

ВШЭ АиСД 2021. Массивы, матрицы

4 дек 2021, 19:02:55

старт: 16 сен 2021, 11:30:00

финиш: 26 сен 2021, 23:59:59

длительность: 10д. 12ч.

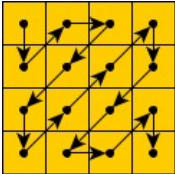
начало: 16 сен 2021, 11:30:00

конец: 26 сен 2021, 23:59:59

Е. Спиральное заполнение матрицы (0.2)

Ограничение времени	0.1 секунд
Ограничение памяти	64.0 Мб
Ввод	стандартный ввод
Вывод	стандартный вывод

Требуется заполнить квадратную матрицу размерности $N \times N$ по следующей схеме (заполнение начинается из **левого верхнего угла**):



Заполнение выполняется арифметической прогрессией в соответствии с заданным начальным значением a_1 и разностью d .

Заполнение и вывод полученной матрицы **должны быть реализованы в виде отдельных функций**.

Формат ввода

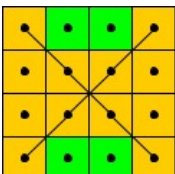
В первой строке – число $N \leq 100$, которое определяет размерность квадратной матрицы.

Во второй строке – два числа a_1 (начальное значение, $|a_1| \leq 10^5$) и d (разность, $|d| \leq 10^4$) арифметической прогрессии для заполнения матрицы.

Формат вывода

В первой строке – сумма элементов матрицы, которые находятся в верхнем и нижнем треугольниках, образованных главной и побочной диагоналями полученной матрицы **БЕЗ** учета элементов, которые стоят на этих диагоналях. Использовать вложенные циклы для подсчета суммы **не разрешается**.

Например, в случае матрицы 4×4 , необходимо просуммировать следующие элементы:



В последующих N строках, если размерность матрицы N не превышает 20 включительно, выводится содержимое полученной матрицы (построчно, через пробел).

Пример

Ввод Вывод

Ввод 4
1 1Вывод 34
1 3 4 10
2 5 9 11
6 8 12 15
7 13 14 16

Язык GNU c++17 7.3

Набрать здесь

Отправить файл

```
1 /*Е. Спиральное заполнение матрицы*/
2
3 #include <iostream>
4 #include <vector>
5
6 int64_t FindSumInTriangle(const std::vector<std::vector<int64_t>>& matrix) {
7     int64_t counter = 0;
8     int counterLines = 0;
9
10    bool restart = false;
11
12    for (unsigned int j = 0, i = 0; ; ++j) {
13        if (j + 1 == matrix.size()) {
14            ++counterLines;
15            restart = true;
16        }
17
18        if (counterLines == matrix.size()) {
19            break;
20        }
21
22        if (j > i && j < matrix.size() - i - 1) {
23            counter += matrix[i][j];
24        }
25
26        if (i > j && j > matrix.size() - i - 1) {
27            counter += matrix[i][j];
28        }
29
30        if (restart) {
31            j = -1;
32            ++i;
33            restart = false;
34        }
35    }
36
37    return counter;
38 }
```

Отправить

Предыдущая

Следующая