­­Министерство науки и высшего образования РФ  
ФГАОУ ВПО

Национальный исследовательский технологический университет «МИСиС»

Институт Информационных технологий и компьютерных наук (ИТКН)

Кафедра Инфокоммуникационных технологий (ИКТ)

**Отчет по лабораторной работе №4**

по дисциплине «Программирование и алгоритмизация»

на тему «Матрицы. Типовые алгоритмы обработки матриц»

Выполнил:  
студент группы БИВТ-22-15

Ханцева А.А.

Проверил:   
Нафиков А.М.

Москва, 2022

Оглавление

[Цель 2](#_Toc118639969)

[Задания 2](#_Toc118639970)

[I уровень 2](#_Toc118639971)

[II уровень 2](#_Toc118639972)

[III уровень 2](#_Toc118639973)

[Блок-схемы 4](#_Toc118639974)

[I уровня. Задание 26 I уровня. Задание 27 4](#_Toc118639975)

[I уровня. Задание 28 II уровня. Задание 8 5](#_Toc118639976)

[II уровня. Задание 9 II уровня. Задание 1 6](#_Toc118639977)

[III уровня. Задание 12 III уровня. Задание 13 III уровня. Задание 14 7](#_Toc118639978)

[Листинг программы 8](#_Toc118639979)

[I уровня. Задания 26-27-28 8](#_Toc118639980)

[II уровня. Задания 8-9-1 11](#_Toc118639981)

[III уровня. Задания 12-13-14 14](#_Toc118639982)

[Примеры работы программ 16](#_Toc118639983)

[I уровня. Задания 26-27-28 16](#_Toc118639984)

[II уровня. Задания 8-9-1 17](#_Toc118639985)

[III уровня. Задания 12-13-14 18](#_Toc118639986)

[Вывод 19](#_Toc118639987)

# Цель

Освоение работы с двумерным массивом. Приобретение навыков решения задач с использованием типовых алгоритмов обработки массивов и их сочетаний. Составление задач так, чтобы она могла быть использована для обработки матриц произвольного размера.

# Задания

## I уровень

1. В матрице *А* размера 5 × 7 строку, содержащую максимальный элемент в 6-м столбце, заменить заданным вектором *В* размера 7.
2. В матрице *В* размера 5 × 7 4-й столбец матрицы заменить одномерным массивом, состоящим из максимальных элементов строк, расположенных в обратном порядке (т.е 1-й элемент 4-го столбца – это максимальный элемент 5-й строки и т.д.).
3. В матрице *А* размера 7 × 5 удалить строку с максимальной суммой положительных элементов строки.

## II уровень

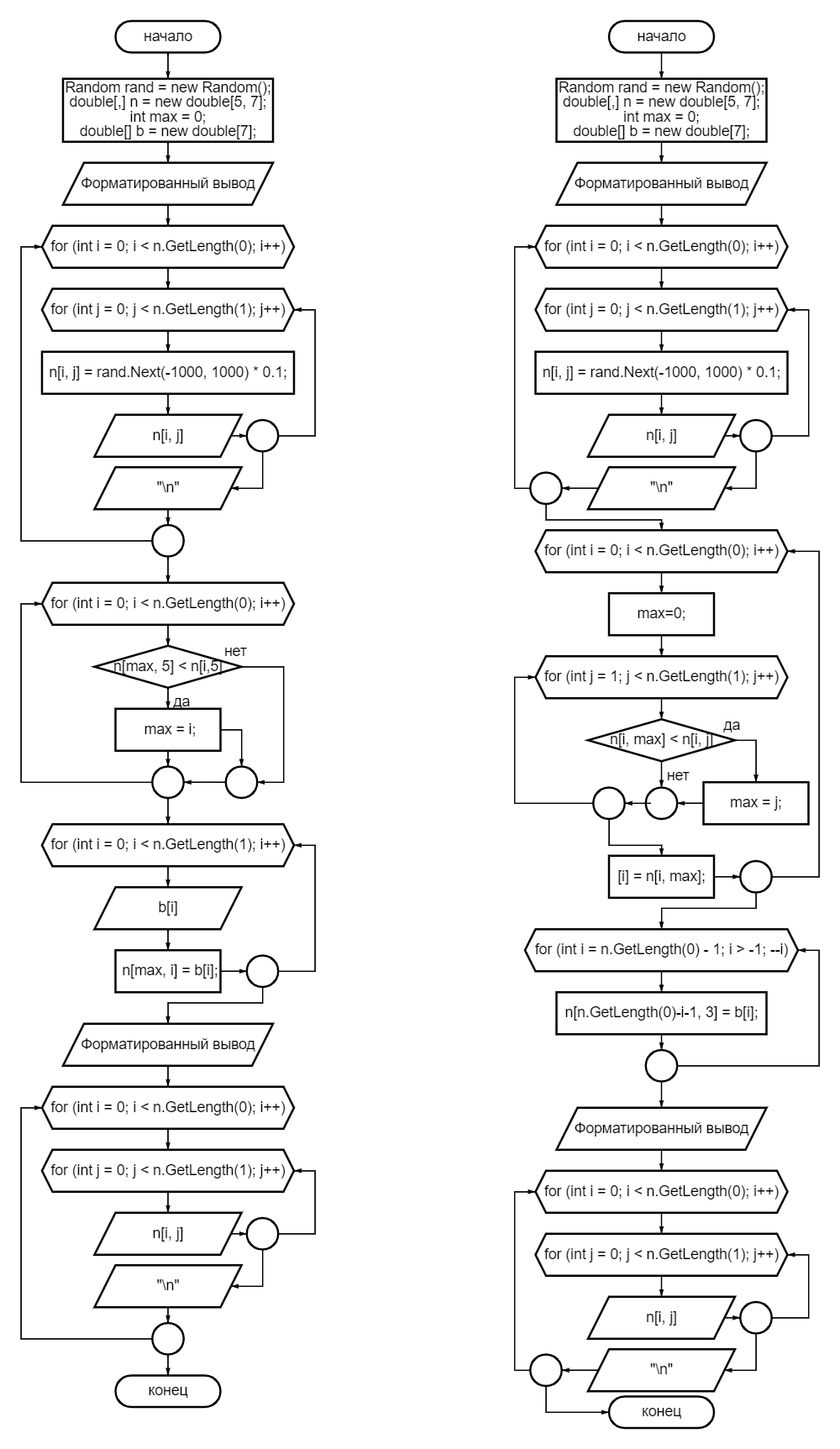
1. В матрице *В* размера 6 × 6 поменять местами максимальные элементы 1-й и 2-й строк, 3-й и 4-й, 5-й и 6-й.
2. В матрице *А* размера 6 × 7 расположить элементы строк в обратном порядке.
3. Дана матрица *A* размера 5 × 7. Для каждой строки сравнить элементы, расположенные непосредственно перед и после максимального элемента этой строки, и меньший из них увеличить в 2 раза. Если максимальный элемент является первым или последним в строке, то увеличить в 2 раза только один соседний с максимальным элемент.

## III уровень

1. Привести заданную квадратную матрицу *а* размера *n* × *n* к такому виду, чтобы все элементы ниже главной диагонали были нулевыми. Использовать линейные преобразования (умножение строки на число, сложение строк). Указание. В цикле по номеру столбца (*j* = 0, 1, … , *n* - 2). В каждом столбце обнулить элементы с (*j* + 1*)*-го до последнего, для этого из каждой *k*-й строки (*k* = *j* + 1, … , *n* - 1) вычесть *j*-ю строку, умноженную на коэффициент *р* = *аkj* / *ajj*; т.е. в цикле по номеру строки *k* (*k* = *j* + 1, … , *n* - 1) организовать цикл по номеру столбца *m* (*m* = *j*, … , *n* – 1), в котором выполнять *аkm* = *аkm* – *аjm  ·p*.
2. В задаче 12 преобразовать матрицу таким образом, чтобы все элементы выше главной диагонали были нулевыми.
3. В задаче 12 преобразовать матрицу таким образом, чтобы все элементы ниже и выше главной диагонали были нулевыми.

# Блок-схемы

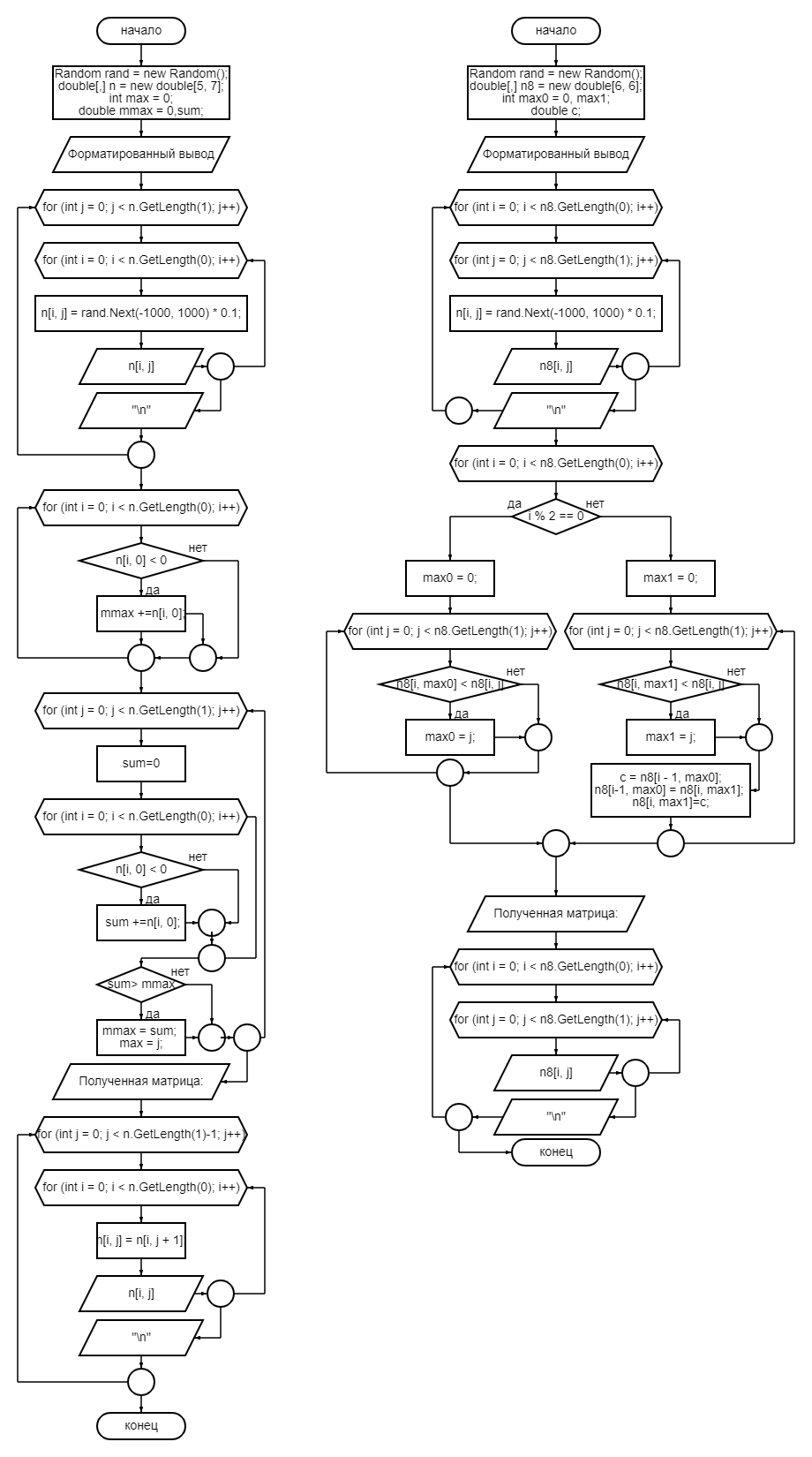
## I уровня. Задание 26 I уровня. Задание 27



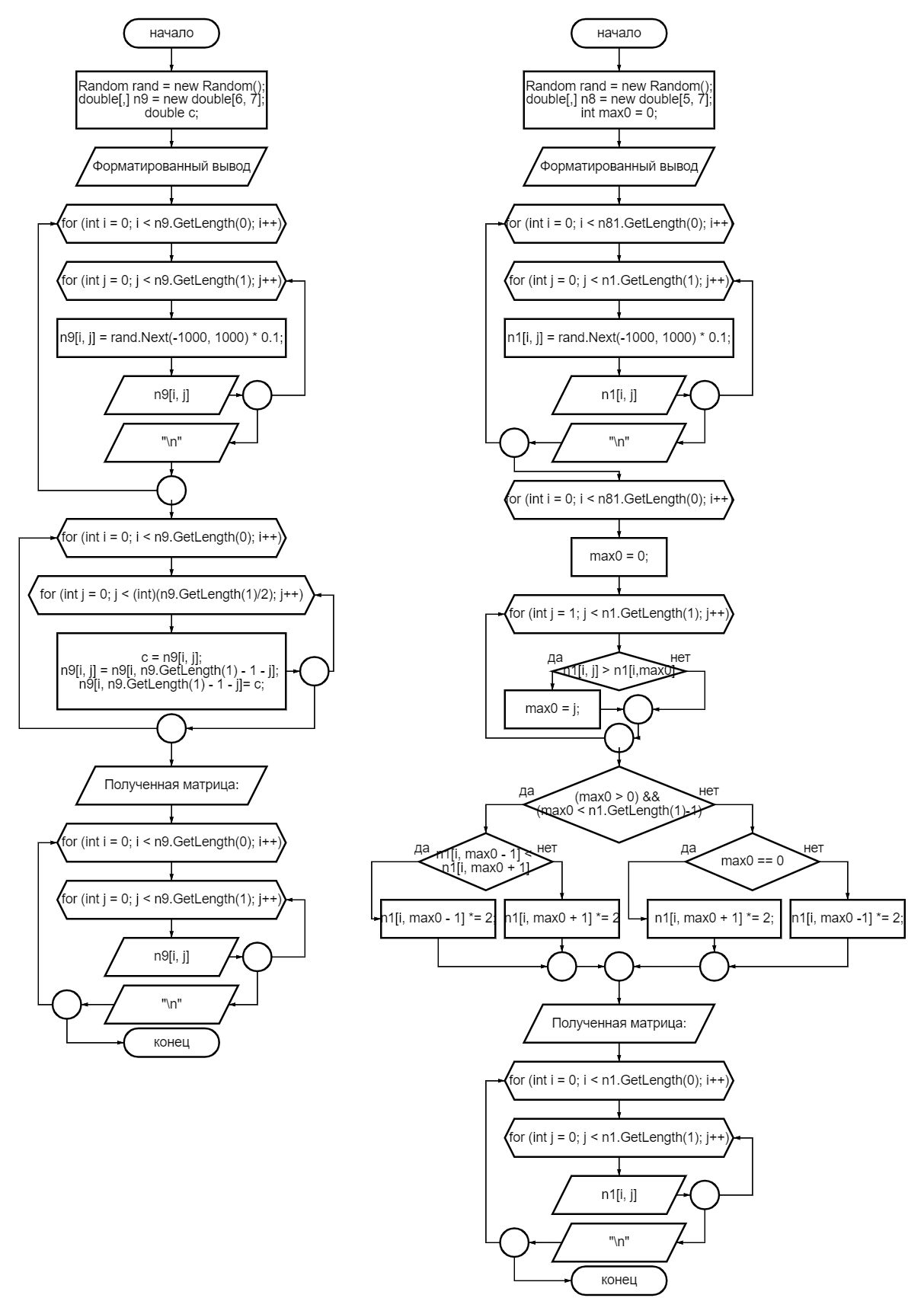
Полученная матрица:

Полученная матрица:

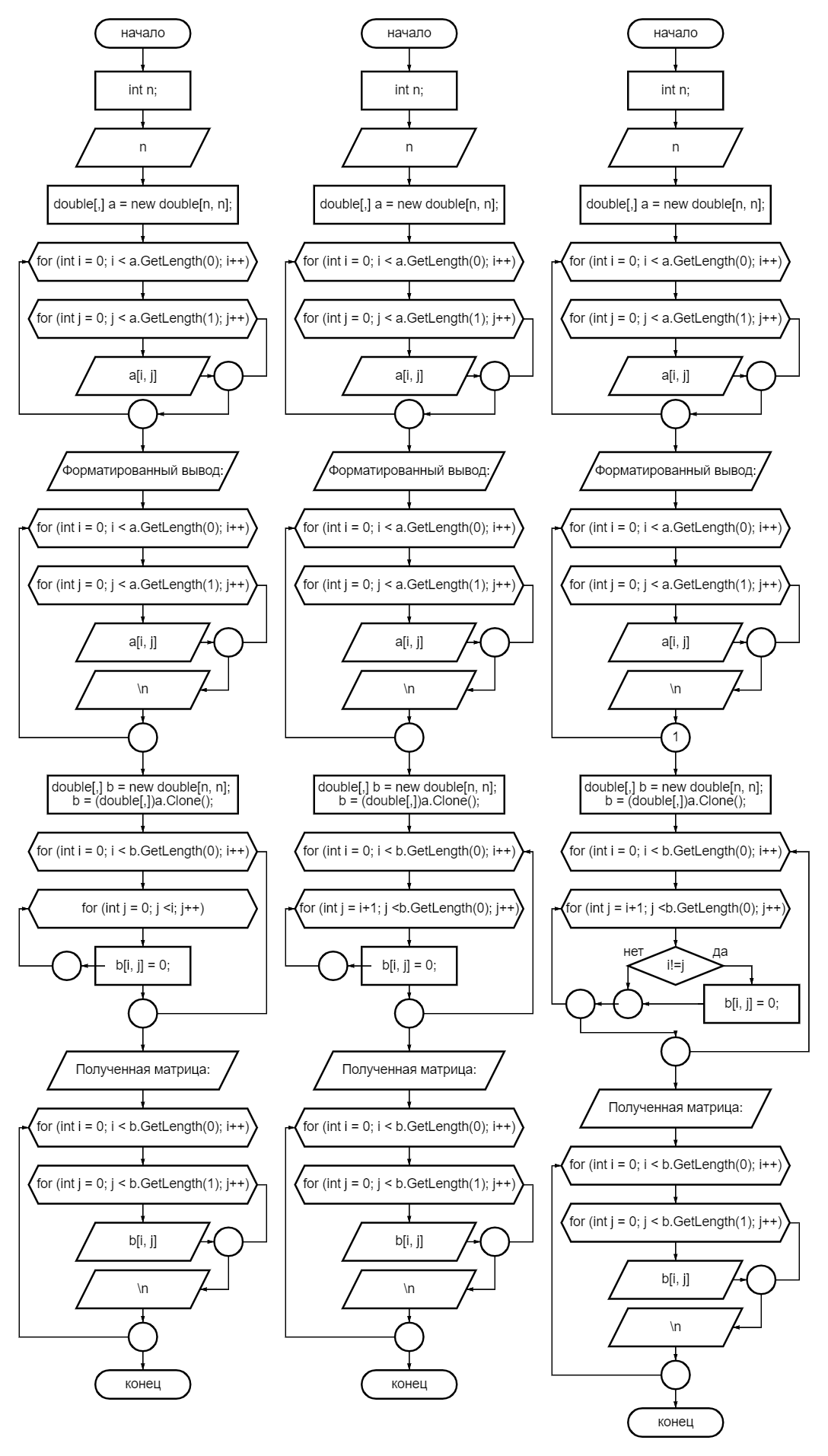
## I уровня. Задание 28 II уровня. Задание 8



## II уровня. Задание 9 II уровня. Задание 1



## III уровня. Задание 12 III уровня. Задание 13 III уровня. Задание 14



# Листинг программы

## I уровня. Задания 26-27-28

internal class Program

{

static void Main(string[] args)

{

Random rand = new Random();

Console.WriteLine("Лабораторная работа №4. Сложность 1\n\nЗадание 26\n Матрицы размера 5 на 7 по строкам.");

double[,] n = new double[5, 7];

Console.WriteLine("Форматированный вывод:");

for (int i = 0; i < n.GetLength(0); i++)

{

for (int j = 0; j < n.GetLength(1); j++)

{

n[i, j] = rand.Next(-1000, 1000) \* 0.1;

Console.Write($"{n[i, j],8}");

}

Console.Write("\n");

}

int max = 0;

for (int i=1; i< n.GetLength(0); i++)

if (n[max, 5] < n[i,5])

max = i;

double[] b = new double[7];

Console.WriteLine("Введите через Enter массив b размера 7:");

for (int i = 0; i < n.GetLength(1); i++)

{

Console.Write($"b[{i + 1}]=");

while (!double.TryParse(Console.ReadLine().Replace('.', ','), out b[i]))

Console.Write($"Упс! Некорректное значение. Попробуйте ещё раз b[{i + 1}]=");

}

for (int i = 0; i < n.GetLength(1); i++)

n[max, i] = b[i];

Console.WriteLine("Полученная матрица:");

for (int i = 0; i < n.GetLength(0); i++)

{

for (int j = 0; j < n.GetLength(1); j++)

Console.Write($"{n[i, j],8}");

Console.Write("\n");

}

Console.Write("\nЗадание 27\n Используем в качестве изначальной матрицы" +

" полученную из предыдущего задания");

for (int i = 0; i < n.GetLength(0); i++)

{

max = 0;

for (int j = 1; j < n.GetLength(1); j++)

if (n[i, max] < n[i, j])

max = j;

b[i] = n[i, max];

}

Console.Write($"\n");

for (int i = n.GetLength(0) - 1; i > -1; --i)

n[n.GetLength(0)-i-1, 3] = b[i];

Console.WriteLine("Полученная матрица:");

for (int i = 0; i < n.GetLength(0); i++)

{

for (int j = 0; j < n.GetLength(1); j++)

Console.Write($"{n[i, j],8}");

Console.Write("\n");

}

Console.Write($"\nЗадание 28\n Так как, по условию, в данном задание заданная" +

$" матрица имеет столько же строк сколько столбцов предыдущая матрица и столько стобов сколько строк, " +

$"используем матрицу полученную из предыдущего задания. Изначальная матрица:\n");

for (int j = 0; j < n.GetLength(1); j++)

{

for (int i = 0; i < n.GetLength(0); i++)

Console.Write($"{n[i, j],8}");

Console.Write("\n");

}

double mmax = 0,sum;

for (int i = 0; i < n.GetLength(0); i++)

if (n[i, 0] > 0)

mmax += n[i, 0];

max = 0;

for (int j = 1; j < n.GetLength(1); j++)

{

sum = 0;

for (int i = 0; i < n.GetLength(0); i++)

if (n[i, j] > 0)

sum += n[i, j];

if (sum> mmax)

{

mmax = sum;

max = j;

}

}

for (int j = max; j < n.GetLength(1)-1; j++)

for (int i = 0; i < n.GetLength(0); i++)

n[i, j] = n[i, j + 1];

Console.WriteLine("Полученная матрица:");

for (int j = 0; j < n.GetLength(1)-1; j++)

{

for (int i = 0; i < n.GetLength(0); i++)

Console.Write($"{n[i, j],8}");

Console.Write("\n");

}

}

}

## II уровня. Задания 8-9-1

internal class Program

{

static void Main(string[] args)

{

Random rand = new Random();

Console.WriteLine("Лабораторная работа №4. Сложность 2\n\nЗадание 8\n Матрица размера 6 на 6 по строкам");

double[,] n8 = new double[6, 6];

Console.WriteLine("Форматированный вывод:");

for (int i = 0; i < n8.GetLength(0); i++)

{

for (int j = 0; j < n8.GetLength(1); j++)

{

n8[i, j] = rand.Next(-1000, 1000) \* 0.1;

Console.Write($"{n8[i, j],8}");

}

Console.Write("\n");

}

int max0=0,max1;

double c;

for (int i = 0; i < n8.GetLength(0); i++) {

if (i % 2 == 0) {

max0 = 0;

for (int j = 0; j < n8.GetLength(1); j++)

if (n8[i, max0] < n8[i, j])

max0 = j;

}

else

{

max1 = 0;

for (int j = 0; j< n8.GetLength(1); j++)

if (n8[i, max1] < n8[i, j])

max1 = j;

c = n8[i - 1, max0];

n8[i-1, max0] = n8[i, max1];

n8[i, max1]=c;

}

}

Console.WriteLine("Полученная матрица:");

for (int i = 0; i < n8.GetLength(0); i++)

{

for (int j = 0; j < n8.GetLength(1); j++)

Console.Write($"{n8[i, j],8}");

Console.Write("\n");

}

Console.WriteLine("\nЗадание 9\n Матрица размера 6 на 7 по строкам");

double[,] n9 = new double[6, 7];

Console.WriteLine("Форматированный вывод:");

for (int i = 0; i < n9.GetLength(0); i++)

{

for (int j = 0; j < n9.GetLength(1); j++)

{

n9[i, j] = rand.Next(-1000, 1000) \* 0.1;

Console.Write($"{n9[i, j],8}");

}

Console.Write("\n");

}

for (int i = 0; i < n9.GetLength(0); i++)

for (int j = 0; j < (int)(n9.GetLength(1)/2); j++)

{

c = n9[i, j];

n9[i, j] = n9[i, n9.GetLength(1) - 1 - j];

n9[i, n9.GetLength(1) - 1 - j]= c;

}

Console.WriteLine("Полученная матрица:");

for (int i = 0; i < n9.GetLength(0); i++)

{

for (int j = 0; j < n9.GetLength(1); j++)

Console.Write($"{n9[i, j],8}");

Console.Write("\n");

}

Console.WriteLine("\nЗадание 1\n Матрица размера 5 на 7 по строкам:");

double[,] n1 = new double[5, 7];

Console.WriteLine("Форматированный вывод:");

for (int i = 0; i < n1.GetLength(0); i++)

{

for (int j = 0; j < n1.GetLength(1); j++)

{

n1[i, j] = rand.Next(-1000, 1000) \* 0.1;

Console.Write($"{n1[i, j],8}");

}

Console.Write("\n");

}

for (int i=0; i < n1.GetLength(0); i++)

{

max0 = 0;

for(int j=1;j< n1.GetLength(1); j++)

if (n1[i, j] > n1[i,max0])

max0 = j;

if ((max0 > 0) && (max0 < n1.GetLength(1)-1))

{

if (n1[i, max0 - 1] < n1[i, max0 + 1])

n1[i, max0 - 1] \*= 2;

else

n1[i, max0 + 1] \*= 2;

}

else {

if (max0 == 0)

n1[i, max0 + 1] \*= 2;

else

n1[i, max0 -1] \*= 2;

}

}

Console.WriteLine("Полученная матрица:");

for (int i = 0; i < n1.GetLength(0); i++)

{

for (int j = 0; j < n1.GetLength(1); j++)

Console.Write($"{n1[i, j],8}");

Console.Write("\n");

}

}

}

## III уровня. Задания 12-13-14

internal class Program

{

static void Main(string[] args)

{

Console.Write("Лабораторная работа №4. Сложность 3\n Введите " +

"элементы матрицы размера n на n по строкам каждый элемент " +

"через Enter. Сначала введите n:");

int n;

while (!int.TryParse(Console.ReadLine().Replace('.', ','), out n))

Console.Write($"Упс! Некорректное значение. Попробуйте ещё раз n=");

double[,] a = new double[n, n];

for (int i = 0; i < a.GetLength(0); i++)

for (int j = 0; j < a.GetLength(1); j++)

{

Console.Write($"a[{i + 1},{j + 1}]=");

while (!double.TryParse(Console.ReadLine().Replace('.', ','), out a[i, j]))

Console.Write($"Упс! Некорректное значение. Попробуйте ещё раз a[{i + 1},{j + 1}]=");

}

Console.WriteLine("Форматированный вывод:");

for (int i = 0; i < a.GetLength(0); i++)

{

for (int j = 0; j < a.GetLength(1); j++)

Console.Write($"{a[i, j],5}");

Console.Write("\n");

}

Console.WriteLine($"\nЗадание 12");

double[,] b = new double[n, n] ;

b = (double[,])a.Clone();

for (int i = 0; i < b.GetLength(0); i++)

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

b[i, j] = 0;

Console.WriteLine("Полученная матрица:");

for (int i = 0; i < b.GetLength(0); i++)

{

for (int j = 0; j < b.GetLength(1); j++)

Console.Write($"{b[i, j],5}");

Console.Write("\n");

}

Console.WriteLine($"\nЗадание 13");

b = (double[,])a.Clone();

for (int i = 0; i < b.GetLength(0); i++)

for (int j = i+1; j < b.GetLength(0); j++)

b[i, j] = 0;

Console.WriteLine("Полученная матрица:");

for (int i = 0; i < b.GetLength(0); i++)

{

for (int j = 0; j < b.GetLength(1); j++)

Console.Write($"{b[i, j],5}");

Console.Write("\n");

}

Console.WriteLine($"\nЗадание 14");

b = (double[,])a.Clone();

for (int i = 0; i < b.GetLength(0); i++)

for (int j = 0; j < b.GetLength(0); j++)

if (i!=j)

b[i, j] = 0;

Console.WriteLine("Полученная матрица:");

for (int i = 0; i < b.GetLength(0); i++)

{

for (int j = 0; j < b.GetLength(1); j++)

Console.Write($"{b[i, j],5}");

Console.Write("\n");

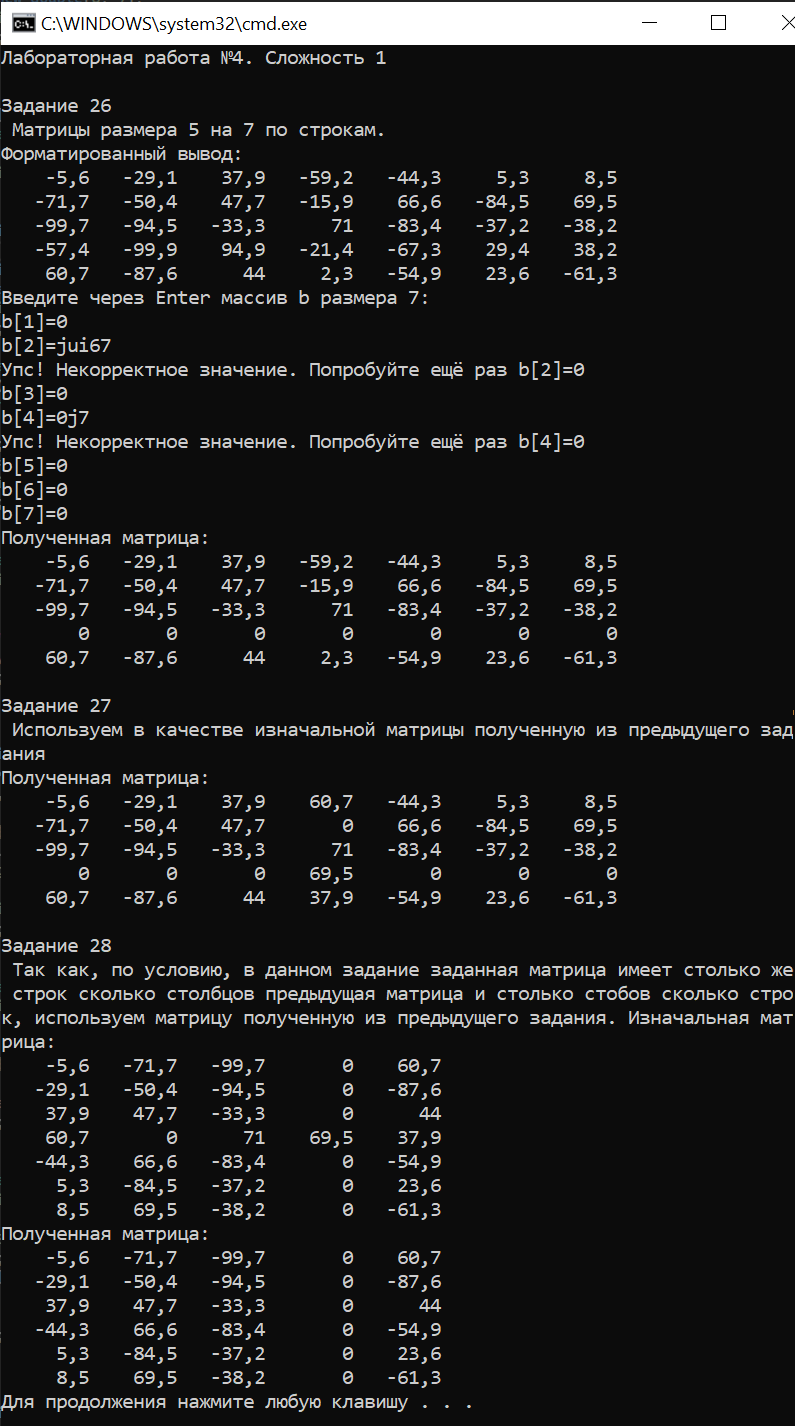
}

}

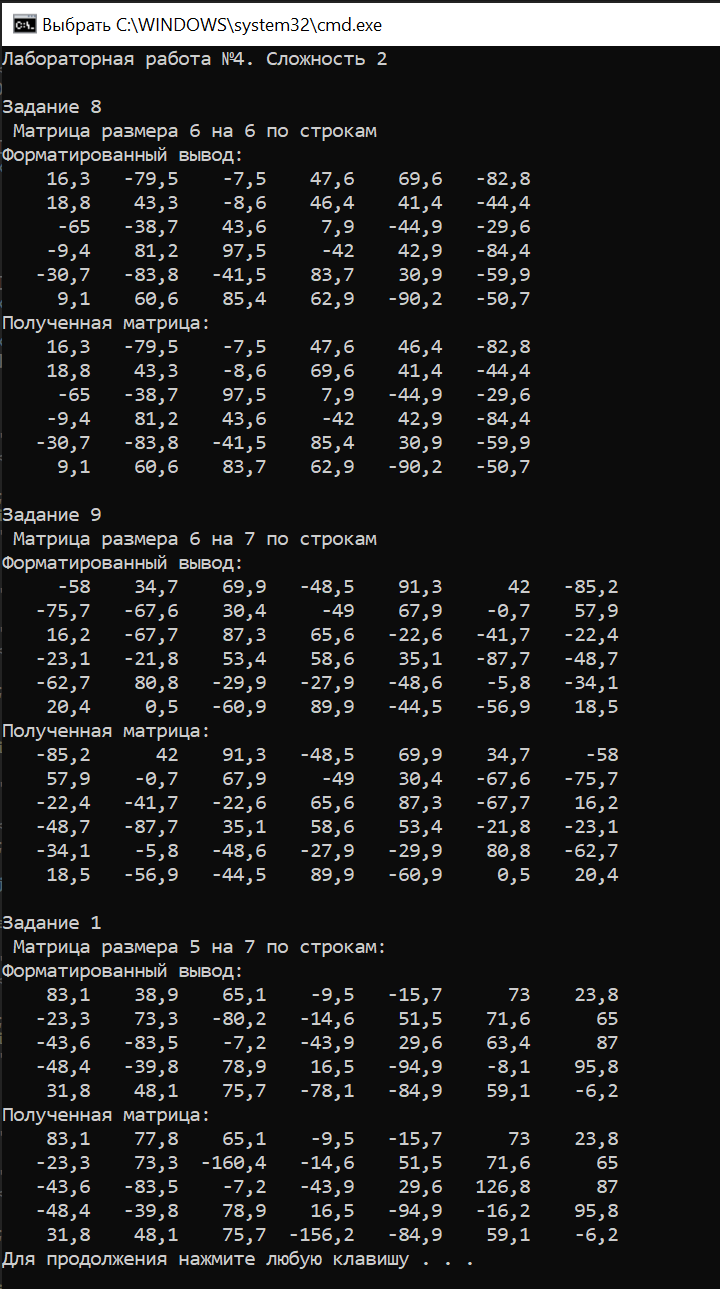
}

# Примеры работы программ

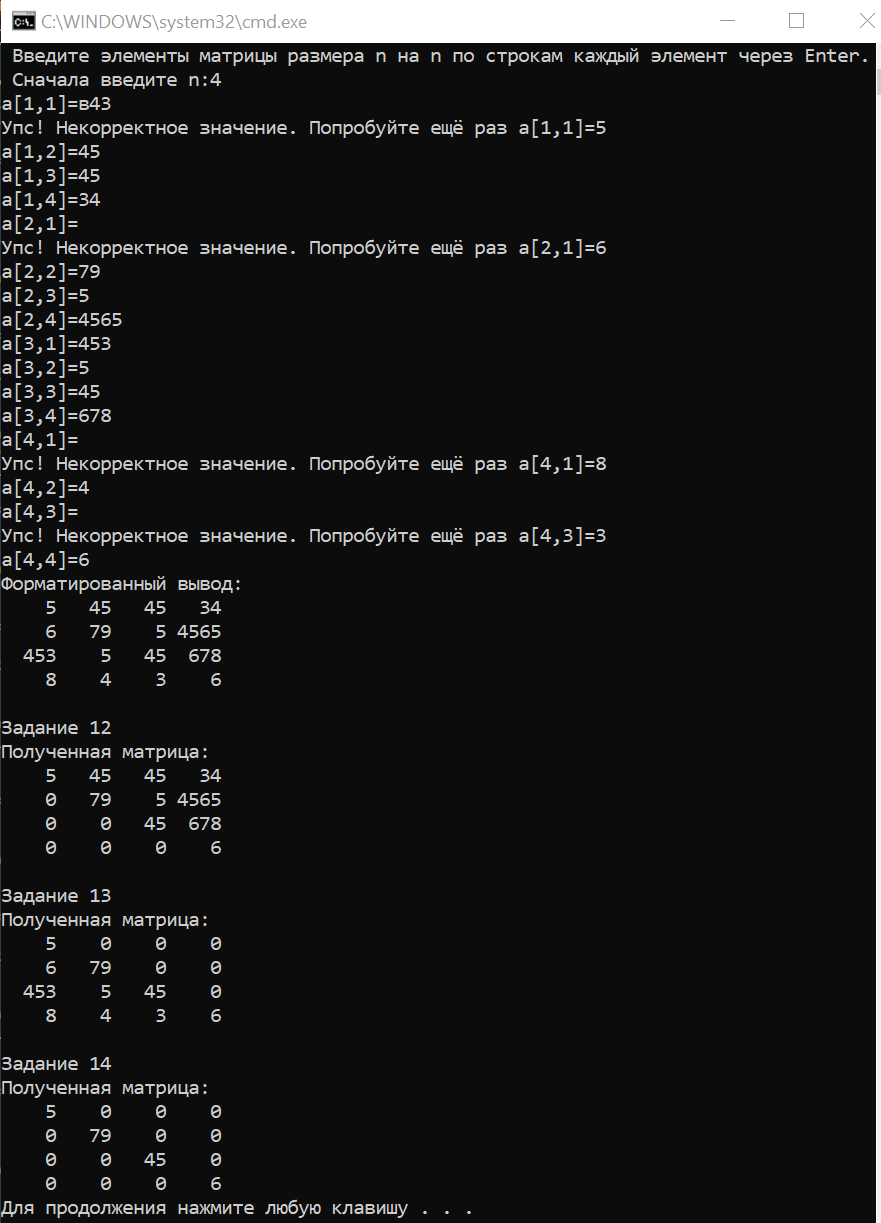
## I уровня. Задания 26-27-28



## II уровня. Задания 8-9-1



## III уровня. Задания 12-13-14



# Вывод

Освоен перебор элементов матрица такие как по строкам, по столбцам, под и над главной диагональю. Изучен класс Random для генерации случайных чисел для двумерного массива.