Shift Matrix Row-Wise

Once upon a time, there was a group of students who were working on a project to design a gaming platform. They had a **2D grid** of **N * N** size which represented the game board. Each cell of the grid had some data associated with it.

One day, they encountered a problem where they had to shift the elements of the grid **row-wise** in clock wise direction by a certain number of positions, **k**. This was necessary to create a more interesting and challenging gaming experience for their users.

The students decided to write a Java program to solve this problem. They came up with an algorithm to **shift** the elements of the **grid row-wise by k positions**. After implementing the algorithm, they were able to shift the elements of each **row** by **k** positions.

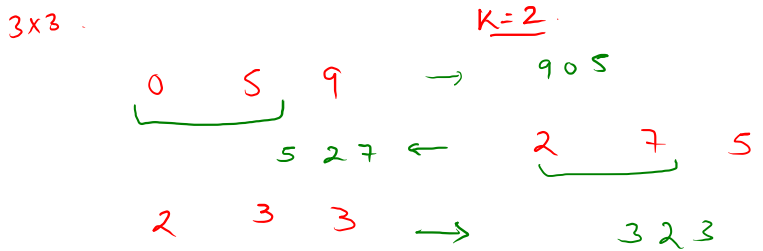
Write a program that shift each row of matrix by **k**.

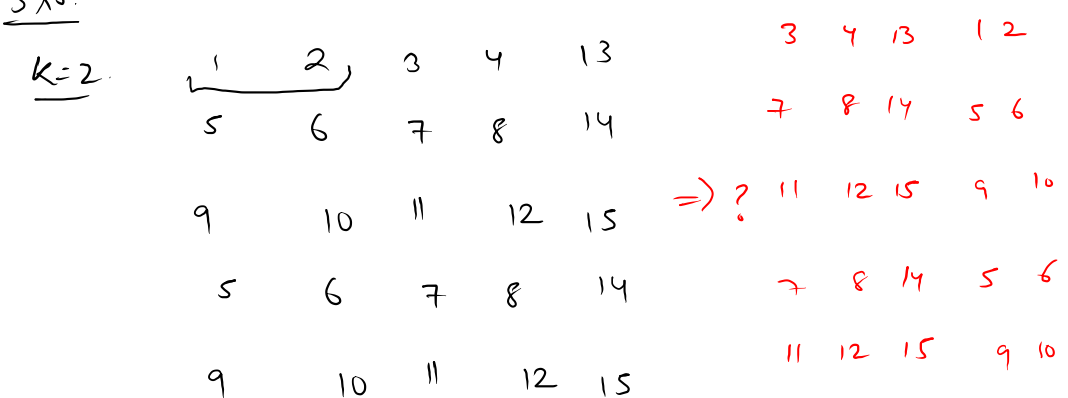
Sample Input 0

```
3
0 5 9
2 7 5
2 3 3
2
```

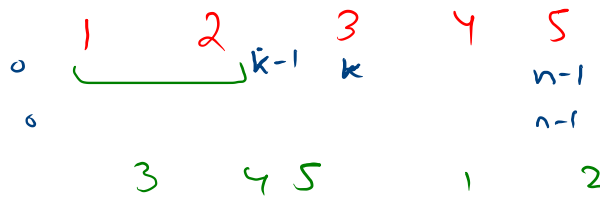
Sample Output 0

```
9 0 5
5 2 7
3 2 3
```





k=2

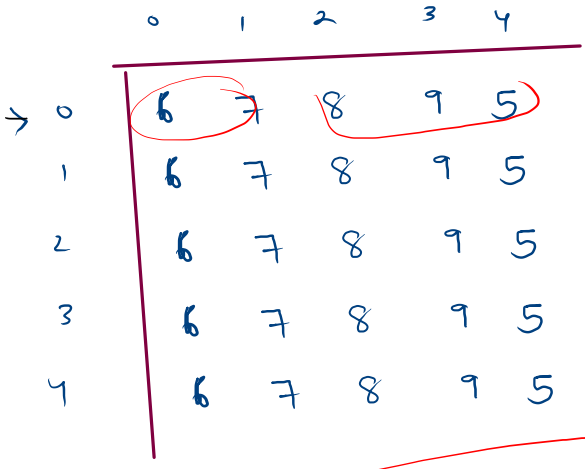


2 1 5 4 3

3 4 5 1 2

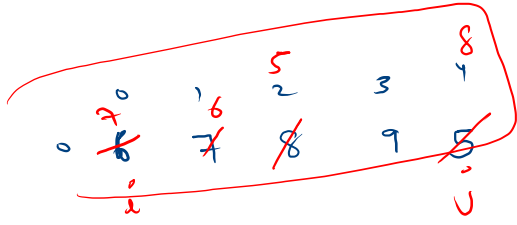
```java
import java.io.*;
import java.util.*;
public class Solution {
    public static void reverse(int [][] A, int i, int j, int row){
        while(i < j){
            int tmp = A[row][i];
            A[row][i] = A[row][j];
            A[row][j] = tmp;
            i++;
            j--;
        }
    }
    public static void rotate(int [][] A, int row, int k){
        int n = A.length;
        reverse(A, 0, k-1, row);
        reverse(A, k, n-1, row);
        reverse(A, 0, n-1, row);
    }
    public static void main(String[] args) {
        Scanner scn = new Scanner(System.in);
        int n = scn.nextInt();
        int [][] A = new int[n][n];
        for(int i = 0; i < n; i++){
            for(int j = 0; j < n; j++){
                A[i][j] = scn.nextInt();
            }
        }
        int k = scn.nextInt();
        for(int row = 0; row < n; row++){
            rotate(A, row, k);
        }
        for(int i = 0; i < n; i++){
            for(int j = 0; j < n; j++){
                System.out.print(A[i][j] + " ");
            }
            System.out.println();
        }

    }
}
```



8 9 5 6 7

K=2



rotate (A, 0, 2)

↳ rev(A, 0, 1, 0)

tmp =

Modify The Matrix

Once upon a time, there was a company that was developing a system to track the inventory levels of different products in different **warehouses**. They had a boolean matrix **Mat** of size **M X N**, where each cell represented the availability of a product in a specific warehouse. If the value of a cell was **true (or 1)**, it meant that the **product** was available in that **warehouse**.

The company wanted to modify the matrix in such a way that if a cell, **Mat[i][j]** was **true**, then all cells in the **ith row** and **jth column** of the matrix would also be set to **true**. This would ensure that if a **product** was available in a particular **warehouse**, all the products in that **row** and **column** would also be considered available.

Can you write a program thet modify the matrix such that if a matrix cell **Mat[i][j]** is **1 (or true)** then make all the cells of **ith row** and **jth column** as **1**.

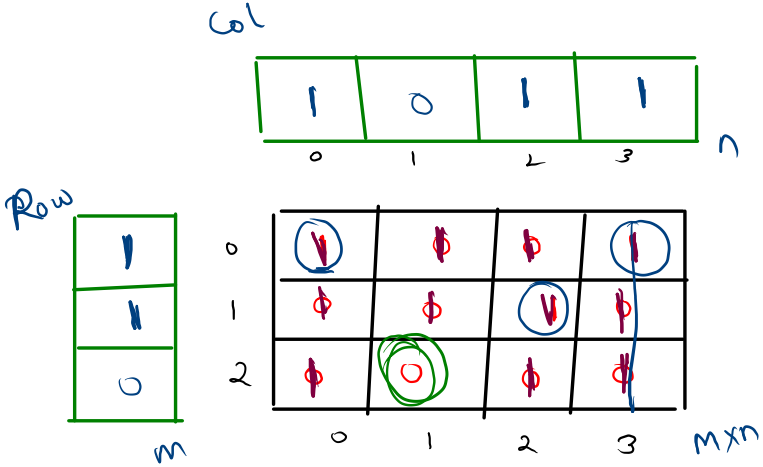
if A[i][j] is 1
it means
make i^{th} row as 1
j^{th} col as 1

Sample Input 0

```
3
4
1 0 0 1
0 0 1 0
0 0 0 0
```

Sample Output 0

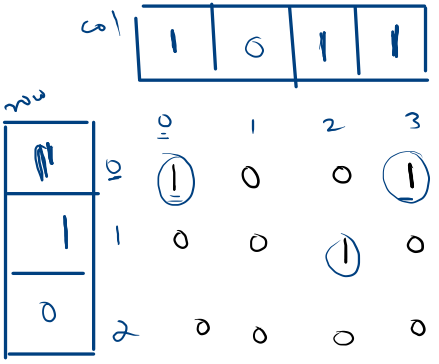
```
1 1 1 1
1 1 1 1
1 0 1 1
```



```java
import java.util.*;
public class Solution {
    public static void main(String[] args) {
        Scanner scn = new Scanner(System.in);
        int m = scn.nextInt();
        int n = scn.nextInt();

        int [][] A = new int[m][n];
        for(int i = 0; i < m; i++){
            for(int j = 0; j < n; j++){
                A[i][j] = scn.nextInt();
            }
        }
        int [] row = new int[m];
        int [] col = new int[n];
        //step 1: mark original 1
        for(int i = 0; i < m; i++){
            for(int j = 0; j < n; j++){
                if(A[i][j] == 1){
                    row[i] = 1;
                    col[j] = 1;
                }
            }
        }
        //step2 : update cells
        for(int i = 0; i < m; i++){
            for(int j = 0; j < n; j++){
                if(row[i] == 1 || col[j] == 1){
                    A[i][j] = 1;
                }
            }
        }
        for(int i = 0; i < m; i++){
            for(int j = 0; j < n; j++){
                System.out.print(A[i][j] + " ");
            }
            System.out.println();
        }
    }
```

```
15          int [] row = new int[m];
16          int [] col = new int[n];
17          //step 1: mark original 1
18          for(int i = 0; i < m; i++){
19              for(int j = 0; j < n; j++){
20                  if(A[i][j] == 1){
21                      row[i] = 1;
22                      col[j] = 1;
23                  }
24              }
25          }
26          //step2 : update cells
27          for(int i = 0; i < m; i++){
28              for(int j = 0; j < n; j++){
29                  if(row[i] == 1 || col[j] == 1){
30                      A[i][j] = 1;
31                  }
32              }
33          }
```



Print Characters

Amy is a high school student who is passionate about **coding**. One day, her computer science teacher gives the class an assignment to **print all the characters of a given string in separate lines.**

Amy immediately gets to work and writes a simple program. However, she feels that her solution is too basic and wants to find a more efficient way to solve the problem.

can you help Amy by writing a program thst print all the characters of a given string in separate lines.

Sample Input 0

```
String
```

Sample Output 0

```
S
t
r
i
n
g
```

Print Indices of Vowels

Maggie is a language enthusiast who loves exploring the intricacies of different languages. One day, while studying English, she comes across a coding challenge that involves printing the **indices of vowels** in a given **string.**

Maggie is determined to solve the challenge and begins working on the problem.

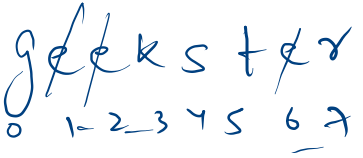
Help Maggie and write a program that prompts the user to input a **string**, and then scans the string for **vowels** while keeping track of the **indices**. Whenever you find a **vowel print the index.**

Sample Input 0

```
aqua
```

Sample Output 0

```
0 2 3
```




```java
import java.io.*;
import java.util.*;

public class Solution {

    public static void main(String[] args) {
        Scanner scn = new Scanner(System.in);
        String s = scn.next();

        for(int i = 0; i < s.length(); i++){
            char ch = s.charAt(i);
            if(ch == 'a' || ch == 'e' || ch == 'i' || ch == 'o' || ch == 'u'){
                System.out.print(i + " ");
            }
        }
    }
}
```

Count Words

Hi → 1

Hi_aman → 2

Sample Input 0

```
Welcome to geekster
```

Sample Output 0

```
3
```

Hi_there_cool ans = 3

every_tree_is_green ans = 4

ch == ' '

ans = 3.

A_b_c_d.