, 1 and 5	Should return true	As expected

Checking if deque contains 12	Should return false	As expected
Getting and printing the head of the deque using element method	Should return head	As expected
Getting the head of the deque using getFirst()	Should return element at the head	As expected
Getting the tail of the deque using getLast()	Should return element at the tail	As expected
Inserting an element to queue of our deque using deque.offer(20)	Should be able to insert an elment ‘20’	As expected
Testing offerFirst with offerFirst(12)	Shold add an element ‘12’	As expected
OfferLast with offerLast(14)	Should add 14 at tail	As expected
Checking the first element in deque using peek() method	Should return the element at the head	As expected
Checking the first element in deque using peekFirst() method	Should return the element at the head	As expected
Checking the first element in deque using peekLast() method	Should return the element at the tail	As expected
Testing poll() to get and remove the first element at the same time	Should return the first element in deque and remove it	As expected
Testing pollFirst() to get and remove the first element at the same time	Should return the first element in deque and remove it	As expected
Testing pollLast() to get and remove the first element at the same time	Should return the last element in deque and remove it	As expected
Testing pop() to get and remove the first element at the same time. Should show stack functionality	Should return the first element in deque and remove it	As expected
Calling push method which adds an element	Should add ‘999’ that was provided	As expected
Calling remove() which returns head of our deque and removes it	Should remove and return 999	As expected
Removing specific element by calling remove(T t)	Should remove ‘1’ with which it was called	As expecetd

Calling removeFirst() which returns head of our deque and removes it	Should remove and return 8	As expected
Adding 2 at tail and at head	Should add element	As expected
Removing first occurrence of 2	Should remove first 2 from the left	As expected
Removing last occurrence of 2	Should remove first 2 from the right	As expected
Removing first occurrence of 2 again(now it is at the middle of our deque)	Should remove first 2 from the left	As expected
RemoveLast() to get the element at the tail	Should remove 20 from the end of deque	As expected
Check the current size of deque	Should be 5	As expected
Adding a null element to deque	Should throw an exception	As expected
Using iterator to remove each element of deque	Should empty out the deque	As expected
Adding 1,2,3 and 4	Should add to before empty deque	As expected
Removing again now using descending iterator	Should remove them all	As expected
Checking the size after the removal	Should be 0	As expected
Trying to remove an element after the list is empty(using remove())	Should throw an exception	As expected

Running command and results


Seç Komut İstemi
C:\Users\cse222\Desktop\Q2\src>java Main
This test for Deque data strcture
Making deque with integers inside
Adding numbers from 1 to 5 using addLast method
Adding numbers from 5 to 10 using addFirst method
Printing deque using iterator
10 9 8 7 6 1 2 3 4 5
Printing deque using descending iterator
5 4 3 2 1 6 7 8 9 10
CHecking if deque contains 10, 1 and 5
It contains 10,1 and 5
CHecking if deque contains 12
Does not contain 12
Getting and printing the head of the deque using element method
Element() returns: 10
Getting and printing the head of the deque using getFirst method
GetFirst() returns: 10
Getting and printing the last element of the deque using getLast method
GetLast() returns: 5
Inserting an element to queue of our deque using deque.offer(20)
Printing after inserting 20 with offer
10 9 8 7 6 1 2 3 4 5 20
Inserting an element to our deque using deque.offerFirst(12)
Printing after inserting 12 with offerFirst
12 10 9 8 7 6 1 2 3 4 5 20
Inserting an element to our deque using deque.offerLast(14)
Printing after inserting 14 with offerFirst
12 10 9 8 7 6 1 2 3 4 5 20 14
Returns the first element of the 'queue' of our deque using peek() method
Peek() returns: 12
Using peekFirst() to check out the first element of deque
peekFirst() returns: 12
Using peekLast() to check out the last element of deque
peekLast() returns: 14
Using poll to remove and get the first element of deque
Deque.poll() returns: 12
Printing after call to poll() method - 12 should be removed from the head
10 9 8 7 6 1 2 3 4 5 20 14
Using pollFirst to remove and get the first element of deque
Deque.pollFirst() returns: 10
Printing after call to pollFirst() method - 10 should be removed from the head
9 8 7 6 1 2 3 4 5 20 14
Size of deque is: 11
Using pollLast to remove and get the last element of deque
Deque.pollLast() returns: 14
Printing after call to pollLast() method - 14 should be removed from the tail
9 8 7 6 1 2 3 4 5 20
Size of deque is: 10
Calling pop which pops an element from the 'stack' represented by this deque(pops element from the head(should remove 9 from deque
Pop returns: 9
8 7 6 1 2 3 4 5 20
Calling push method which adds an element to 'stack' represented by deque
After pushing 999 our deque is
999 8 7 6 1 2 3 4 5 20
Size of deque is: 10
Aramak için buraya yazın

```
Seç Komut İstemi
Using pollFirst to remove and get the first element of deque
Deque.pollFirst() returns: 10
Printing after call to pollFirst() method - 10 should be removed from the head
9 8 7 6 1 2 3 4 5 20 14
Size of deque is: 11
Using pollLast to remove and get the last element of deque
Deque.pollLast() returns: 14
Printing after call to polllast() method - 14 should be removed from the tail
9 8 7 6 1 2 3 4 5 20
Size of deque is: 10
Calling pop which pops an element from the 'stack' represented by this deque( pops element from the head(should remove 9 from deque
Pop returns: 9
8 7 6 1 2 3 4 5 20
Calling push method which adds an element to 'stack' represented by deque
After pushing 999 our deque is
999 8 7 6 1 2 3 4 5 20
Size of deque is: 10
Calling remove() which removes head of our deque and removes it
Remove's output: 999
8 7 6 1 2 3 4 5 20
Size of deque is: 9
Removing a specific element using remove(T t) method. Calling it with '1'
After removing '1' deque is
8 7 6 2 3 4 5 20
Deque size: 8
Calling removeFirst which removes the first element of deque
removeFirst returned: 8
7 6 2 3 4 5 20
Deque size:7
Adding 2 at the end and at the tail of our deque
2 7 6 2 3 4 5 20 2
Calling removeFirstOccurence for 2
After removing first occurrence of our deque it is:
7 6 2 3 4 5 20 2
Removing the last occurrence of 2
7 6 2 3 4 5 20
Removing first occurence of it again
7 6 3 4 5 20
Size of deque: 6
RemoveLast() returns: 20
7 6 3 4 5
Size of deque now is: 5
Trying to add a null element to queue
Exception caught when trying to add null to our deque
2 7 6 3 4 5
Size of deque after removing each element using iterator: 0
Adding 1,2,3 and 4 to deque
4 3 2 1
Size of deque is: 4
size of deque after removing each element using descending iterator: 0
Trying to use remove method on an empty deque
Exception caught when trying to remove from an empty deque

C:\Users\cse222\Desktop\Q2\src>_
```

Aramak için buraya yazın

Class diagram