29) INTERLEAVE THE FIRST AND THE SECOND HALF OF THE QUEUE

METHOD:
LINK OF EXPLANATION: Interleave the first half of the queue with second half | Stacks & Q.
In this sum, It is specified that we can use only stack as the extra space.
Suppose the queue is 1 2 3 4 5 6 7 8 9 10
So the Main aim of this question is to take the first half of the queue into the queue such that the starting element is at the top.
But we can see that when we push the elements into the stack the first element will go in the bottom of the stack.
So we pop the elements again and push the elements into the queue. So the queue becomes
6 7 8 9 10 5 4 3 2 1
So we move the first half again into the back by simultaneously popping and pushing the elements into the queue.
5 4 3 2 1 6 7 8 9 10
Now we push the first half again into the stack. So we get the first element of the original queue into the top of the stack.
Then we pop one one element alternately from the stack and queue and push them into the queue
So the queue becomes : 1 6 2 7 3 8 4 9 5 10

CODE OF THE METHOD:

```
#include<iostream>
#include<stack>
#include<queue>
using namespace std;

void fun(queue<int> &q){
   stack<int> s;
   int n=q.size();
   for(int i=1;i<=n/2;i++){
      s.push(q.front());
      q.pop();
   }
   while(s.size()!=0){
      q.push(s.top());
      s.pop();
   }
   for(int i=1;i<=n/2;i++){
      q.push(q.front());
      q.pop();
   }
   for(int i=1;i<=n/2;i++){
      s.push(q.front());
      q.pop();
   }
   for(int i=1;i<=n/2;i++){
      q.push(s.top());
      q.push(q.front());
```

```
        q.pop();
        s.pop();
    }
}

int main(){
    int n;
    cout<<"\n Enter the number of elements present in the queue:";
    cin>>n;
    queue<int> q;
    cout<<"\n Enter the elements in the queue:";
    for(int i=0;i<n;i++){
        int t;
        cin>>t;
        q.push(t);
    }
    fun(q);
    cout<<"\n The Interleaved queue:";
    for(int i=0;i<n;i++){
        cout<<q.front()<<" ";
        q.pop();
    }
    return 0;
}
```