EECE.4810/EECE.5730: Operating Systems

Spring 2017

Lecture 6: Key Questions
February 6, 2017

1. (Review) Explain what synchronization is and why it's necessary. Use the "Too much milk" problem as an example.

2. What is mutual exclusion?

3. What is a critical section?

4. Analyze the solution below to the "too much milk" problem. Does this solution work?

```
Peter                          Janet
if (noNote) {                  if (noNote) {
   leave note                     leave note
   if (noMilk) {                  if (noMilk) {
      buy milk                       buy milk
   }                              }
   remove note                    remove note
}                              }
```

5. Analyze the solution below to the "too much milk" problem. Does this solution work?

```
Peter                          Janet
leave notePeter                leave noteJanet
if (no noteJanet) {            if (no notePeter) {
   if (noMilk) {                  if (noMilk) {
      buy milk                       buy milk
   }                              }
   remove notePeter               remove noteJanet
}                              }
```

6. Analyze the solution below to the "too much milk" problem. Does this solution work? What are the benefits? What are the problems?

```
Peter                          Janet
leave notePeter                leave noteJanet
while (noteJanet) {
   do nothing
}                              if (no notePeter) {
if (noMilk) {                     if (noMilk) {
   buy milk                          buy milk
}                                 }
remove notePeter               }
                               remove noteJanet
```

7. What is a lock, or mutex? Explain how a lock can be used to solve the “too much milk” problem.

8. Describe how to implement a thread-safe queue with locks.

9. Explain fine-grained locking.

10. Suppose you wanted the dequeue() function to wait if the queue is empty. How can you avoid busy waiting?

11. Describe condition variables and the operations one can perform on them.

12. What is a monitor? How can you implement a thread-safe queue using monitors?