Diagonal Difference ☆

Given a square matrix, calculate the absolute difference between the sums of its diagonals.

For example, the square matrix arr is shown below:

```
1 2 3
4 5 6
9 8 9
```

The left-to-right diagonal = $1 + 5 + 9 = 15$. The right to left diagonal = $3 + 5 + 9 = 17$. Their absolute difference is $|15 - 17| = 2$.

Function description

Complete the *diagonalDifference* function in the editor below. It must return an integer representing the absolute diagonal difference.

diagonalDifference takes the following parameter:

- arr: an array of integers .

Input Format

The first line contains a single integer, n, the number of rows and columns in the matrix arr. Each of the next n lines describes a row, $arr[i]$, and consists of n space-separated integers $arr[i][j]$.

Constraints

- $-100 \leq arr[i][j] \leq 100$

Output Format

Print the absolute difference between the sums of the matrix's two diagonals as a single integer.

Sample Input

```
3
11 2 4
4 5 6
10 8 -12
```

Sample Output

```
15
```

Explanation

The primary diagonal is:

```
11
   5
     -12
```

Sum across the primary diagonal: 11 + 5 - 12 = 4

Author vatsalchanana
Difficulty Easy
Max Score 10
Submitted By 475582

NEED HELP?

View discussions
View editorial
View top submissions

RATE THIS CHALLENGE

☆ ☆ ☆ ☆ ☆

MORE DETAILS

Download problem statement
Download sample test cases
Suggest Edits

The secondary diagonal is:

```
    4
  5
10
```

Sum across the secondary diagonal: 4 + 5 + 10 = 19

Difference: |4 - 19| = 15

Note: |x| is the absolute value of x

Current Buffer (saved locally, editable) C++

```cpp
#include <bits/stdc++.h>

using namespace std;

// Complete the diagonalDifference function below.
int diagonalDifference(vector<vector<int>> arr) {


}

int main()
{
    ofstream fout(getenv("OUTPUT_PATH"));

    int n;
    cin >> n;
    cin.ignore(numeric_limits<streamsize>::max(), '\n');

    vector<vector<int>> arr(n);
    for (int i = 0; i < n; i++) {
        arr[i].resize(n);

        for (int j = 0; j < n; j++) {
            cin >> arr[i][j];
        }

        cin.ignore(numeric_limits<streamsize>::max(), '\n');
    }

    int result = diagonalDifference(arr);

    fout << result << "\n";

    fout.close();

    return 0;
}

```

Line: 1 Col: 1

Upload Code as File Test against custom input Run Code Submit Code