

2019 Online Code Jam 1 - Editorial

1st LGE Code Jam 2019 - Problem A

1st LGE Code Jam 2019 - Problem A (English)

1st LGE Code Jam 2019 - Problem B

1st LGE Code Jam 2019 - Problem B (English)

1st LGE Code Jam 2019 - Problem C

1st LGE Code Jam 2019 - Problem C (English)

1st LGE Code Jam 2019 - Problem D

1st LGE Code Jam 2019 - Problem D (English)

1st LGE Code Jam 2019 - Problem E

1st LGE Code Jam 2019 - Problem E (English)

1st LGE Code Jam 2019 - Problem F

1st LGE Code Jam 2019 - Problem F (English)

2019 Online Code Jam 2 - Editorial

코드잼 준비하기

2019년 활동

2018년 활동

2016년 활동

2015년 활동

2014년 활동

2013년 활동

Space tools

1st LGE Code Jam 2019 - Problem B (English)

Created by 허지영 jiyoung.huh, last modified by 박은영 eunyoung.park on 2019/05/14

[Array Manipulation]

If you want to solve this problem with the Brute Force algorithm to sequentially process all cases for a given condition, the time complexity of $O(MN^2)$ is required to process M operations on all the elements of the two-dimensional array of N x N.

In this problem, since the sizes of N and M are $1 \leq N, M \leq 1000$ and the time complexity of the algorithm is $O(MN^2) = 10^9$, the total execution time for test case $T \leq 10$ is $O(TMN^2) = 10^{10}$ and the problem can not be solved within a given time (2 seconds).

However, the output required by the problem does not print all the elements of the two-dimensional array after M operations, but prints only the sum of each row and column.

If you think a bit more, instead of applying M operations to elements of rows from r1 to r2 and columns from c1 ~ c2,we can consider an algorithm that updates only the sum of each row and column using two one-dimensional arrays.

In this case, the time complexity is $O(MN)$ because it computes only the sum of rows from r1 to r2 and columns from c1 ~ c2.

Total execution time for test case $T \leq 10$ is $O(TMN) = 10^7$ and the problem can be solved within the time (2 seconds).

Problem B

```
1  #include <stdio>
2  #include <stdlib>
3  #include <string>
4
5  using namespace std;
6
7  #define MAX_N    1001
8  int N, M;
9  int R[MAX_N];
10 int C[MAX_N];
11
12 void solve()
13 {
14     int x, v;
15     int r1, r2, c1, c2;
16
17     for(int i=0; i<M; i++) {
18         scanf("%d %d %d %d %d", &r1, &c1, &r2, &c2, &v);
19         r1--; r2--; c1--; c2--;
20         x = (r2-r1+1) * v;
21         for(int j=c1; j<=c2; j++) C[j] += x;
22         x = (c2-c1+1) * v;
23         for(int j=r1; j<=r2; j++) R[j] += x;
24     }
25
26     for(int i=0; i<N; i++) printf("%d ", R[i]); printf("\n");
27     for(int i=0; i<N; i++) printf("%d ", C[i]); printf("\n");
28 }
29
30 int main(int argc, char* argv[])
31 {
32     int T;
33     int x;
34
35     scanf("%d", &T);
36     while(T-->0) {
37         memset(R, 0, sizeof(R));
38         memset(C, 0, sizeof(C));
39         scanf("%d %d", &N, &M);
40         for(int i=0; i<N; i++) {
41             for(int j=0; j<N; j++) {
42                 scanf("%d", &x);
43                 R[i] += x;
44                 C[j] += x;
45             }
46         }
47         solve();
48     }
49
50
51     return 0;
52 }
```

Like

Be the first to like this

No labels

Write a comment...