**Problem:** **Implement stack using queues**

1. Explanation of the problem with examples (minimum three examples).
2. Example 1

Input: **5, 4**

Output:

**4** - when calling pop() method

**5** - when calling top() method

**false** - when calling empty() method

1. Example 2

Input: **7, 8, 9**

Output:

**9** - when calling pop() method

**8** - when calling top() method

**8** – when calling pop() method

**false** - when calling empty() method

1. Example 3

Input: **2, 3**

Output:

**3** – when calling pop() method

**2** – when calling pop() method

**true** – when calling empty() method

1. Explanation of possible solution(s)

* Create queue by using the java queue collection, pop() method of stack should be called queue.remove(), top() method of stack should be called queue.peek().
* Only update the push() method and make it work as a stack data structure instead of queue.

1. Implementation and testing of your solution

* See the source code in the parent folder

1. Time and Space complexity of your algorithm

Time complexity:

push(int x): O(n)

pop(): O(1)

top(): O(1)

empty(): O(1)

Space complexity: O(n)