**Hubei University Of Technology**

**Operating Systems Course Exercise**

**Title：The Banker’s Algorithm**

**Student ID: 1911521213 Name: 王一然 Class: 19 lq CST**

**Description**

Process scheduling is a very important problem in operating system design. Each process requires some amount of resources to run. The process will release all the resources allocated to it when completes. Different resource allocating strategy varies much in efficiency. Even some unsuitable strategy can result in dead lock. Now there are *n* processes and *m* kinds of resources. At the beginning, each process has been allocated some amount of resources for each kind. However, the allocated resources might not be enough. They still need some extra resources for each kind. And you are given the available resources for each kind in the computer now. Can you tell whether it is possible to schedule these processes in a suitable order so that all of them can be executed successfully?

**Input**

There are four parts in the input. The first part contains two positive integers *n* and *m* representing the number of processes and the number of resources. The second part is the following line which contains *m* integers, called resource vector, namely, the total number for each resource. The third part is the following *n* lines. Each line contains *m* integers. These integers make the claim matrix. The last part is the rest *n* lines. Each line contains *m* integers. These integers make the allocation matrix.

**Output**

If all processes can be executed, output ‘Yes’. Otherwise, output ‘No’.

**Sample Input (blue) and Output (red)**

(1) Test 1

5 3

10 5 7

7 5 3

3 2 2

9 0 2

2 2 2

4 3 3

0 1 0

2 0 0

3 0 2

2 1 1

3 3 2

No

(2) Test 2

5 3

10 5 7

7 5 3

3 2 2

9 0 2

2 2 2

4 3 3

0 1 0

3 2 2

3 0 2

2 1 1

0 0 2

Yes

(3) Test 3

4 5

5 2 6 5 3

1 1 2 1 3

2 2 2 1 0

2 1 3 1 0

1 1 2 2 1

1 0 2 1 1

2 0 1 1 0

1 1 0 1 0

1 1 2 2 1

Yes

(4) Test 4

4 5

5 2 6 5 3

1 1 2 1 3

2 2 2 1 0

2 1 3 1 0

1 1 2 2 1

1 0 2 1 1

2 0 1 1 0

1 1 3 1 0

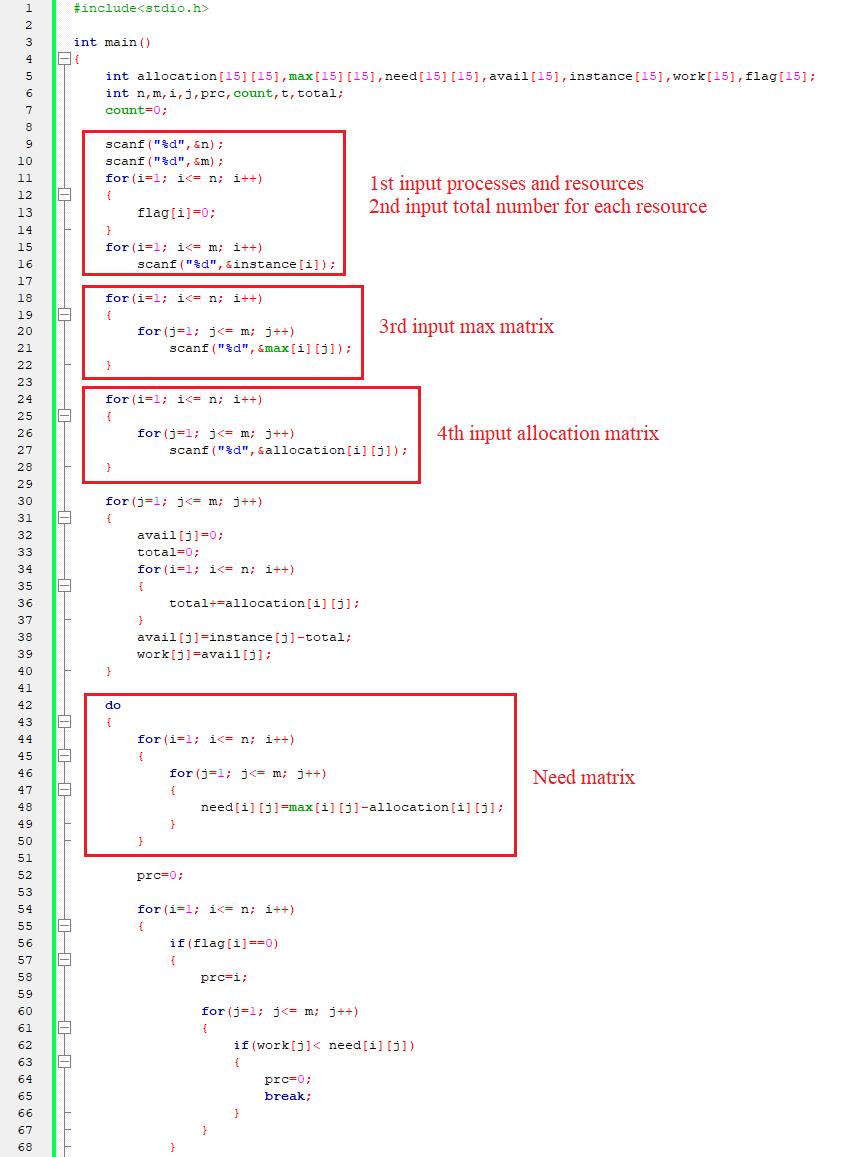
1 1 1 1 0

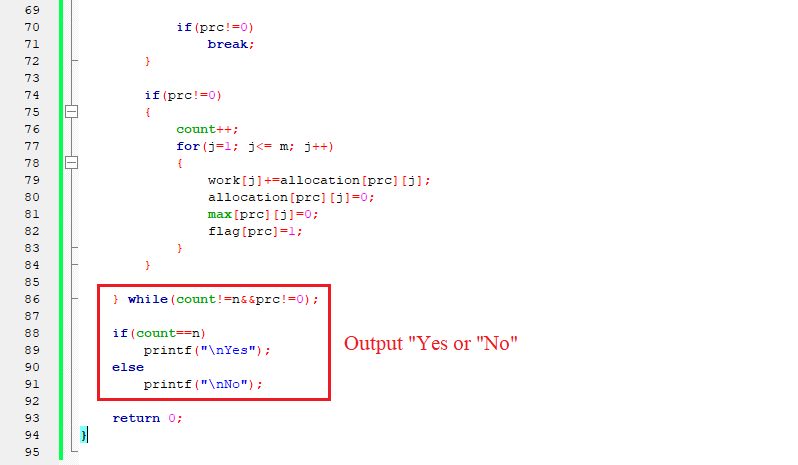
No

**Reference**

Text book: p454-456, ***The Banker’s Algorithm for Multiple Resources***

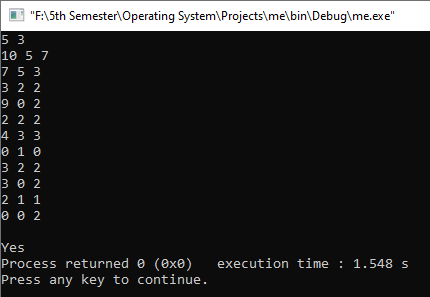
**Program source code**





**Program test**

**1st test:**



**2nd test:**

