Operating Systems CS220
Assignment 07

Submission deadline: Monday, 28th December, 2020 by 1.00 PM sharp.

Question 1: Playing Cricket

You need at least one bowler and one batsman to play cricket.

There are two threads, one for bowlers and one for batsmen. There is room for only one practice session. A practice session is possible if there is a bowler and a batsman available. If a batsman arrives, and a bowler is already available, he will engage a bowler for a practice session and vice versa. If however, a new batsman arrives and a bowler is not available, the batsman will hold the court and wait for the bowler to arrive and vice versa. No two bowlers can coexist the same time the same way as no two batsmen.

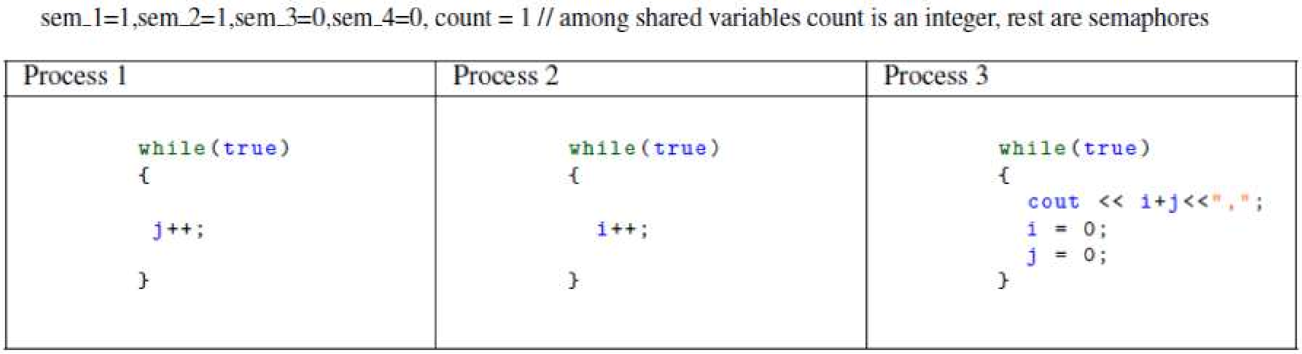
Use the following to start off:

int bowler = 0; int batsman = 0; //shared variables between both threads

Semaphore batsmanSem = 0; bowlerSem = 0, mutex = 1;

Question 2: Process Synchronization

Three processes are running in parallel sharing variables i and j, which are initialized as i = 0 and j = 0. You have to synchronize the following processes such that the output on the console is a continuous string of integers as 3,6,9,12,… The code should be maximally parallel and efficient. Note: You can do anything you like for the solution, except writing a new statement which uses variables i or j on left hand side. You can however, introduce a new variable e.g., k



sem_1=1,sem_2=1,sem_3=0,sem_4=0, count = 1 // among shared variables count is an integer, rest are semaphores

Process 1	Process 2	Process 3

`while(true)`
`{`
` j++;`
`}`
	`while(true)`
	`{`
	`  i++;`
	`}`
		`while(true)`
		`{`
		`  cout << i+j<<",";`
		`  i = 0;`
		`  j = 0;`
		`}`

Question 3: Multithreading

Thread waiting for each other?

Is it possible for a parent to have two threads and one thread waits on another thread to complete? If yes, write the pseudocode for it. If no, why?