National University of Computer and Emerging Sciences Chiniot-Faisalabad Campus



Lab 11

CL2006 – Operating System - Lab

|  |  |
| --- | --- |
| Course Instructor | Juhinah Batool |
| Lab Instructor | Juhinah Batool |
| Semester | Fall 2024 |

**FAST School of Computing**

**Department of AI & DS**

**Instructions**

1. Make a PDF document with the convention “ROLLNO\_ LAB#\_ SECTION” and put all your source code and snapshots of its output in it.

2. Plagiarism is strictly prohibited, if you take a code snippet off the internet, mention its reference.

3. Do not discuss solutions with one another. Copying the solution from any source can lead to ZERO marks.

**Lab Task: Understanding and Handling Deadlocks**

**Task 1: Simulating a Deadlock Scenario (30 minutes)**

1. **Objective**: Understand the conditions that lead to a deadlock by simulating one.
2. **Description**:
   * Write a program in C/C++ that simulates a deadlock using two threads and two shared resources.
   * Each thread will:
     + Lock the first resource.
     + Wait for a short delay (simulate some work).
     + Attempt to lock the second resource.
   * The program should demonstrate a deadlock when both threads hold one resource and wait for the other.

**Task 2: Deadlock Detection (1 hour)**

1. **Objective**: Implement a mechanism to detect deadlocks in a system.
2. **Description**:
   * Create a program that simulates a system with multiple processes and resources.
   * Use the **Resource Allocation Graph (RAG)** to represent the system's state.
   * Implement a function to detect a cycle in the RAG, indicating a deadlock.
3. **Instructions**:
   * Represent resources and processes as a graph with adjacency lists/matrices.
   * Implement a cycle detection algorithm (e.g., depth-first search) to identify deadlocks.
4. **Hints**:
   * Use a 2D matrix to represent the graph.
   * Mark visited nodes during DFS to detect cycles.
5. **Expected Output**: The program should print whether a deadlock exists and which processes are involved.