**Структуры данных. Алгоритмы обработки структур данных**

Задание 1. В каждой задаче задается квадратная целочисленная матрица NxN (значение N вводится с клавиатуры, N<10). Программа должна заполнять матрицу случайными числами из диапазона [a, b] (a, b вводятся с клавиатуры) и осуществлять вывод на экран исходной матрицы. Затем необходимо произвести необходимые действия и напечатать результаты. Вычислить произведение чисел, принадлежащих промежутку [С, D]. Вычислить сумму элементов каждого столбца.

Листинг программы:

Console.WriteLine("Введите размер матрицы N (N < 10):");

int N = Convert.ToInt32(Console.ReadLine());

if (N >= 10)

{

Console.WriteLine("Размер матрицы должен быть меньше 10.");

return;

}

Console.WriteLine("Введите диапазон a и b для заполнения матрицы:");

Console.Write("a: ");

int a = Convert.ToInt32(Console.ReadLine());

Console.Write("b: ");

int b = Convert.ToInt32(Console.ReadLine());

int[,] matrix = new int[N, N];

Random random = new Random();

for (int i = 0; i < N; i++)

{

for (int j = 0; j < N; j++)

{

matrix[i, j] = random.Next(a, b + 1);

}

}

Console.WriteLine("Исходная матрица:");

PrintMatrix(matrix, N);

Console.WriteLine("Введите промежуток [C, D]:");

Console.Write("C: ");

int C = Convert.ToInt32(Console.ReadLine());

Console.Write("D: ");

int D = Convert.ToInt32(Console.ReadLine());

long product = 1;

bool hasNumbersInRange = false;

for (int i = 0; i < N; i++)

{

for (int j = 0; j < N; j++)

{

if (matrix[i, j] >= C && matrix[i, j] <= D)

{

product \*= matrix[i, j];

hasNumbersInRange = true;

}

}

}

if (hasNumbersInRange)

{

Console.WriteLine($"Произведение чисел в промежутке [{C}, {D}]: {product}");

}

else

{

Console.WriteLine($"Нет чисел в указанном диапазоне [{C}, {D}].");

}

Console.WriteLine("Сумма элементов каждого столбца:");

for (int j = 0; j < N; j++)

{

int columnSum = 0;

for (int i = 0; i < N; i++)

{

columnSum += matrix[i, j];

}

Console.WriteLine($"Сумма столбца {j + 1}: {columnSum}");

}

static void PrintMatrix(int[,] matrix, int N)

{

for (int i = 0; i < N; i++)

{

for (int j = 0; j < N; j++)

{

Console.Write(matrix[i, j] + "\t");

}

Console.WriteLine();

}

}

Таблица 1.1 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| N: 3, a: 1, b: 2 | Matrix: 2 1 2 1 2 2 1 2 2 |

Анализ результатов:

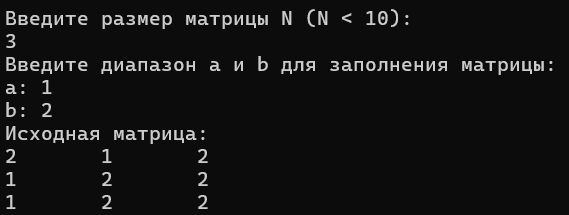


Рисунок 1.1 – Результат работы программы