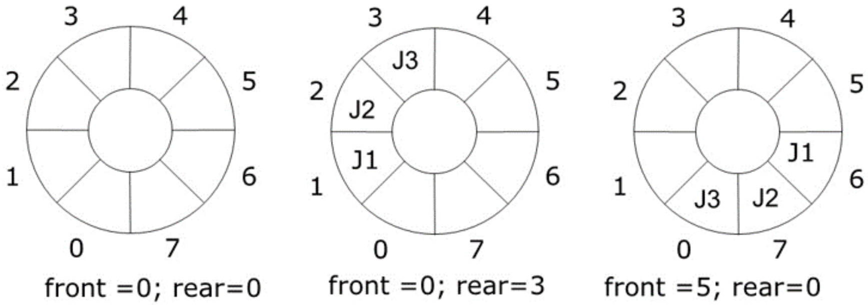
Practice Lessons

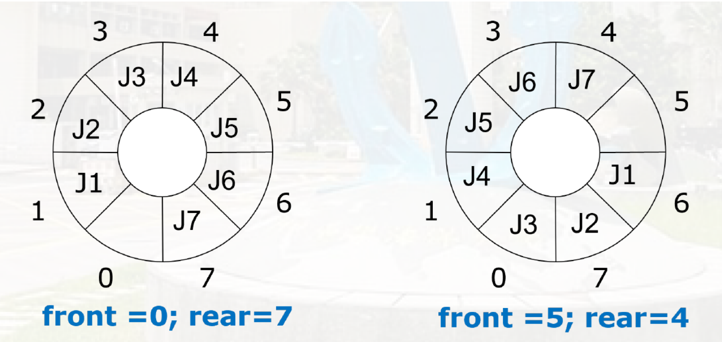
Oct. 9 2025

NTOU CSE

1). The Circular Queue



Condition of FULL



Note: using STL like `std::queue` is forbidden.

Input

- First line: an integer `N` > 0, denoting the queue size

  Followed by the inputs as below:

  `1 x` --> enqueue x (print “`FULL`” if the queue is full)

  `0` --> pop the circular queue and print the popped number; print “`EMPTY`” if the queue is empty

  `-1` --> the end of the input

Output

- Print the outcome after each step.

2). Maze

- Input:
  - Two integers `N` and `M` in first line denoting the size of the maze.
  - Starting from the second row:
    `N` rows of `M` integers (0: available position or 1: unavailable position like walls) separated by space. (The `1`th and `N`th row and the `0`th and `M`th columns are `1`'s)
  - Last line: four integers representing the positions of the entrance and the exit.
- Output:
  - The path (constituted by a sequence of positions from the entrance to the exit, each position is represented by a 2-tuple) or print "`None`" if no such path exists.

Sample input:

```
7 7
1 1 1 1 1 1 1
1 0 1 1 1 0 1
1 1 0 0 0 1 1
1 1 1 0 1 1 1
1 0 0 1 0 1 1
1 1 1 1 1 0 1
1 1 1 1 1 1 1
1 1 5 5
```

Sample output:

```
(1,1) (2,2) (2,3) (2,4) (3,3) (4,4)
```

Sample input:

```
5 6
1 1 1 1 1 1
1 0 0 1 0 1
1 0 0 1 1 1
1 0 0 0 0 1
1 1 1 1 1 1
3 1 1 4
```

Sample output:

```
None
```