

```
#include "Buffer.h"
#include "events.h"
#include "protocol.h"

Buffer::Buffer():count_(0)
{
    InitializeCriticalSection(&cs_);
    hEvent_ = CreateEvent(NULL, TRUE, FALSE, COMMOUT_START_EVENT);
}

void Buffer::send(Packet p) {
    EnterCriticalSection(&cs_);
    packetList_.push_front(p);

    if(count_ == 0) { // buffer is empty
        SetEvent(hEvent_);
    }

    count_++;
    LeaveCriticalSection(&cs_);
}

Packet Buffer::peek() {
    EnterCriticalSection(&cs_);
    Packet p = packetList_.back(); // Kyle is a goof
    LeaveCriticalSection(&cs_);
    return p;
}

void Buffer::pop() {
    EnterCriticalSection(&cs_);
    packetList_.pop_back();

    count_--;
    LeaveCriticalSection(&cs_);
}

bool Buffer::danger() {
    return count_ >= BUFFMAX;
}

bool Buffer::safe() {
    return count_ <= BUFFMIN;
}

bool Buffer::empty() {
    if(count_==0)
        return true;
}
```

```
    return false;  
}
```