
I used the stack method to create this result. A stack is a linear data structure in which elements may only be added or removed from the top side of the list. The LIFO (Last In First Out) principle dictates that the element put last is the first to come out of a stack. Pushing an element into a stack is referred to as a push operation, while removing an element from a stack is referred to as a pop operation. With a pointer named top, we always maintain track of the last entry in the list in stack.


I used the queue technique to create this output. A queue is a linear data structure in which entries can only be entered from one side of the list, called the back, and only removed from the other side, called the front. The FIFO (First In First Out) principle governs the queue data structure, which means that the element placed first in the list is the first one withdrawn from the list. Enqueue operations are used to add items to a queue, and dequeue operations are used to remove items from a queue. We always keep two pointers in queue, one pointing to the element that was placed first and is still present in the list (the front pointer), and the other pointing to the element that was inserted last (the rear pointer).

The front and rear arrows of the queue wrap around to the beginning of the array to accomplish this outcome of not being able to insert an item even if the queue is not full. This is known as a ring buffer or circular queue. The new element is always added at the rear of a circular queue.