

# Problem Set 7

CS 411

Due at the beginning of class on the first class day of the following week.

Sections 6.1–6

1. Race conditions are possible in many computer systems. Consider a banking system that maintains an account balance with two functions: `deposit(amount)` and `withdraw(amount)`. These two functions are passed the amount that is to be deposited or withdrawn from the bank account balance. Assume that a husband and wife share a bank account. Concurrently, the husband calls the `withdraw()` function and the wife calls `deposit()`. Using a specific example, describe how a race condition is possible. What might be done to prevent the race condition from happening?
2. The mutual exclusion and progress requirements may appear to provide everything needed for a solution to the critical section problem. Why is the bounded waiting requirement necessary?
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] = { false, false };
int turn = 0;
```

The structure of process  $P_i$  ( $i = 0$  or  $1$ ) is shown below. The other process is  $P_j$  ( $j = 1$  or  $0$ ). 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 (true);
```