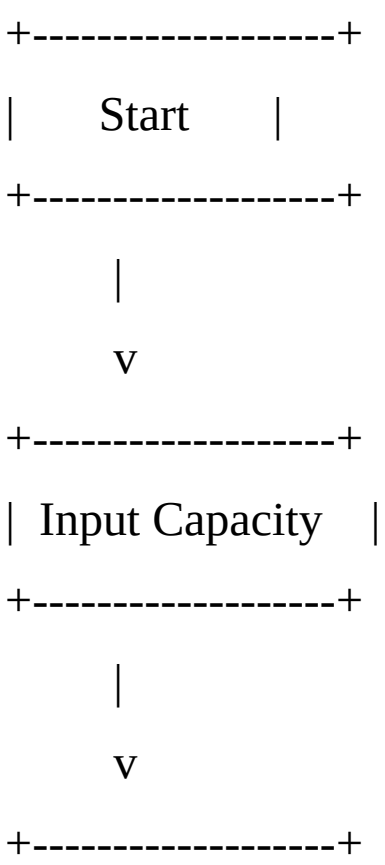
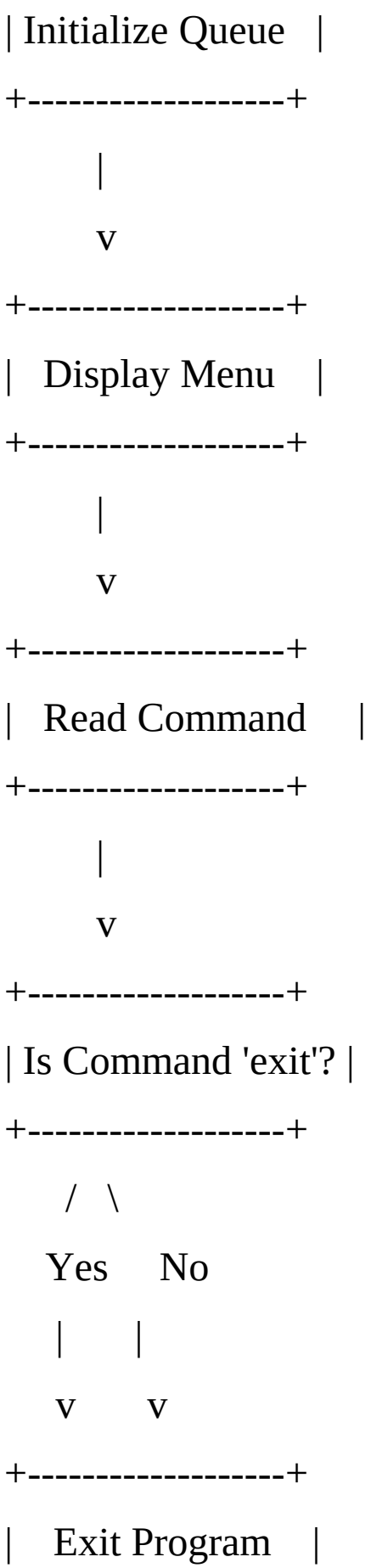
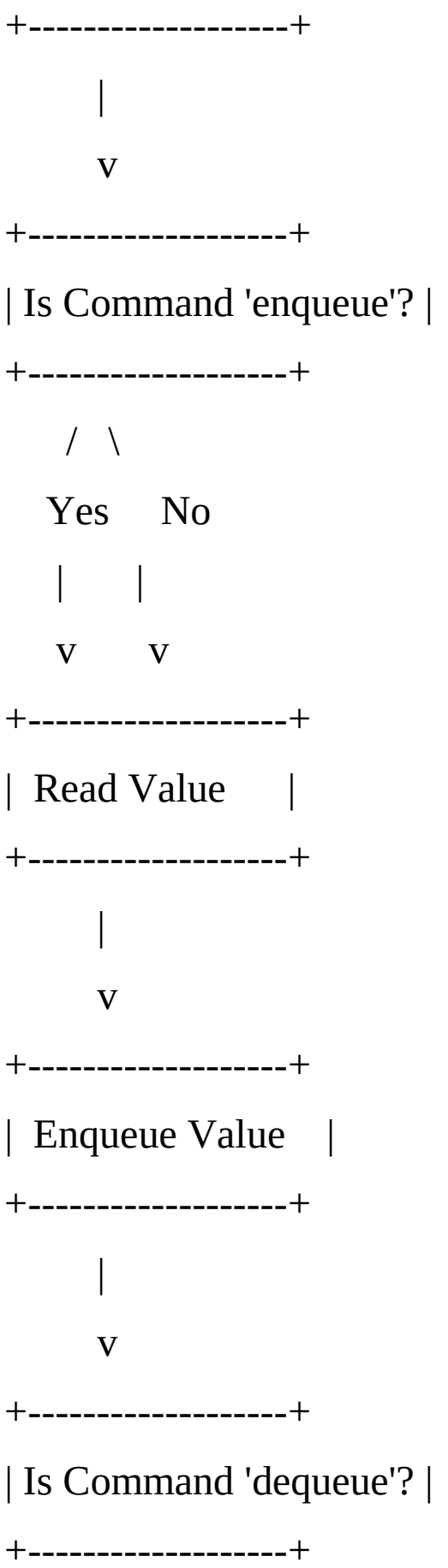
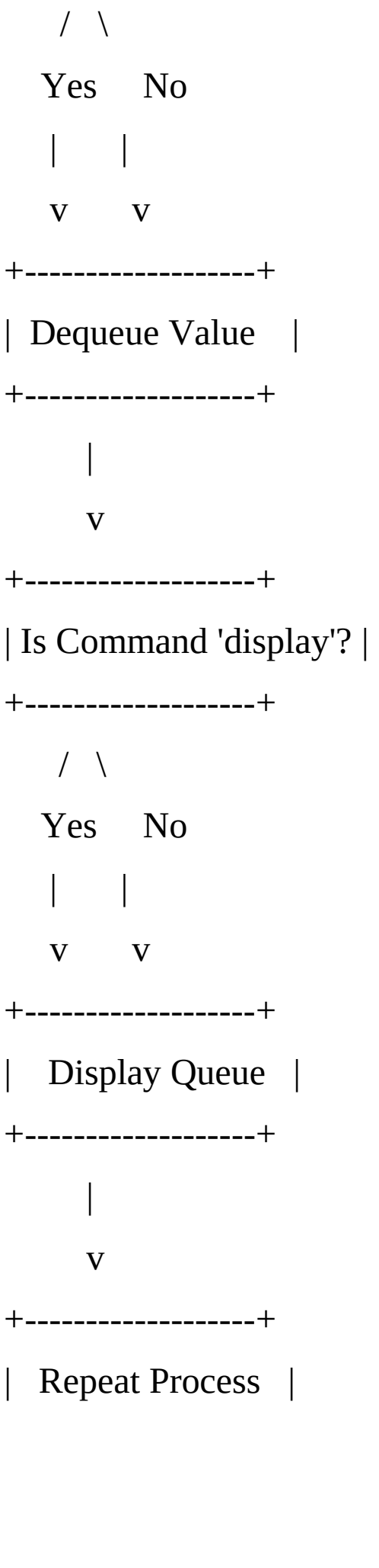
❖ Circular Queue:

- Explanation- The problem is to implement a circular queue data structure using an array, which allows for efficient utilization of space by treating the array as circular. This implementation supports the basic operations of adding (enqueue) and removing (dequeue) elements while maintaining a first-in, first-out (FIFO) order. The program should provide an interactive interface for users to manipulate the queue and display its contents.
- Time Complexity- Enqueue: O(1)
  Dequeue: O(1)
  Display: O(n)
- Space Complexity- O(n)
- Flowchart-




Initialize Queue
Display Menu
Read Command
Is Command 'exit'?
Yes
No
Exit Program


Is Command 'enqueue'?
Yes
No
Read Value
Enqueue Value
Is Command 'dequeue'?


/ \
Yes No
| |
v v
+------------------+
| Dequeue Value |
+------------------+
|
v
+------------------+
| Is Command 'display'? |
+------------------+
/ \
Yes No
| |
v v
+------------------+
| Display Queue |
+------------------+
|
v
+------------------+
| Repeat Process |