Operating Systems

Homework #4
Due: 2004/5/12

1. Explain why spinlocks are not appropriate for uniprocessor systems yet may be suitable for multiprocessor systems.

2. Prove that, in the bakery algorithm, the following property holds: If P_i is in its critical section and P_k ($k \neq i$) has already chosen its number[k] $\neq$ 0, then (number[i], i) < (number[k], k).

3. The first known correct software solution to the critical-section problem for two processes was developed by Dekker. The two processes, P_0 and P_1, share the following variables:

```
boolean flag [2]; /* initially false
int turn;
```

   The structure of process P_i (i == 0 or 1), with P_j (j == 1 or 0) being the other process, is shown in the following figure. Prove that the algorithm satisfies all three requirements for the critical-section problem.

```
do {
    flag [i] = true;
    while (flag [j]) {
          if (turn == j) {
                flag [i] = false;
                while (turn == j);
                flag [i] = true;
          }
    }

          critical section

    turn = j;
    flag [i] = false;

          remainder section

} while (1);
```

4. The Sleeping-Barber Problem. A barbershop consists of a waiting room with *n* chairs and the barber room containing the barber chair. If there are no customers to be served, the barber goes to sleep. If a customer enters the barbershop and all chairs are occupied, then the customer leaves the shop. If the barber is busy but chairs are available, then the customer sits in one of the free chairs. If the barber is asleep, the customer wakes up the barber. Write a program to coordinate the barber and the customers.

5. Write a bounded-buffer monitor in which the buffers (portions) are embedded within the monitor itself.