CS414 Assignment #1 **Due: 07/26/2000**

1) In class we studied the Bakery algorithm to solve the critical section problem for multiple processes. It may seem that some of the variables used are redundant. Suppose the Bakery algorithm is modified so that references to the Boolean array *chosen* is commented, and the new algorithm is:

```
 1 /* var choosing: shared array[0..n-1] of boolean; */
 2     number: shared array[0..n-1] of integer;
       ...
 3 repeat
 4     /* choosing[i] := true; */
 5     number[i] := max(number[0],number[1],...,number[n-1]) + 1;
 6     /* choosing[i] := false; */
 7     for j := 0 to n-1 do begin
 8         /* while choosing[j] do no_op; */
 9         while number[j] != 0 and
10                   (number[j], j) < (number[i],i) do no_op;
11    end;
12    (* critical section *)
13    number[i] := 0;
14    (* remainder section *)
15 until false
```

Will this modified algorithm effectively solve the critical section problem? If yes, prove it; else give a case in which this algorithm will fail!!

2) In this question you will solve another classical problem: **THE SLEEPING BARBER PROBLEM.** Assuming that the barber and his customers are male, the problem is defined as:

A barber provides a hair-cutting service. He has a shop with two doors, an entrance and an exit. He spends his time serving customers one-at-a-time. When none are in the shop, the barber sleeps in the barber's chair.

When a customer arrives and finds the barber sleeping, the customer awakens the barber and sits in the barber's chair to receive his haircut. After the cut is done, the barber sees the customer out through the exit door.

If the barber is busy when a customer arrives, the customer waits in one of the chairs provided meant for waiting customers. If they are all occupied he goes away.

After serving a customer the barber looks to see if any are waiting and if so proceeds to serve one of them. Otherwise, he sleeps again in his chair.

a) Give a solution to this problem using semaphores
b) Modify the solution in (a) to give an implementation of your solution using monitors.

(Hint: The manipulation of shared variables in the barber and customer process has to be in the critical section. Use two counting semaphores to keep track of the state of the barber and customers. For the monitor solution, it is advisable to adapt your solution of (a).)

3) A priority based preemptive scheduling algorithm is proposed in which processes change their priority at the rate of *'x'* when they are waiting in the ready queue and at a rate *'y'* when the CPU is serving them. Comment on the performance of this algorithm for different values of *'x'* and *'y'*. ($x>0$, $y>0$)