Tabitha Roemish

Project 2: Circular doubly-linked list

Test Plan & Output

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test | Expected | Actual | Reason for Discrepancy | Reason for test |
| CDLList construct(5, 6) | (6,6,6,6,6) | (6,6,6,6,6) | n/a | Check constructor builds list with 5 and fills with 6. |
| CDLList construct2(construct) | (6,6,6,6,6) | (6,6,6,6,6) | n/a | Check copy constructor works |
| CDLList Construct3(construct2.begin(), construct2.end()) | (6,6,6,6,6) | (6,6,6,6,6) | n/a | Check copy constructor with iterators works & iterator functions begin(), end() |
| Construct2.push\_front(5); | (5,6,6,6,6,6) | (5,6,6,6,6,6) | n/a | Check push\_front works |
| Construct2.push\_back(5); | (5,6,6,6,6,6,5) | (5,6,6,6,6,6,5) | n/a | Check push\_back works |
| Construct = construct2  Construct output - > | (5,6,6,6,6,6,5) | (5,6,6,6,6,6,5) | n/a | Check assignment operator works |
| Construct2.pop\_front() | (6,6,6,6,6,5) | (6,6,6,6,6,5) | n/a | Check pop front works |
| Construct2.pop\_back() | (6,6,6,6,6) | (6,6,6,6,6) | n/a – initial output was not the same. Testing was completed while building main. | Check that pop\_back works |
| Construct2[3] = 2 | (6,6,6,2,6) | (6,6,6,2,6) | n/a | Check [] operator overload works |
| Test List = {0,2,4,6,8}  CDLList::iterator start = begin();  Start++ | 2 | 2 | n/a | Check post incrementing iterator function works |
| Test List = {0,2,4,6,8}  CDLList::iterator start = end();  Start-- | 6 | 6 | n/a | Check post decrementing iterator works |
| Test List = {0,2,4,6,8}  CDLList::iterator start = begin();  ++Start | 0 | 0 | n/a | Check pre incrementing iterator function works |
| Test List = {0,2,4,6,8}  CDLList::iterator start = end();  --Start | 8 | 8 | n/a | Check pre decrementing iterator works |
| Queue<string> construct1(5, “Test”) | (Test, Test, Test, Test, Test) | (Test, Test, Test, Test, Test) | n/a | Check queue construct with parameters works |
| Queue<string> construct2(Construct1) | (Test, Test, Test, Test, Test) | (Test, Test, Test, Test, Test) | n/a | Check queue copy constructor that takes queue works |
| Queue<string>construct3(CDLList test) CDLList = {x,x,x,x,x} | (x, x, x, x, x, x) | (x, x, x, x, x, x) | n/a | Check copy constructor taking in CDLList works |
| Queue<string> construct4(construct3.begin(), contruct3.end()) | (x, x, x, x, x, x) | (x, x, x, x, x, x) | n/a | Check copy constructor using iterators works |
| Construct3.push(“new”) | (x, x, x, x, x, x, New) | (x, x, x, x, x, x, New) | n/a | Check push works for queue |
| Construct3.pop() | (x, x, x, x, x, New) | (x, x, x, x, x, New) | n/a | Check pop works for queue |
| construct3.begin() | x | x | n/a | Check iterators for queue work |
| contruct3.end()) | new | new | n/a | Check iterators for queue work |

Output:

