Interleave the First Half With the Second Half of the Queue CodeStudio

Approach 1: Function to interleave the elements of a queue using a stack-based approach.

- **Algorithm**:
    - Initialize an empty stack of integers.
    - Calculate the size of the queue.
    - Push the first half of the elements from the queue into the stack.
    - Enqueue back the elements from the stack to the queue.
    - Dequeue the first half of the elements from the queue and enqueue them back.
    - Push the first half of the elements into the stack again.
    - Interleave the elements of the queue and the stack.
- **Time Complexity**: **O(n)**
    - **The function iterates through the queue three times, each involving O(n) operations, where 'n' is the number of elements in the queue.**
- **Space Complexity**: **O(n)**
    - **Additional space is used for the stack, which can have up to 'n' elements in the worst case.**

Approach 2: Function to interleave the elements of a queue using a queue-based approach.

- **Algorithm**:
    - Initialize two empty queues: **firstHalf** and **secondHalf**.
    - Calculate the size of the queue.
    - Enqueue the first half of the elements from the queue into **firstHalf**.
    - Enqueue the second half of the elements from the queue into **secondHalf**.
    - Interleave and enqueue elements from both **firstHalf** and **secondHalf** back into the original queue.
- **Time Complexity**: **O(n)**

    - The function iterates through the queue and the two halves once, each involving O(n) operations, where 'n' is the number of elements in the queue.

- Space Complexity: O(n)
    - Additional space is used for the two queues, firstHalf and secondHalf, which together can store up to 'n' elements in the worst case.