DP-S006 (E024)

Solved Challenges 1/2

Back To Challenges List





Minimum Sum - K*K Sub-Matrix

ID:11114 **Solved By 719 Users**

The program must accept an integer matrix of size **R*C** and an integer **K** as the input. The program must print the sum of integers in the K*K sub-matrix which has the minimum sum S among the all possible K*K submatrices of the given R*C matrix as the output.

Boundary Condition(s):

2 <= R, C <= 1000

2 <= K <= R and C

Input Format:

The first line contains R and C separated by a space.

The next R lines, each containing C integers separated by a space.

The $(R+2)^{nd}$ line contains K.

Output Format:

The first line contains S.

Example Input/Output 1:

Input:

5 4

8497

4052

3596

3004

8861

3

Output:

29

Explanation:

The **3*3** sub-matrix which has the minimum sum is given below.

405

359

300

Example Input/Output 2:

Input:

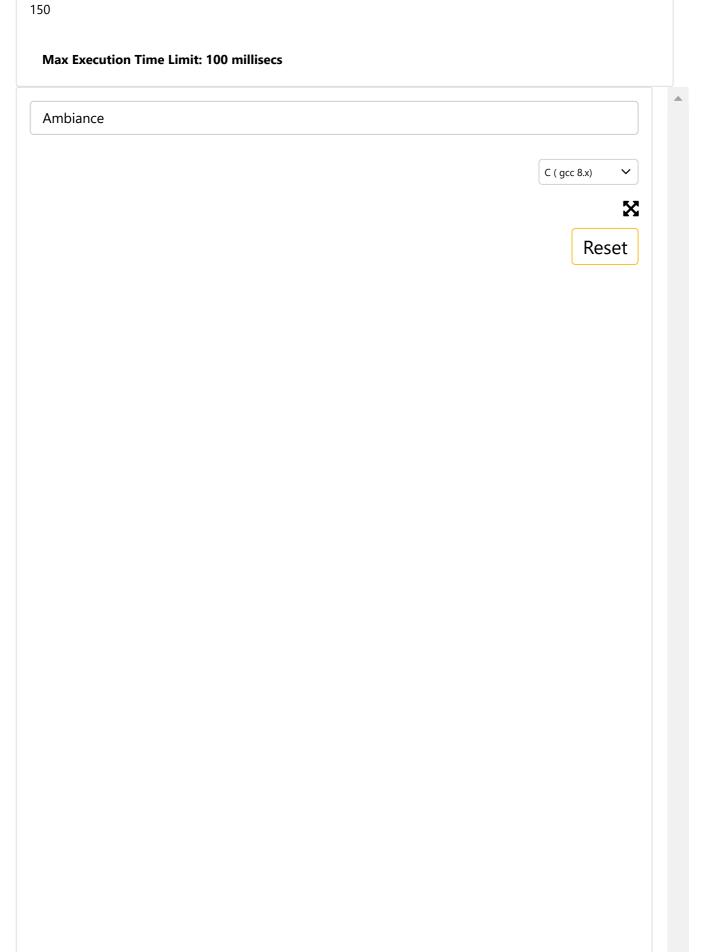
44

10 80 50 70

40 30 50 50

50 70 30 20 70 10 40 70 2

Output:



```
1 #include<stdio.h>
 2 #include<stdlib.h>
 3 #include<limits.h>
 4 int main()
 5
   {
 6
        int R,C;
 7
        scanf("%d%d",&R,&C);
 8
        int sum matrix[R][C+1];
 9
10
        for(int row=0;row<R;row++)</pre>
11
             for(int col=0;col<C+1;col++)</pre>
                  sum matrix[row][col] = 0;
12
13
14
        for(int row=0;row<R;row++)</pre>
15
             for(int col = 1;col<C+1;col++)</pre>
16
17
             {
18
                  int curr;
19
                  scanf("%d",&curr);
20
                  sum_matrix[row][col] = curr + sum_matrix[row][col-
21
             }
22
        }
23
24
        int k,min_sum;
        scanf("%d",&k);
25
26
        min sum = INT MAX;
27
28
        for(int row=0;row<=R-k;row++)</pre>
29
        {
             for(int col=1;col<=C-k+1;col++)</pre>
30
31
32
                  int sum=0;
33
                  for(int srow=row;srow<row+k;srow++)</pre>
34
35
                      sum += sum_matrix[srow][col+k-1] - sum_matrix[
36
                  if(sum<min sum)</pre>
37
38
                      min_sum = sum;
39
             }
40
        printf("%d", min_sum);
41
42
    }
```

Save

Run

Run with a custom test case (Input/Output)