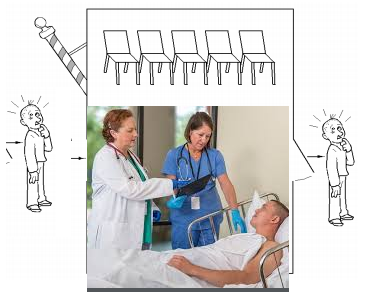
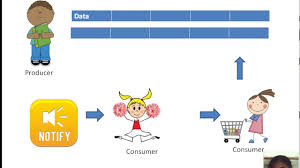
**PROCESS SYNCHRONIZATION QUESTIONS**

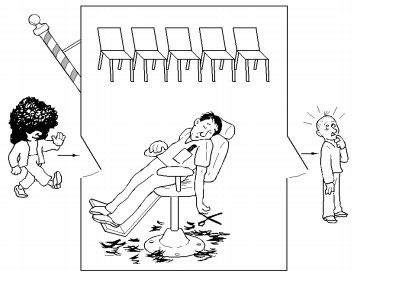
**From the below question you should relate to the classical synchronization solution and try to solve.**

1. Consider a clinic with one doctor and a very large waiting room (of infinite capacity). Any patient entering the clinic will wait in the waiting room until the doctor is free to see her. Similarly, the the doctor also waits for a patient to arrive to treat. All communication between the patients and the doctor happens via a shared memory buffer. Any of the several patient processes, or the doctor process can write to it. Once the patient “enters the doctor’s office”, she conveys her symptoms to the doctor using a call to consultDoctor(), which updates the shared memory with the patient’s symptoms. The doctor then calls treatPatient() to access the buffer and update it with details of the treatment. Finally, the patient process must call noteTreatment() to see the updated treatment details in the shared buffer, before leaving the doctor’s office. A template code for the patient and doctor processes is shown below. Enhance this code to correctly synchronize between the patient and the doctor processes. Your code should ensure that no race conditions occur due to several patients overwriting the shared buffer concurrently



2. Consider a producer-consumer situation, where a process P produces an integer using the function produceNext() and sends it to process C. Process C receives the integer from P and consumes it in the function consumeNext(). After consuming this integer, C must let P know, and P must produce the next integer only after learning that C has consumed the earlier one. Assume that P and C get a pointer to a shared memory segment of 8 bytes, that can store any two 4-byte integer-sized fields, as shown below. Both fields in the shared memory structure are zeroed out initially. P and C can read or write from it, just as they would with any other data object. Briefly describe how you would solve the producer-consumer problem described above, using only this shared memory as a means of communication and synchronization between processes P and C. You are provided template code below which gets a pointer to the shared memory, and produces/consumes integers. You must write the code for communicating the integer between the processes using the shared memory, with synchronization logic as required.



3. The barber shop has one barber, one barber chair, and n chairs for waiting customers, if any, to sit on. If there are no customers present, the barber sits down in the barber chair and falls asleep, as illustrated in Figure. When a customer arrives, he has to wake up the sleeping barber. If additional customers arrive while the barber is cutting a customer's hair, they either sit down (if there are empty chairs) or leave the shop (if all chairs are full). The problem is to program the barber and the customers without getting into race conditions

4. Three engineering professors have gone to the faculty club to eat licorice. Each piece of licorice costs 36 cents. To buy a piece of licorice, a professor needs a quarter, a dime, and a penny (they do not give change, and they don’t take American Express). The first professor has a pocket full of pennies, the second a supply of quarters, and the third a supply of dimes. A wealthy alum walks up to the bar and lays down at random two of the three coins needed for a piece of licorice. The professor with the third coin takes the money and buys the licorice. The cycle then repeats. Show how to synchronize the professors and the alum



5. A group of students are studying for a CPS 110 exam. The students can study only while eating pizza. Each student executes the following loop: while (true) { pick up a piece of pizza; study while eating the pizza}. If a student finds that the pizza is gone, the student goes to sleep until another pizza arrives. The first student to discover that the group is out of pizza phones Satisfactions at Brightleaf to order another pizza before going to sleep. Each pizza has S slices. Write code to synchronize the student s and the pizza delivery.

