Indian Institute of Technology Jodhpur
Operating Systems Lab (CSL 3030)
Assignment 7

Dated 26th October, 2021 **Total marks: 30**

In this assignment, you will solve the problems of synchronization and deadlock.
Part-I: to be solved in the lab

Late-Night Pizza. A group of students are studying for a CSL 3030 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 makes a call to Pizza Hut to order another pizza before going to sleep. Each pizza has *S* slices.
Write code to synchronize the student threads and the pizza delivery thread. Your solution should avoid deadlock and make a call to Pizza Hut (i.e., wake up the delivery thread) exactly once each time a pizza is exhausted. No piece of pizza may be consumed by more than one student.

Part-II

1. Refer to the page no. 129 of the Semaphore Book.
Sleeping barbar's synchronization problem is stated as follows: 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. Implement the solution to this classical synchronization problem using Monitor.

2. Write a program in C/C++ language to implement Banker's Algorithm for deadlock avoidance. Your program will read the number of processes, number of resource type and the matrices (Available, Max, Allocation). Your program will also take an input for a process request (process no. and a request string depicting the number of instances required for each resource type). The program will first check whether the input state is safe or unsafe, and then, output whether the stated request can be fulfilled or not.

Deadline: 1st November, 2021, 23:59 hrs

Evaluation:

- Correct implementation of the first program (10pts)
- Use of Monitor for the first one (5pts)
- Correct implementation of the second program (input state is safe/unsafe) (10pts)
- Deciding whether the process request can be granted or not by computing the safe/unsafe state (5pts)