Solution Review: Generate Binary Numbers from 1 to n using Queue

In this lesson, we will do a solution review for the 'Generate Binary Numbers from 1 to n using Queue' challenge.

We'll cover the following

Solution: Using a Queue

main.py

DoublyLinkedList.py

Queue.py

```
from Queue import MyQueue


def find_bin(number):
    result = []
    queue = MyQueue()
    queue.enqueue(1)
    for i in range(number):
        result.append(str(queue.dequeue()))
        s1 = result[i] + "0"
        s2 = result[i] + "1"
        queue.enqueue(s1)
        queue.enqueue(s2)
```

```
14
15      return result  # For number = 3 , result = {"1","1
16
17
18  print(find_bin(2))
19
```

Start with Enqueuing `1`. Dequeue a number from the queue, append `0` to it, and enqueue it back to queue. Perform the second step, but with appending `1` to the original number and enqueue back to the queue. **The size of Queue should be 1 more than the number** because, for a single number, we're enqueuing two variations of it, one with appended 0 while the other with 1 being appended.

Time Complexity

The time complexity of this solution is in O(n)O(n) as constant-time operations are executed for n times.

Back

Challenge 1: Generate Binary Number...

Next

Challenge 2: Implement Two Stacks I

Completed