BJIT

Omar Faruk Pial (30127)

1. Answer: There can be a deadlock. As the Producer class is adding value in buffer until it gets equal to maxSize and if it gets maxSize (buffer is full) then it waits the buffer to get notify. Similarly the Consumer removes value until it gets empty, if buffers gets empty the buffer waits to get notify. If the Producer class's buffer is full and it is trying to add new value in it, also the Consumer class's buffer is empty and it is trying to remove value then that can create a deadlock. To prevent this we can use other object, like Object lock, and we will synchronize the lock, wait the lock and notifyAll the lock. That will helps us to prevent deadlock.

2. We can wait till the buffer reduced to not full state. If it is not full only then we will add value. We can use BlockingQueue object to handle this.

3. 3.We can wait till the buffer becomes not empty. If it is not empty only then we will remove value. We can use BlockingQueue object to handle this.

4. Synchronization is mainly used for concurency. It is necessary for reliable communication between threads. Here the buffer synchronization ensures that the buffer is shared to one thread at a time, when the task of the thread returns, then the other threads can use the buffer.

5. Mainly, notifyAll() sends notification to all the threads that's are waiting. And the notify sends notification to only one thead that's are waiting. The main potential risk of using notify instead of notifyAll is that, notify sends notification randomly, to which thread it will send notification that is uncertain. That's why we needs to use the notifyAll here.