HW 1: Processes, Threads, and Concurrency

Due: In class on Tuesday, 2/10/09

February 3, 2009

1 Scheduling

In class we stated that First Come First Serve (FCFS) was a fair scheduling algorithm because it prevents starvation. That is to say that every ready job in the system will eventually be scheduled to run. Determine whether each of the following scheduling algorithms is fair and explain your answer.

a) Shortest Job First (SJF)

b) Processor Sharing (PS)

c) Priority Scheduling (PRI)

2 Who needs locks? - Let's use memory

In this question we consider possible approaches to solving concurrency problems without using traditional locks such as mutexes and semaphores. In each of these problems we will assume there are exactly two threads, with IDs 0 and 1. We will assume that each thread has a `my_id` variable, set to the value of it's thread id, and a `their_id` variable, set value of the other thread's ID. For each of the approaches below you should determine whether the approach guarantees (i) mutual exclusion, (ii) forward-progress, and/or (iii) bounded waiting. Be sure to explain your answers carefully.

a) Taking Turns

```
/* global variables */
volatile int turn = 0;
```

```
/* code snippet running on each thread */
...
while (turn != my_id)
  continue;

CRITICAL SECTION

turn = their_id;
...
```

b) Being Polite

```
/* global variables */
volatile boolean want[2] = {false, false};
```

```
/* code snippet running on each thread */
...
want[my_id] = true;
while (want[their_id])
  continue;

CRITICAL SECTION

want[my_id] = false;
...
```

3 Internet orders and semaphores

As Valentine's Day approaches, your favorite campus student organization decides to raise some extra money selling chocolate and flowers. In order to get an edge over your competition on campus, you plan to setup a website where orders can be placed. The goods can then be delivered to students as they become available.

For this problem you will need to solve this concurrency control situation using semaphores. Please do not use other concurrency control tools, such as mutexes, condition variables, atomic instructions, etc. Your solution should enforce the policies described below while avoiding deadlock, starvation, livelock, etc.

When an order is placed on the website, the placeOrder() function is called. This function allocates the necessary quantities of chocolate and flowers to fulfill the order. It then alerts a volunteer from the student organization that the order is ready to be packed and delivered.

```
void placeOrder(unsigned int chocolateBoxes, unsigned int roses)
{
  ...
}
```

As orders come in, volunteers are notified that the orders are ready, and package and deliver these orders by calling packOrder() and doDelivery(). These volunteers run the packAndDeliverOrders() function:

```
void packAndDeliverOrders()
{
  while(beforeValentinesDay) {
    ...
  }
}
```

Finally, a second set of volunteers buys the boxes of chocolate and the roses that are to be resold on campus whenever they decide that more are needed. When they do this, they will call the boughtChocolates() and boughtRoses() functions to update the records as to what is available.

```
void boughtChocolate(unsigned int chocolateBoxes) {
  ...
}

void boughtRoses(unsigned int roses) {
  ...
}
```

Your job is to implement these four functions (placeOrder(), packAndDeliverOrders(), boughtChocolate(), and boughtRoses()) using semaphores. You may introduce whatever global variables you may need and you are free to implement and init() function if you would like.

4 Computer Lab Monitor

In this problem we consider a computer lab with a variety of available accessories - scanners, memory card readers, and video capture devices. When a user logs into a computer in the lab they specify which (if any) accessories they would like access to while in the lab. The lab computers all coordinate via a central server to allocate the required devices. You will need to write the code running on this server using a monitor. (Never mind that students may be standing in front of their computer waiting a very long time to log on. They can play video games on their cell phone or something...)

The computer lab contains 5 scanners, 8 memory card readers, and 3 video capture devices. There are 3 different classes that have students making use of the lab - photography, filmmaking, and digital music. The photography students require access to a scanner and a memory card reader. The filmmaking students need to use a video capture device and a memory card reader. The digital music students will need access only to a memory card reader.

Choose your favorite monitor semantics and use them to write a solution. Be sure to tell us which semantics you have chosen. You will need to write a logon and logoff function for each type of student. You may use any global variables you need, but be sure to tell us what they are.