

Previous

PSA

1	2	3	4	5
0	1	2	3	4
1	3	6	10	15

$$arr[i] + psa[i-1]$$

1. Greatest till me.

2	4	1	5	3
0	1	2	3	4
2	4	4	5	5

$$\max(psa[i-1], arr[i])$$

2. Print range b/w L and R.

n=5

4	2	1	3	5
0	1	2	3	4

L=1  
R=3

4	6	7	10	15
---	---	---	----	----

3. Print freq of alphabet

a b c c c b a d p

1. Making arr of size

26

a-2
b-2
c-3
d-1
p-1

a-2
b-2
c-3
d-1
p-1

4. a b a c c c c b

a-1
b-2
c-4

freq ↑

(c)

maxf = 4  
mfc = 'c'

3 > maxf

4 > maxf

4 > maxf

2 > maxf

2 > 4

# Int with Maximum Freq

Given an array that contains only one-digit integers from 0-9. Print the digit which occurs in the array the maximum number of times. You must take a constant extra space and also you have to complete the question in the single traversal.

eg.  $\{ \overset{\checkmark}{4} \overset{\checkmark}{5} \overset{\checkmark}{7} \overset{\checkmark}{5} 6 2 5 2 \}$   
 0 1 2 3 4 5 6 7

1.  $f \rightarrow$  array  $\Rightarrow$   $\text{arr} \Rightarrow 10$

int 

0	1	2	3	4	5	6	7	8	9

$\text{maxf} = 10$   
 $\text{maxv} = \text{arr}[0]$   
 $\text{arr}[0] = 10$   
 $\text{arr}[1] = 5$

$2 \times \text{mf} \checkmark$   
 $1 > \text{mf}$   
 $2 > \text{mf}$

Ques. 3 1 2 4 3 5 7 3 3  
3

a b c c c c d  
c?

```
int [] freq = new int[10];
int maxFreq = 1;
int maxVal = arr[0];
freq[arr[0]] = 1;

for(int i = 1; i < n; i++){
    freq[arr[i]]++;
    if(freq[arr[i]] > maxFreq){
        maxVal = arr[i];
        maxFreq = freq[arr[i]];
    }
}
System.out.println(maxVal);
```

$\text{freq} \rightarrow \text{size} \rightarrow 10$   
 $\text{arr} = [0, 9]$

$\text{maxf} = 10$   
 $\text{maxv} = \text{arr}[0] = 10$   
 $\text{arr}[1] = 5$

$f[0] > m$   
 $f[1] > m$   
 $f[2] > m$

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

public class Solution {
    public static void main(String[] args) {
        Scanner sc = new Scanner(System.in);
        int n = sc.nextInt();
        int [] arr = new int[n];

        for(int i = 0; i < n; i++){
            arr[i] = sc.nextInt();
        }

        int [] freq = new int[10];
        int maxFreq = 1;
        int maxVal = arr[0];
        freq[arr[0]] = 1;

        for(int i = 1; i < n; i++){
            freq[arr[i]]++;
            if(freq[arr[i]] > maxFreq){
                maxVal = arr[i];
                maxFreq = freq[arr[i]];
            }
        }
        System.out.println(maxVal);
    }
}
```

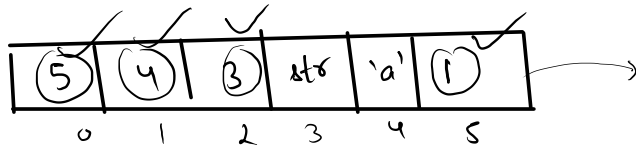
eg. 2 4 4 4 9 4 7 4  
4

1 2 3 1 2 1 1  
4

0 4 5 7 3 9 3  
4

int

arr



n=6  
→ size.

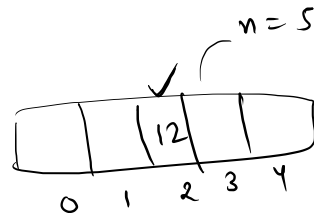
2D → matrix.

1D → n.

arr

	0	1	2	3
0				
1				15
2				
3		14		
4				

5 1D array



arr [ 1 ] [ 3 ] = 15; n x m  
5 x 4.  
Syso (arr [ 1 ] [ 3 ] );

arr [ 2 ] = 12;

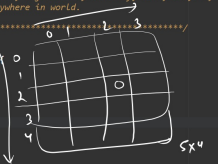
arr [ 3 ] [ 1 ] = 14;

# 2D - Arrays (Matrices)

```
Welcome to GDB Online.
GDB online is an online compiler and debugger tool for C, C++, Python, Java, PHP, Ruby,
C#, OCaml, VB, Swift, Pascal, Fortran, Haskell, Objective-C, Assembly, HTML, CSS, JS,
Code, Compile, Run and Debug online from anywhere in world.

=====
public class Main
{
    public static void main(String[] args) {
        int n = 5;
        int m = 4;

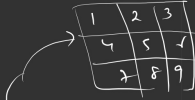
        // int [] arr = new int[n];
        int [][] A = new int[n][m];
    }
}
```



```
public class Main
{
    public static void main(String[] args) {
        int n = 5;
        int m = 4;

        // int [] arr = new int[n];
        // int [] arr = {1,2,3,4,5};
        int [][] A = new int[n][m];

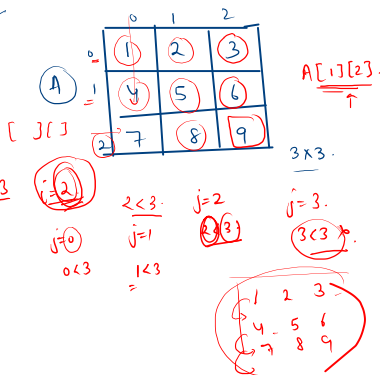
        int [][] A = { {1,2,3}, {4,5,6}, {7,8,9} };
    }
}
```



```
int [][] A = { {1,2,3}, {4,5,6}, {7,8,9} };

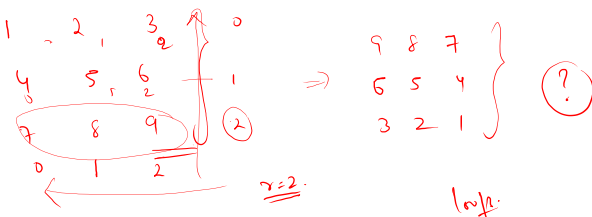
for(int i = 0; i < 3; i++){
    for(int j = 0; j < 3; j++){
        System.out.print(A[i][j] + " ");
    }
    System.out.println();
}
```

3 < 3



$i=0$   
 $j=0$   $0 < 3$   
 $j=1$   $1 < 3$   
 $j=2$   $2 < 3$   
 $j=3$   $3 < 3$  ✓

$i=3$   
 $j=0$   $0 < 3$   
 $j=1$   $1 < 3$   
 $j=2$   $2 < 3$   
 $j=3$   $3 < 3$  ✗



# Print Alternate Row

You are given a 2D matrix, your task is print the **alternate rows** of the matrix starting from the **0th** row.

## Sample Input 0

rows  
4  
cols  
6

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

## Sample Output 0

2	3	8	7	0	4
0	0	8	1	0	8

0 1 2 3 4 5

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

i  
i+2

4x6

A

?

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

public class Solution {

    public static void main(String[] args) {
        Scanner scn = new Scanner(System.in);
        int n = scn.nextInt(); //4
        int m = scn.nextInt(); //6

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

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

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

```
public class Solution {

    public static void main(String[] args) {
        Scanner scn = new Scanner(System.in);
        int n = scn.nextInt(); //4
        int m = scn.nextInt(); //6

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

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

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

yes

$O(n^2)$