

```
1: /*Copyright [2016] <Albara Mehene> */
2: /*#define BOOST_TEST_DYN_LINK
3: #define BOOST_TEST_MODULE Main
4: */
5:
6: #define BOOST_TEST_DYN_LINK
7: #define BOOST_TEST_MODULE Main
8: #include <boost/test/unit_test.hpp>
9:
10: #include <stdint.h>
11: #include <iostream>
12: #include <string>
13: #include <exception>
14: #include <stdexcept>
15:
16: #include "RingBuffer.hpp"
17:
18: BOOST_AUTO_TEST_CASE(RBconstructor) {
19:     // normal constructor
20:     BOOST_REQUIRE_NO_THROW(RingBuffer(100));
21:
22:     // this should fail
23:     BOOST_REQUIRE_THROW(RingBuffer(0), std::exception);
24:     BOOST_REQUIRE_THROW(RingBuffer(0), std::invalid_argument);
25: }
26:
27: BOOST_AUTO_TEST_CASE(RBenque_dequeue) {
28:     RingBuffer rb(100);
29:
30:     rb.enqueue(2);
31:     rb.enqueue(1);
32:     rb.enqueue(0);
33:
34:     BOOST_REQUIRE(rb.dequeue() == 2);
35:     BOOST_REQUIRE(rb.dequeue() == 1);
36:     BOOST_REQUIRE(rb.dequeue() == 0);
37:
38:     BOOST_REQUIRE_THROW(rb.dequeue(), std::runtime_error);
39: }
40:
41: BOOST_AUTO_TEST_CASE(test1) {
42:     RingBuffer rb(3);
43:     rb.enqueue(2);
44:     rb.enqueue(3);
45:     rb.enqueue(4);
46:     BOOST_REQUIRE(rb.isFull());
47:     BOOST_REQUIRE_THROW(rb.enqueue(5), std::runtime_error)
48: }
```