DAA ASSIGNMENT

N.Tejaswini - 21071A67B0

1 .Given a row wise sorted matrix of size **R*C** where R and C are always **odd**, find the median of the matrix.

#using python

Test Case 1:

```
Input:
R = 3, C = 3
M = [[1, 3, 5],
     [2, 6, 9],
     [3, 6, 9]]
Output: 5
```

```
matrix=[[1,3,5],[2,6,9],[3,6,9]]

arr= []


for i in range(3):

   for j in range(3):

      arr.append(matrix[i][j])


arr.sort()

print("Hence,",arr[4],"is median.")
```

Output:

```
Hence, 5 is median.
>
```

Test Case 2:

Input:

R = 3, C = 1

M = [[1], [2], [3]]

Output: 2

```
matrix=[[1],[2],[3]]
arr= []

for i in range(3):
   for j in range(1):
      arr.append(matrix[i][j])

arr.sort()
print("Hence,",arr[1],"is median.")
```

Output:

```
Hence, 5 is median.
>
```

2. Given the arrival and departure times of all trains that reach a railway station, the task is to find the minimum number of platforms required for the railway station so that no train waits. We are given two arrays that represent the arrival and departure times of trains that stop.

Test Case 1:

Input: *arr[] = {9:00, 9:40, 9:50, 11:00, 15:00, 18:00}, dep[] = {9:10, 12:00, 11:20, 11:30, 19:00, 20:00}*
Output: *3*

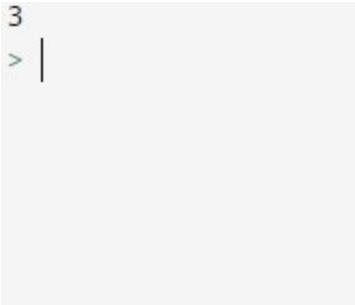
```
def platform(arr,dep,n):
    arr.sort()
    dep.sort()
    plat_needed=1
    res=1
    i=1
    j=0
    while (i<n and j<n):
        if (arr[i]<=dep[j]):
            plat_needed+=1
            i+=1
        elif (arr[i]>dep[j]):
```

```
            plat_needed-=1
            j+=1

        if (plat_needed>res):
            res=plat_needed
    return res

arr=[900,940,950,1100,1500,1800]
dep=[910,1200,1120,1130,1900]
n=len(arr)
print(platform(arr,dep,n))
```

Output:

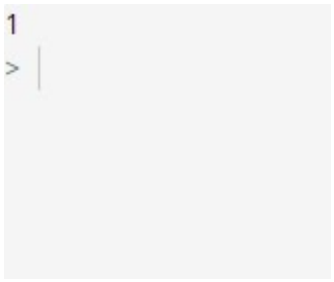


Test Case 2:

Input: *arr[] = {9:00, 9:40}, dep[] = {9:10, 12:00}*
Output: *1*

```
arr=[900,940]
dep=[910,1200]
```

Output:


1
>