

ASK/VIEW DOUBT

SOLUTION

Problem

Result

Code : Min Cost Path

Send Feedback

Given an integer matrix of size m*n, you need to find out the value of minimum cost to reach from the cell (0, 0) to (m-1, n-1).

From a cell (i, j), you can move in three directions : (i+1, j), (i, j+1) and (i+1, j+1).

Cost of a path is defined as the sum of values of each cell through which path passes.

Input Format :

Line 1 : Two integers, m and n
Next m lines : n integers of each row (separated by space)

Output Format :

Minimum cost

Constraints :
1 <= m, n <= 20

Sample Input 1 :

3 4
3 4 1 2
2 1 8 9
4 7 8 1

Sample Output 1 :

13

1#include <iostream>
2#include <vector>
3#include <climits>
4using namespace std;
5
6int minCostPath(int **input, int m, int n) {
7/* Don't write main().
8 * Don't read input, it is passed as function argument.
9 * Return output and don't print it.
10 * Taking input and printing output is handled automatically.
11 */
12
13int **ans = new int*[n];
14for(int i=0;i<n;i++){
15ans[i] = new int[m];
16}
17
18ans[m-1][n-1] = input[m-1][n-1];
19
20for(int i=m-1;i>=0;i--){
21for(int j=n-1;j>=0;j--){
22
23if(i==m-1 && j==n-1)
24continue;
25if(i==m-1){
26
27ans[i][j] = ans[i][j+1]+input[i][j];
28
29}
30else if(j==n-1){
31ans[i][j] = ans[i+1][j]+input[i][j];
32
33}
34else{
35ans[i][j] = min(ans[i+1][j+1],min(ans[i+1][j],ans[i][j+1]))+input[i][j];
36
37}
38
39}

< PREVIOUS

> NEXT

CUSTOM INPUT

SUBMIT SOLUTION