Operating Systems

Lab-6: Implementation of Banker's Algorithm

Implement Banker's algorithm for resource allocation.

Input will be taken from a file named "input.txt". The file contents are as follows.

3 // The first line indicates the number of resources (m)

5 // The second line indicates the number of processes (n)

10 5 7 //The third line indicates the total number of instances of each resource (1*m). Here, there are 10, 5, and 7 instances of R1, R2, and R3 respectively

7 5 3 //The next n lines indicate the maximum requirements of each process (max matrix: n*m)

3 2 2

9 0 2

2 2 2

4 3 3

0 1 0 //The next n lines indicate the allocated resources for each process (allocation matrix: n*m)

2 0 0

3 0 2

2 1 1

0 0 2

For the given test case, the output should be:

16 //The number of safe sequences

1 3 0 2 4 // Safe sequences are printed in lexicographical (dictionary) order

1 3 0 4 2

1 3 2 0 4

1 3 2 4 0

1 3 4 0 2

1 3 4 2 0

1 4 3 0 2

1 4 3 2 0

3 1 0 2 4

3 1 0 4 2

3 1 2 0 4

3 1 2 4 0

3 1 4 0 2

3 1 4 2 0

3 4 1 0 2

3 4 1 2 0

Submission Instructions

Place your C/C++ file and the readme.txt in a folder named as your complete roll number (capital case). Your C file should also be named as [your_roll_number].c. Zip the folder. The zip file should also be named as [your_roll_number].

Note: Zip the folder. Do not zip the files directly.

Upload the zip file.