Trial one - state retained on concurrent run

Used bash for output for some highlighting

```
> Executing task in folder c: gcc '/mnt/c/Users/Islam/Documents/60days/c/ubuntu/os/psync/queue.c' -o /mnt/c/Users/Islam/Docum


Terminal will be reused by tasks, press any key to close it.

> Executing task in folder c: ./queue.out <


 Enter how many times you want the process to run2

 Enter 1 for producer, 2 for consumer, 3 for both, and 4 to exit1

 Producer produces item 1
Producer: x=1, full=1, empty=9

 Producer produces item 2
Producer: x=2, full=2, empty=8

 Enter 1 for producer, 2 for consumer, 3 for both, and 4 to exit1

 Producer produces item 3
Producer: x=3, full=3, empty=7

 Producer produces item 4
Producer: x=4, full=4, empty=6


 Enter 1 for producer, 2 for consumer, 3 for both, and 4 to exit2

Consumer consumes item 3
Consumer: x=3, full=3, empty=7
```

```
Consumer consumes item 2
Consumer: x=2, full=2, empty=8

 Enter 1 for producer, 2 for consumer, 3 for both, and 4 to exit1

 Producer produces item 3
Producer: x=3, full=3, empty=7

 Producer produces item 4
Producer: x=4, full=4, empty=6

 Enter 1 for producer, 2 for consumer, 3 for both, and 4 to exit3

 Producer produces item 5
Producer: x=5, full=5, empty=5

Consumer consumes item 4
Consumer: x=4, full=4, empty=6

 Producer produces item 5
Producer: x=5, full=5, empty=5

Consumer consumes item 4
Consumer: x=4, full=4, empty=6

 Enter 1 for producer, 2 for consumer, 3 for both, and 4 to exit4

 ERR: Invalid choice
```

From the outputs the following outliers should be noted:

- The last run where we do both consumer and producer is idempotent (doesn't change the overall state of the system) as expected. i.e our last consume is the same as the last produce before we started offf
- The other two runs of consumer/producer change the state of the system permanently.

Inputs: n=2, `choice={1,2,3,4}` In order.

Further confirmation can be done by testing another round of consume after the last choice in this trial.