1. **Working Set Calculation and Thrashing Analysis**

Consider 3 processes with a randomly generated reference string between page numbers 1 to 10 (for example, let the page reference string of one process be as follows:

0, 1, 2, 3, 0, 1, 4, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 0, 1

The system uses the Working Set Model for memory management. Assume a fixed working set window size “Delta” time unit.

a) Calculate the Working Set for the process at time units t=4, t=6, t=11, and t=18.

b) If the number of available physical frames (m) is only 3 or 4, and the current total size of the working set of all active processes exceeds m, describe the state the system is likely entering and the characteristic symptom a user would observe. Use the FIFO page replacement algorithm to simulate the performance.

2. **Write a program to detect deadlock with allocation edges, request edges, and uses a graph traversal technique to detect cycles.**

Required Steps

a) Build the Graph: Combine the allocation and request edges into a single adjacency matrix structure representing the RAG. The graph should be directional (since edges only go from P to R or from R to P).

b) Cycle Detection: Implement a graph traversal algorithm (like Depth-First Search (DFS)) designed to detect cycles in a directed graph. The algorithm must:

c) Identify Deadlocked Processes: If a cycle is detected, the function must identify and return the sequence of nodes that form the cycle.

Example Cases:

i) Test Case 1: Deadlock Detected (Circular Wait)

This scenario represents the classic P1 to R1, R1 to P2, P2 to R2, and R2 to P1 cycle.
Allocation Edges: [('R1', 'P1'), ('R2', 'P2')]
Request Edges: [('P1', 'R2'), ('P2', 'R1')]
Expected Output: Deadlock detected! Cycle: P1 -> R2 -> P2 -> R1 -> P1

ii) Test Case 2: No Deadlock (Safe State)

This scenario has resource dependencies but no circular wait. P1 to R2 held by R2 to P2, but R1 to P2, and P2 is not waiting for anything.
Allocation Edges: [('R1', 'P1'), ('R2', 'P2')]
Request Edges: [('P1', 'R2')]
Expected Output: No deadlock detected.