

```
1: // Copyright 2015 <Angel Z'heondre Calcano>
2: #ifndef _RingBuffer_
3: #define _RingBuffer_
4: #include <stdint.h>
5: #include <iostream>
6: #include <string>
7: #include <vector>
8: #include <stdexcept>
9:
10: class RingBuffer{
11:     int _size, _first, _last, _currentcapacity;
12:     std::vector< double > _buffer;
13: public:
14:     explicit RingBuffer(int x): _size(x) {
15:         if ( x < 1 )
16:             throw std::invalid_argument("Constructor capacity must be > than 0");
17:         _first = _last = _currentcapacity = 0;
18:         for (int i = 0 ; i < x; i++)
19:             _buffer.push_back(100);
20:     }
21:     void RB();
22:     int size();
23:     bool isEmpty();
24:     bool isFull();
25:     void enqueue(int16_t x);
26:     int16_t dequeue();
27:     int16_t peek();
28: };
29: #endif
```