CS736 Quiz #1

1. Briefly (using three sentences) describe the motivation for the design of scheduler activation.
2. In scheduler activation, kernel uses upcalls to inform user-level thread library of events affecting thread scheduling. List the four upcall points (that is, upcall functions) used by kernel.
3. List the interactions between kernel and user-level thread library when the kernel takes a processor away from the application.

Synchronization

Why?

How to Synchronize

- Example: get_freeblock/free_block
  - free blocks organized as a double-linked list

Basic Concepts

- Critical sections
  - busy-waiting versus sleep&wakeup;
- semaphores
  - essentially critical sections + sleep/wakeup
- monitors

Monitors

- encapsulate the lock, data structures, and the procedures all in one place
- entry procedures
- conditional variable

Semaphores by Monitors

```
single resource: monitor
begin: busy: boolean; nonbusy:condition;
procedure acquire
  begin   if busy then nonbusy.wait
          busy :=true;
  end
procedure release
  begin   busy := false; nonbusy.signal; end
```

Monitors by Semaphores

Semantics of Signal

Proof Rules

- assertions
- wait: A {b.wait} A&B
- signal: A&B{b.signal} A

Example 1

- Bounded buffers:

```
append:  if (count==N) nonfull.wait;
    buffer[lastptr]=x; (lastptr++)%=N; count++;
         nonempty.signal;
remove:  if (count==0) nonempty.wait;
    x=buffer[(lastptr-count)%N]; count--;
         nonfull.signal;
```

Example 2

- Disk head scheduler
  - request:

```
if busy { if ( headpos <= dest & up) upsweep.wait(dest)
          else downsweep.wait(cylmax-dest); } …
```

  - release:

```
if (up) { if (upsweep.queue) upsweep.signal
          else {down; downsweep.signal;}
```