

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 sub-matrices 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
8 4 9 7
4 0 5 2
3 5 9 6
3 0 0 4
8 8 6 1
3
```

Output:

```
29
```

Explanation:

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

```
4 0 5
3 5 9
3 0 0
```

Example Input/Output 2:

Input:

```
4 4
10 80 50 70
40 30 50 50
```

50 70 30 20
70 10 40 70
2

Output:
150

Max Execution Time Limit: 100 millisecs

Ambiance

C (gcc 8.x) 



Reset

```
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++)
11         for(int col=0;col<C+1;col++)
12             sum_matrix[row][col] = 0;
13
14     for(int row=0;row<R;row++)
15     {
16         for(int col = 1;col<C+1;col++)
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;
25     scanf("%d",&k);
26     min_sum = INT_MAX;
27
28     for(int row=0;row<=R-k;row++)
29     {
30         for(int col=1;col<=C-k+1;col++)
31         {
32             int sum=0;
33             for(int srow=row;srow<row+k;srow++)
34             {
35                 sum += sum_matrix[srow][col+k-1] - sum_matrix[
36             }
37             if(sum<min_sum)
38                 min_sum = sum;
39         }
40     }
41     printf("%d", min_sum);
42 }
```

SaveRun☐ Run with a custom test case (Input/Output)