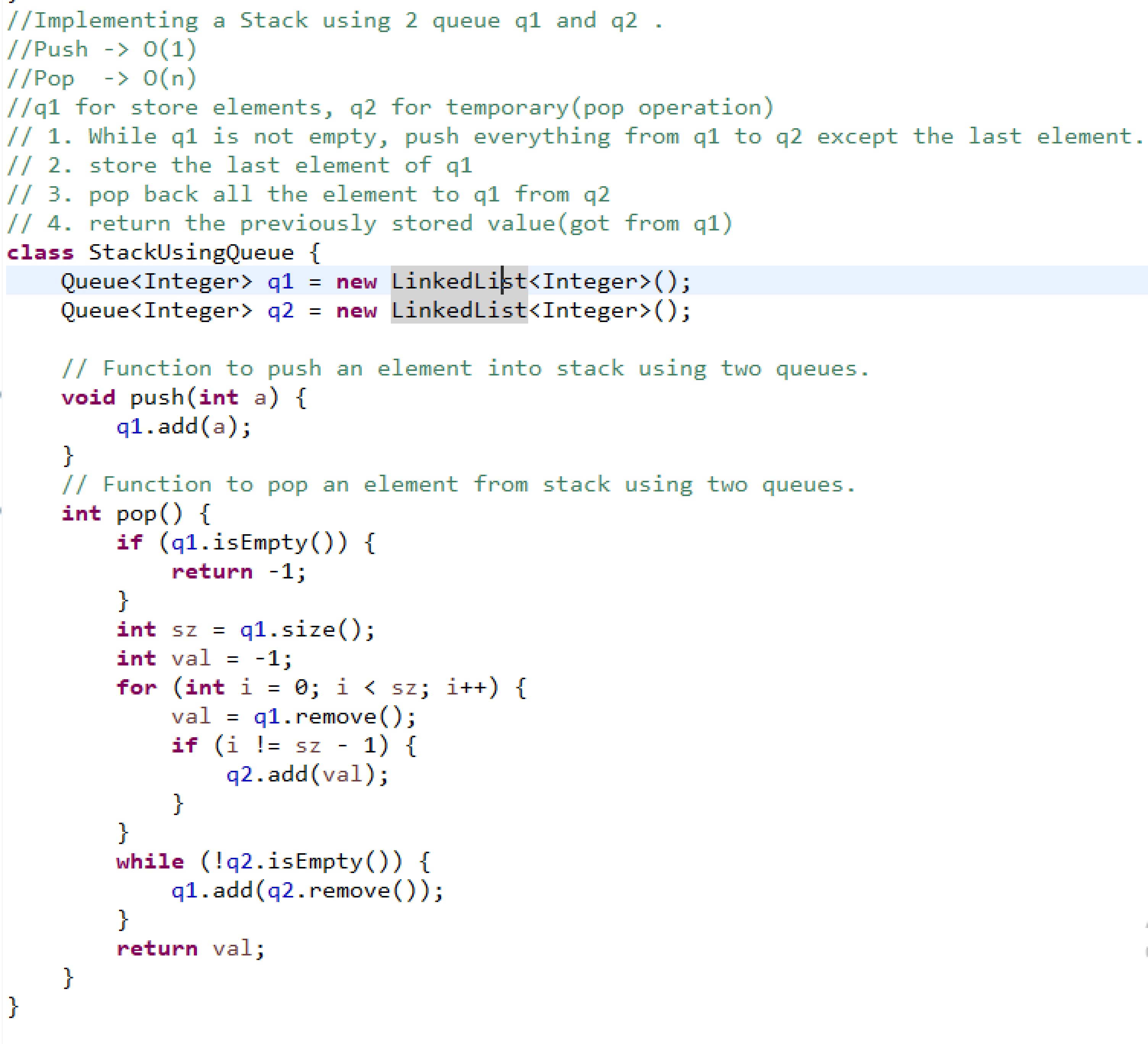

```java
}
//Implementing a Stack using 2 queue q1 and q2 .
//Push -> O(1)
//Pop  -> O(n)
//q1 for store elements, q2 for temporary(pop operation)
// 1. While q1 is not empty, push everything from q1 to q2 except the last element.
// 2. store the last element of q1
// 3. pop back all the element to q1 from q2
// 4. return the previously stored value(got from q1)
class StackUsingQueue {
    Queue<Integer> q1 = new LinkedList<Integer>();
    Queue<Integer> q2 = new LinkedList<Integer>();

    // Function to push an element into stack using two queues.
    void push(int a) {
        q1.add(a);
    }
    // Function to pop an element from stack using two queues.
    int pop() {
        if (q1.isEmpty()) {
            return -1;
        }
        int sz = q1.size();
        int val = -1;
        for (int i = 0; i < sz; i++) {
            val = q1.remove();
            if (i != sz - 1) {
                q2.add(val);
            }
        }
        while (!q2.isEmpty()) {
            q1.add(q2.remove());
        }
        return val;
    }
}
```