

[All Contests](#) > [In21-CS2023-Lab2](#) > 2D Array - DS

2D Array - DS

Problem

Submissions

Leaderboard

Discussions

Submitted a few seconds ago • Score: 35.00

Status: **Accepted**

✓	Test Case #0	✓	Test Case #1	✓	Test Case #2
✓	Test Case #3	✓	Test Case #4	✓	Test Case #5
✓	Test Case #6	✓	Test Case #7	✓	Test Case #8

Submitted Code

Language: C++

[🔗 Open in editor](#)

```
1 #include <bits/stdc++.h>
2
3 using namespace std;
4
5 string ltrim(const string &);
6 string rtrim(const string &);
7 vector<string> split(const string &);
8
9 /*
10  * Complete the 'hourglassSum' function below.
11  *
12  * The function is expected to return an INTEGER.
13  * The function accepts 2D_INTEGER_ARRAY arr as parameter.
14  */
15
16 int hourglassSum(vector<vector<int>> arr) {
17     vector<vector<int>> sumArr(4); // initializing 2D vector sumArr of length 4
18     for (int i=0; i<4; i++){ // outer loop
19         sumArr[i].resize(4); // resizing each element of sumArr to size 4
20         for (int j=0; j<4; j++){ // inner loop
21             int sum = arr[i][j]+arr[i][j+1]+arr[i][j+2]+arr[i+1][j+1]+arr[i+2][j]+arr[i+2]
22 [j+1]+arr[i+2][j+2]; // getting the sum of hourglass elements
23             sumArr[i][j] = sum; // assigning elements to sumArr
24         }
25     }
26     int max = sumArr[0][0]; // initializing a maximum value
27     for (int i=0; i<4; i++){ // outer loop
28         for (int j=0; j<4; j++){ // inner loop
29             if(sumArr[i][j]>max){ // checking if the element of sumArr is bigger than max
30                 max = sumArr[i][j]; // if yes assigning the new value to max
31             }
32         }
33     }
34 }
```

```
33     return max;
34 }
35
36 int main()
37 {
38     ofstream fout(getenv("OUTPUT_PATH"));
39
40     vector<vector<int>> arr(6);
41
42     for (int i = 0; i < 6; i++) {
43         arr[i].resize(6);
44
45         string arr_row_temp_temp;
46         getline(cin, arr_row_temp_temp);
47
48         vector<string> arr_row_temp = split(rtrim(arr_row_temp_temp));
49
50         for (int j = 0; j < 6; j++) {
51             int arr_row_item = stoi(arr_row_temp[j]);
52
53             arr[i][j] = arr_row_item;
54         }
55     }
56
57     int result = hourglassSum(arr);
58
59     fout << result << "\n";
60
61     fout.close();
62
63     return 0;
64 }
65
66 string ltrim(const string &str) {
67     string s(str);
68
69     s.erase(
70         s.begin(),
71         find_if(s.begin(), s.end(), not1(ptr_fun<int, int>(isspace)))
72     );
73
74     return s;
75 }
76
77 string rtrim(const string &str) {
78     string s(str);
79
80     s.erase(
81         find_if(s.rbegin(), s.rend(), not1(ptr_fun<int, int>(isspace))).base(),
82         s.end()
83     );
84
85     return s;
86 }
87
88 vector<string> split(const string &str) {
89     vector<string> tokens;
90
91     string::size_type start = 0;
92     string::size_type end = 0;
93
94     while ((end = str.find(" ", start)) != string::npos) {
95         tokens.push_back(str.substr(start, end - start));
96
97         start = end + 1;
98     }
```

```
99  
100     tokens.push_back(str.substr(start));  
101  
102     return tokens;  
103 }  
104
```

[Interview Prep](#) | [Blog](#) | [Scoring](#) | [Environment](#) | [FAQ](#) | [About Us](#) | [Support](#) | [Careers](#) | [Terms Of Service](#) | [Privacy Policy](#) |