

Too Much Milk With Locks

Both threads:

```
struct lock l;
```

```
...
```

```
lock_acquire(&l);  
if (milk == 0) {  
    buy_milk();  
}  
lock_release(&l);
```

Producer/Consumer, v1

```
char buffer[SIZE];
int count = 0;
int putIndex = 0, getIndex = 0;
struct lock l;
lock_init(&l);

void put(char c) {
    lock_acquire(&l);
    count++;
    buffer[putIndex] = c;
    putIndex++;
    if (putIndex == SIZE) {
        putIndex = 0;
    }
    lock_release(&l);
}
```

```
char get() {
    char c;
    lock_acquire(&l);
    count--;
    c = buffer[getIndex];
    getIndex++;
    if (getIndex == SIZE) {
        getIndex = 0;
    }
    lock_release(&l);
    return c;
}
```

Producer/Consumer, v2

```
char buffer[SIZE];
int count = 0;
int putIndex = 0, getIndex = 0;
struct lock l;
lock_init(&l);

void put(char c) {
    lock_acquire(&l);
    while (count == SIZE) {
        lock_release(&l);
        lock_acquire(&l);
    }
    count++;
    buffer[putIndex] = c;
    putIndex++;
    if (putIndex == SIZE) {
        putIndex = 0;
    }
    lock_release(&l);
}
```

```
char get() {
    char c;
    lock_acquire(&l);
    while (count == 0) {
        lock_release(&l);
        lock_acquire(&l);
    }
    count--;
    c = buffer[getIndex];
    getIndex++;
    if (getIndex == SIZE) {
        getIndex = 0;
    }
    lock_release(&l);
    return c;
}
```

Producer/Consumer, v3

```
char buffer[SIZE];
int count = 0;
int putIndex = 0, getIndex = 0;
struct lock l;
struct condition charAdded;
struct condition charRemoved;

lock_init(&l);
cond_init(&charAdded);
cond_init(&charRemoved);

void put(char c) {
    lock_acquire(&l);
    while (count == SIZE) {
        cond_wait(&charRemoved, &l);
    }
    count++;
    buffer[putIndex] = c;
    putIndex++;
    if (putIndex == SIZE) {
        putIndex = 0;
    }
    cond_signal(&charAdded, &l);
    lock_release(&l);
}
```

```
char get() {
    char c;
    lock_acquire(&l);
    while (count == 0) {
        cond_wait(&charAdded, &l);
    }
    count--;
    c = buffer[getIndex];
    getIndex++;
    if (getIndex == SIZE) {
        getIndex = 0;
    }
    cond_signal(&charRemoved, &l);
    lock_release(&l);
    return c;
}
```

