

1. What is the difference between semaphores and condition variables?
2. A simple implementation of a lock is given below. Why must the interrupts be disabled in the **Release** procedure?

```
class Lock {
    int value = FREE;

    Acquire() {
        Disable interrupts
        if (value == BUSY) {
            put on wait queue, sleep, disable
interrupts
        } else {
            value = BUSY;
        }
        Enable interrupts;
    }

    Release() {
        Disable interrupts;
        if (wait queue not empty) {
            take thread off wait queue, add to
ready queue
        } else {
            value = FREE;
        }
        Enable interrupts;
    }
}
```

3. Suppose we had the following situation:

- There is a box of donuts on the counter.
- Jim goes to the kitchen to eat a donut.
- Fred also goes to the kitchen to eat a donut.
- Bob the health nut doesn't eat donuts, but instead puts a single donut into the donut box every so often for his roomies to eat.

How would you write the code for the threads corresponding to Jim, Fred, and Bob so that each of them can fulfill their tasks? Hint: you might find the synchronization types we talked about in class helpful!