

Array as Hashmap:



freq. array.

8 → abbcabd



a-2

b-3

c-3

d-1

b

0	2	3	1	0	0	0
--------------	--------------	---	---	---	---	---

0 1 2 3 ... 25

a b c d z

2. a-2

b-3

2. int with max freq.



2 1 3 1 4 2 2 2

ans = 2.

1. freq. array.

0	2	4	1	1	1		1
0	1	2	3	4	...	9	

Maximum Freq Character

$s \rightarrow$ a b c c b b b a

a-2
b-4
c-2

\rightarrow (b) ans.

$s \rightarrow$ x y z y z z z \rightarrow (z)

1. freq arr \rightarrow 26.

2. max freq \rightarrow (key)

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

    //freq arr

    int [] freq = new int[26];
    for(int i = 0; i < s.length(); i++){
        char ch = s.charAt(i);
        freq[ch-'a']++;
    }

    // find max freq
    char ans = '#';
    int maxFreq = -1;

    for(int i = 0; i < s.length(); i++){
        char ch = s.charAt(i);
        if(freq[ch-'a'] > maxFreq){
            maxFreq = freq[ch-'a'];
            ans = ch;
        }
    }
    System.out.print(ans);
}
```

```
public class Solution {
    public static boolean goodString(String s){
        //freq arr
        int [] freq = new int[26];
        for(int i = 0; i < s.length(); i++){
            char ch = s.charAt(i);
            freq[ch-'a']++;
        }

        //compare freq
        int initFreq = freq[s.charAt(0) - 'a']; //freq of first char of string

        for(int i = 0; i < s.length(); i++){
            char ch = s.charAt(i);
            int currFreq = freq[ch-'a'];

            if(currFreq != 0 && currFreq != initFreq){
                return false;
            }
        }
        return true;
    }
}
```

true / false. } → use function.

freq. arr \rightarrow

2	2	2	0	0	...	25
---	---	---	---	---	-----	----

8 → $\begin{pmatrix} d & c & b & b & c & d \\ 0 & 1 & 2 & 3 & 4 & 5 \end{pmatrix}$

0	2	2	2	0	0	0
0	1	2	3	... 25		
a	b	c	d	... z		

initf = 2

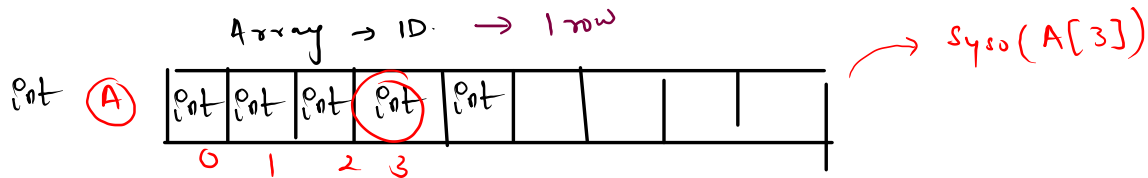
$i = 1, 2, 3, 4$

$$\underline{8} < 6 \quad 2 < 6$$
$$CF = \hat{z}$$
$$ch = b \checkmark$$

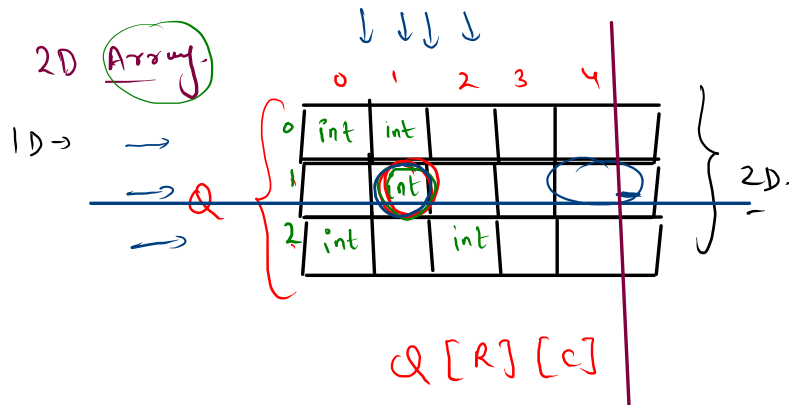
$CF = \underline{(2)}$
 $3 < 6$.

 $4 < 6$

2D Array \rightarrow Matrix.

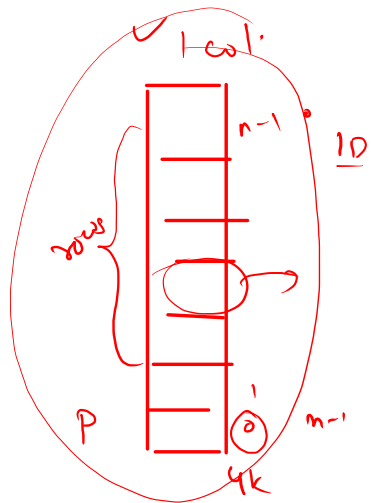


each cell of my Arr can store int value.



$Q[1][1]$

$Q[1][4]$



$P[3]$

$$4k + 4 \times 3$$

$$4k + 12$$

$$= 4012$$

Batch →

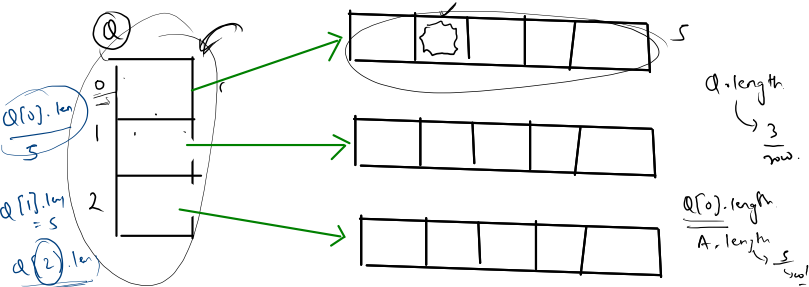
↓ Marks

Q

	0	1			
0	25	10			
1	17	12			
2	14				

Q.length → total row.

Q[0].length → total column.



At each cell of 'Q' array we are storing 1D array of type int.

Q[0][1]

	0	1	2	3	4
0					
1					
2					

3x5

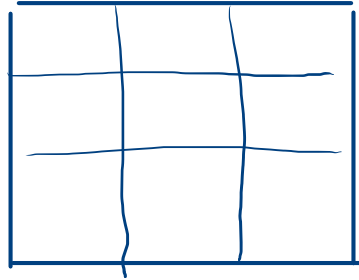
```
*****  
public class Main  
{  
    public static void main(String[] args) {  
        //      int [] A = new int[1];  
        //      System.out.println(A.length);  
  
        int[][] Q = new int[3][5];  
  
        int row = Q.length;  
        int col = Q[3].length;  
  
        System.out.println(row + " " + col);  
    }  
}
```

Print the Matrix Row-wise

$m \rightarrow r$
 $n \rightarrow c$

Sample Input 0

m n
3 3
0 1 2 3
1 4 5 6
2 7 8 9



Sample Output 0

1 2 3
4 5 6
7 8 9

```
public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);
    int m = scn.nextInt();    //row
    int n = scn.nextInt();    //col

    int [][] Q = new int[m][n];

    //input
    for(int i = 0; i < m; i++){
        for(int j = 0; j < n; j++){
            Q[i][j] = scn.nextInt();
        }
    }

    //output
    for(int i = 0; i < m; i++){
        for(int j = 0; j < n; j++){
            System.out.print(Q[i][j] + " ");
        }
        System.out.println();
    }
}
```

Print alternate row.

```
public static void main(String[] args) {
    Scanner scn = new Scanner(System.in);
    int m = scn.nextInt();    //row
    int n = scn.nextInt();    //col

    int [][] Q = new int[m][n];

    //input
    for(int i = 0; i < m; i++){
        for(int j = 0; j < n; j++){
            Q[i][j] = scn.nextInt();
        }
    }

    //output
    // for(int i = 0; i < m; i += 2){
    //     for(int j = 0; j < n; j++){
    //         System.out.print(Q[i][j] + " ");
    //     }
    //     System.out.println();
    // }

    for(int i = 0; i < m; i++){
        if(i % 2 == 0){
            for(int j = 0; j < n; j++){
                System.out.print(Q[i][j] + " ");
            }
            System.out.println();
        }
    }
}
```


Print the matrix left-diagonal wise

Sample Input 0

```
3
1 2 3
4 5 6
7 8 9
```

Sample Output 0

```
1 2 4 3 5 7 6 8 9
```

