Cagurangan, Wyvern D.

BSIT2

Task Performance

1. Describe the deadlock scenario illustrated above based on your understanding. (5 points)

- **Processes P and Q are deadlocked, each waiting for a resource held by the other. Therefore, neither can proceed with their execution, so they are both constant.**

2. What do you think would happen if both Process P and Q needed to get the same resource? (5 points)

- **If Processes P and Q require the same resource, they can execute at the same time as they share it. Deadlocks occur when multiple processes deal for non-sharable resources.**

3. Which concurrency mechanism would you suggest that might prevent the deadlock situation above? Rationalize your answer. (5 points)

- **A condition variable is a recommended concurrency mechanism to prevent deadlocks. It allows a process to be blocked until a specific condition is met. For instance, if Process Q has only one resource and both are needed for execution, a condition variable can be used to block Process Q thus, allowing Process P to acquire both resources, ensuring progress in its execution.**

4. Define in detail the Execution Paths 2 to 6. (5 items x 3 points)

Example: Execution Path 1 – Process Q acquires Resource B and then Resource A. Process Q then releases Resource B and A, respectively.

- **Execution Path 2 – Process Q acquires Resources B and then Resources A. Process Q then delivers both Resources and Process P resumes execution and acquires both Resources.**
- **Execution Path 3 – Process P acquires Resource A and Process Q acquires Resource B that results in a deadlock.**
- **Execution Path 4 – Process Q acquires Resource B and Process P acquires Resource A that results in a deadlock.**
- **Execution Path 5 – Process P acquires then releases both Resources. Process Q continues the execution and acquires both Resources.**
- **Execution Path 6 – Process P acquires both Resources and releases them. Process Q then can acquire both Resources and resume execution.**

5. Do Execution Paths 3 and 4 encompass the first three conditions for a deadlock to occur? Explain your answer. (5 points)

- **Yes, execution Paths 3 and 4 satisfy the initial three conditions required for a deadlock: Mutual Exclusion, Hold and Wait, and No Preemption. These conditions are essential for the deadlock to occur.**

6. If you are to implement deadlock prevention before the processes above reach the critical section, would it be an indirect method or an indirect method? Why? (5 points)

- **I would go for the indirect method because it is simpler to just prevent the three conditions for a deadlock to happen so that it will not occur.**

7. Which deadlock avoidance approach would you suggest for the given situation above and why? (5 points)

- **Resource Ordering because this ensures that processes never enter a circular wait state, breaking the circular wait condition and preventing deadlocks.**

8. Would you agree that deadlock is relative to the number of processes and available resources in an operating system? Why or why not? (5 points)

- **I agree that deadlock occurrence in an operating system is contingent on the interactions and resource demands among processes. A higher number of processes deals for limited resources increases the likelihood of deadlocks, while a system with fewer processes and abundant resources is less prone to such issues. The balance between processes and resources is crucial in determining the potential for deadlocks in an operating system.**

9. If you are asked to reconstruct the progress diagram above to eliminate the critical section, which is the deadlock-inevitable region, which aspect(s) or area(s) would you modify? Explain how the modification eliminates the deadlock. (5 points)

- **I would modify the area where 3 and for are present and I will reroute it to avoid the creation of the inevitable region which is prone to deadlock.**