Aayush Kumar
CO21BTECH11002
Theory Assignment 1

1)

An example to show that the given implementation is not linearizable can be:

1. Thread A gets the index for enqueuing as index 0.
2. Thread B gets the index for enqueuing as index 1.
3. Thread B enqueues item2.
4. Thread C tries to dequeue before thread A has enqueued.
5. It dequeues at the index of ‘head’ which is currently 0.
6. Thread C finds the queue to be empty and throws an exception.

Even though thread A called enqueue before thread B, the FIFO property is not satisfied here. Hence, the method is not linearizable.

2)

For line 15:

1. Thread A gets the index using getAndIncrement() and gets, say, 0.
2. Thread B gets the index using getAndIncrement() and gets, say, 1.
3. Now, before thread A enqueues, thread B enqueues item2 at index 1.
4. Thread C tries to dequeue before thread A enqueues.
5. Thread C finds index 0 empty but index 1 full.
6. Thread C dequeues item2.
7. Thread A now enqueues item1 at index 0.
8. Thread C now on trying to dequeue, finds index 0 to be full.
9. Thread C dequeues item1.

For line 16:

1. Thread A gets the index using getAndIncrement() and gets, say, 0.
2. Thread B gets the index using getAndIncrement() and gets, say, 1.
3. Thread A enqueues item1 at index 0.
4. Thread A enqueues item2 at index 1.
5. Now, thread C tries to dequeue.
6. Thread C finds index 0 to be filled and dequeues item1.
7. Thread C finds index 1 to be filled and dequeues item2.

Even though enq() do not have any single linearization point, it does not mean that it is not linearizable. If we can find a point at which linearizability does not break then we can call the method to be linearizable. Above executions just show that there isn’t any single linearization point that works for all method calls.