Operating System

Assignment – 4

Q1. Three concurrent processes X, Y, and Z execute three different code segments that access and update certain shared variables. Process X executes the P operation (i.e., wait) on semaphores a, b and c; process Y executes the P operation on semaphores b, c and d; process Z executes the P operation on semaphores c, d, and a before entering the respective code segments. After completing the execution of its code segment, each process invokes the V operation (i.e., signal) on its three semaphores. All semaphores are binary semaphores initialized to one. What is a deadlock free order of invoking the P operations by the processes?

Q2. A system contains three programs and each requires three tape units for its operation. What is the minimum number of tape units which the system must have such that deadlocks never arise?

Q3. A system contain four processes uses a multiple-resource Banker’s algorithm for resource allocation. The system has been in operation for some time.

(a) A new process arrives in the system. It is initially not allocated any resources. Is the new resource allocation state of the system safe?

(b) A process is aborted by the OS because it tries to access a file for which it lacks appropriate privileges. Is the new resource allocation state of the system safe?

Q4. In the following system:

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  |  | | --- | --- | --- | --- | |  | R1 | R2 | R3 | | P1 | 3 | 6 | 8 | | P2 | 4 | 3 | 3 | | P3 | 3 | 4 | 4 | | |  |  |  |  | | --- | --- | --- | --- | |  | R1 | R2 | R3 | | P1 | 2 | 2 | 3 | | P2 | 2 | 0 | 3 | | P3 | 1 | 2 | 4 | | |  |  |  | | --- | --- | --- | | R1 | R2 | R3 | | 5 | 4 | 10 | | |  |  |  | | --- | --- | --- | | R1 | R2 | R3 | | 7 | 7 | 10 | |
| **Max Need** | **Allocated Resources** | **Total Allocation** | **Total Exist** |

(a) Is the current allocation state safe?

(b) Would the following requests be granted in the current state?

(i) Process p1 requests (1, 1, 0)

(ii) Process P3 requests (0, 1, 0)

(iii) Process P2 requests (0, 1, 0)

Q5. Consider the traffic deadlock depicted in Figure given below:

a. Show that the four necessary conditions for deadlock hold in this example.

b. State a simple rule for avoiding deadlocks in this system.



Q6. Consider the version of the dining-philosophers problem in which the chopsticks are placed at the center of the table and any two of them can be used by a philosopher. Assume that requests for chopsticks are made one at a time. Describe a simple rule for determining whether a particular request can be satisfied without causing deadlock given the current allocation of chopsticks to philosophers.