Lab5 exercise 1: DeckQueue

Part 1 Code Implementation:

```
public static void sort(DeckQueue deckqueue) {
    int n = deckqueue.size();
    //排序完成一共需要n-1轮
    for (int i = 1; i <= n - 1; i++) {
        //第i轮操作需要n-i次换牌，将最小的牌移到top并放到bottom位置
        for (int j = 0; j < n - i; j++) {
            if (deckqueue.second() < deckqueue.first())
                deckqueue.exchangeTopTwo();
            deckqueue.moveTop2Bottom();
        }
        //一轮结束后做i次将最上面的牌移到最下面的操作
        for (int k = 0; k < i; k++) {
            deckqueue.moveTop2Bottom();
        }
    }
}
```

Part 2 Algorithm Test

1. Large Random Numbers Test:

```
Data size: 100, time: 0.
true
Data size: 200, time: 16.
true
Data size: 400, time: 0.
true
Data size: 800, time: 15.
true
Data size: 1600, time: 63.
true
Data size: 3200, time: 179.
true
Data size: 6400, time: 779.
true
Data size: 12800, time: 4028.
true
Data size: 25600, time: 11606.
true
Data size: 51200, time: 32837.
true
```

2. Special Cases Test

(1)Elements in the array are the same:

(2)Array is already sorted:

(3)Array is reverse sorted:

(4)Contains duplicate numbers:

(5)The length of array is 0 or 1: Constructor in DeckQueue don't allow the length of array is 0 or 1, if so it will throw exceptions.

```
All zero array: true
Sorted array: true
Reverse-sorted array: true
Contains repetitive numbers array: true
Exception in thread "main" java.lang.IllegalArgumentException Create breakpoint : array size should >= 2
    at DeckQueue.<init>(DeckQueue.java:11)
    at Main.main(Main.java:63)
```