

Lab 3

Banker's Algorithm

Objective:

The goal of this lab exercise is to implement the Banker's Algorithm. The Banker's Algorithm is a deadlock avoidance algorithm used in operating systems to prevent processes from entering a deadlock state.

Background:

The exercise introduces the Banker's Algorithm, which requires information about maximum resource demands, current resource allocations, and available resources.

Tasks:

- Initializing data by accepting user input for allocated, max matrices and total sources vector.
- Calculating the available resources of each type based on allocated resources.
- Initializing arrays for work and process status.
- Implementing a loop to find a safe sequence.
- Printing out the safe sequence and system safety status.

Submission:

After implementing and testing the program, students are required to submit their source code file. (Note: You can code algorithm in any language that you want. Assume that you always provide valid inputs)

Sample Input:

```
Enter the Maximum Matrix:
7 5 3 3 6
3 2 2 2 3
9 0 2 2 2
2 2 2 2 3
4 3 3 3 4
5 3 3 3 5

Enter the Allocated Matrix:
0 1 0 0 0
2 0 0 1 1
3 0 2 1 0
2 1 1 1 0
0 0 2 2 1
0 0 3 0 0

Enter the Total           Resources:
10 5 7 8 9
```

Sample Output:

```
Safe Sequence:
P1 -> P3 -> P4 -> P0 -> P2 -> P5
System is safe.
```