CS6868 Concurrent Programming

Homework 1: Due 8th February 2013

1. Consider N processes numbered 0 to N-1 in which each process i executes

   A[i] = 1
   B[i] = A[(i-1) mod N]

   If all the reads and writes to A[i] are atomic, what can you say about the values in B at the end when all processes are done.

2. There are S students who are held as captives by a despotic teacher inside a classroom complex. There is a lab with one computer which is initially switched OFF. The students are held in isolated cells and talking to each other is impossible. Each day, the teacher picks a student at random and takes the student to the lab where he/she can turn the computer ON or OFF as per their desire or just leave it in the state it is in. Nobody else can enter the room except the student that the teacher picked. The students are picked uniformly at random by the teacher, and for any number N, each student is guaranteed to be taken to the lab at least N times. The despot tells the students that at any point any student can assert that all the students have indeed visited the lab. If the assertion is false, they will be asked to take a graduate level course CS8686 taught by the teacher. If it is true, all the students are freed to get along with their interesting lives where they spend solitary lives inside their rooms in the hostels.

   The students are allowed to talk to each other once in the beginning and come up with a strategy.

   a) Devise a strategy for the students when the initial state is that the computer is turned OFF.
   b) Devise a strategy for the students when the initial state of the computer is unknown.

3. You are aware of the dining philosophers problem (all standard texts on synchronization have it). The traditional version is that two philosophers must not hold the chopstick at the same time. The textbooks also discuss solutions for this.
   a) Describe one such solution
   b) Does the solution guarantee that no philosopher will starve? If not, can you give a solution in which both the conditions are met?

4. Why should you as a student in CS6868 be able to solve Problem 2? What is the relevance to the course CS6868?