Zero Sum Cells

locked

Problem | Submissions | Leaderboard | Discussions

Given a 2D matrix of integers of size N*M, it is required to find the maximum number of cells that sum to 0 and do not have any duplicates. Solve this problem using a recursive brute force solution.

Note:

There is no specific set of movement directions for that problem, because any subset of cells is allowed. This is a simple include-exclude problem.

Input Format

- one line containing N and M, space separated
- the next N lines, each line contains M space separated numbers

Constraints

- N is from 1 to 4
- M is from 1 to 4
- cell values are from -10^9 to 10^9

Output Format

- one line containing the maximum number of cells that sum to 0 and do not have any duplicates

Sample Input 0

```
3 3
3 -3 4
2 2 0
7 -2 -2
```

Sample Output 0

```
5
```

Explanation 0

The cells are: 3, -3, 2, 0, -2. Note that we cannot add the other 2, -2 because the chosen cells will be containing duplicates.

Sample Input 1

```
2 3
-3 0 3
2 1 0
```

Sample Output 1

```
4
```

Explanation 1

The cells: -3, 0, 2, 1. Note that -3, 0, 3 sum to zero but the length is 3 (less than 4). Note also that we cannot add the other 0 because of the duplicates condition.

Submissions: 36
Max Score: 10
Difficulty: Medium

Rate This Challenge:
☆☆☆☆☆

More

C++20

```
#include <cmath>
#include <cstdio>
#include <vector>
#include <iostream>
#include <algorithm>
using namespace std;


int main() {
    /* Enter your code here. Read input from STDIN. Print output to STDOUT */
    return 0;
}

```

Line: 1 Col: 1

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