Министерство науки и высшего образования Российской Федерации  
Федеральное государственное автономное учреждение высшего образования  
«Пермский национальный исследовательский политехнический университет»

Электротехнический факультет  
Кафедра «Информационные технологии и автоматизированные системы»

**О Т Ч Ё Т**

Дискретная математика

Семестр 4

Лабораторная работа 5

Выполнил  
студент группы РИС-22-2б  
Баяндин К.С.

Проверила  
старший преподаватель кафедры ИТАС  
Рустамханова Г.И.

Пермь, 2024

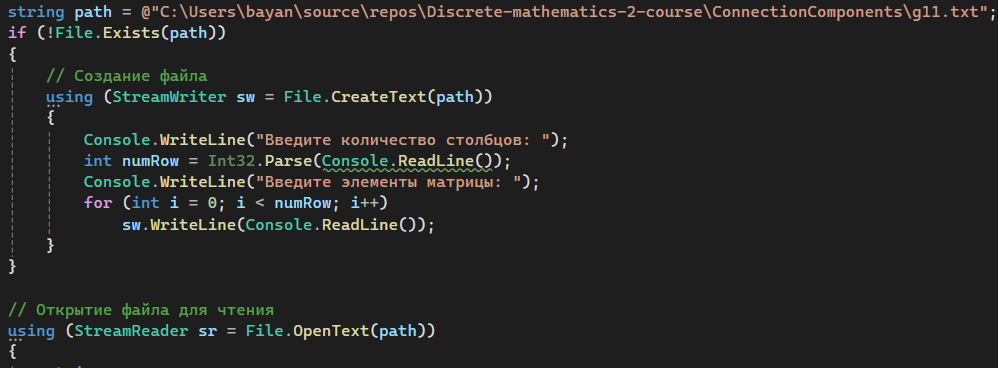
Постановка задачи

На вход поступает матрица смежности без петель (10х10). Ввод матрицы должен быть из файла. Требуется:

1. Построить матрицу достижимости
2. Вычислить компоненты связности

Алгоритмы работы программы

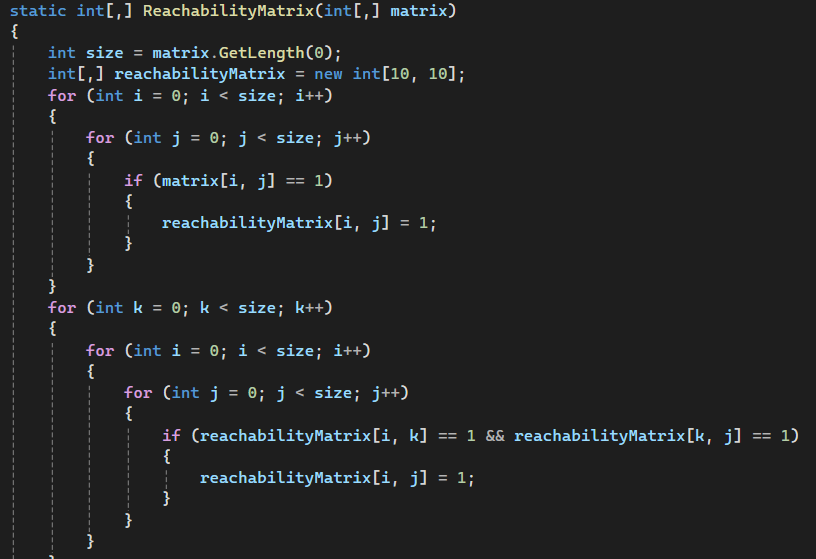
Матрица поступает из файла, в случае если файла не существует, программа позволяет создать матрицу и записать его в файл:



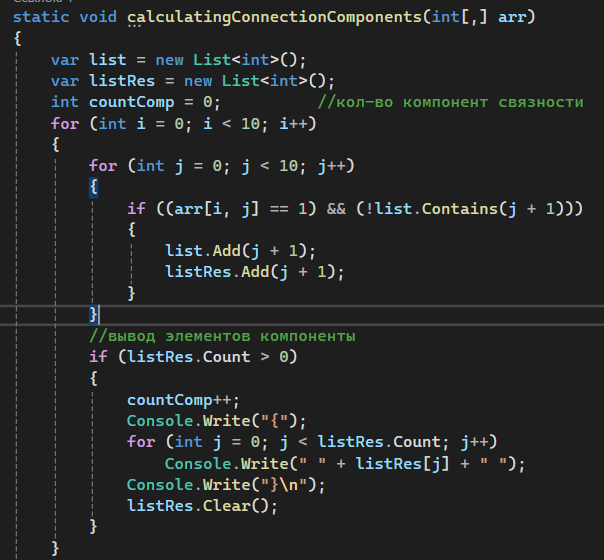
*Рисунок 1 – Ввод матрицы*

Следующий шаг: вывод матрицы из файла. Так как данные из файла поступают в виде строки, следует преобразовать строку в двумерный массив.

Далее реализованы 2 функции: *ReachabilityMatrix()* и *calculatingConnectionComponents()*. Первая функция создает матрицу достижимости по матрице смежности, вторая - вычисляет компоненты связности.

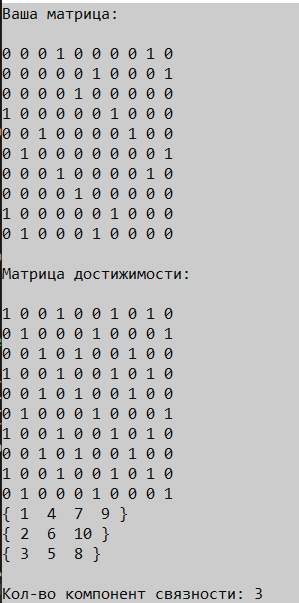


*Рисунок 2 – Матрица достижимости*

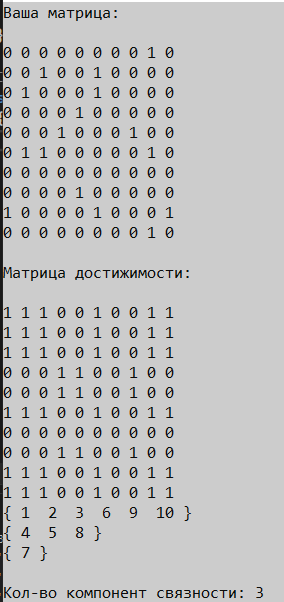


*Рисунок 3 – Вычисление компонент связности*

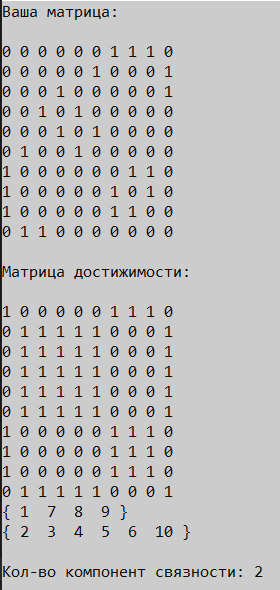
**Результаты работы программы:**



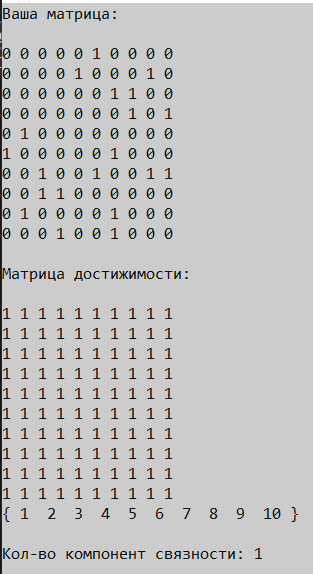
*Рисунок 4 - 1 файл*



*Рисунок 5 - 2 файл*



*Рисунок 6 - 3 файл*



*Рисунок 7 - 4 файл*

**Листинг:**

using System;

using System.Drawing;

namespace ConnectionComponents

{

class Program

{

//матрица достижимости

static int[,] ReachabilityMatrix(int[,] matrix)

{

int size = matrix.GetLength(0);

int[,] reachabilityMatrix = new int[10, 10];

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

{

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

{

if (matrix[i, j] == 1)

{

reachabilityMatrix[i, j] = 1;

}

}

}

for (int k = 0; k < size; k++)

{

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

{

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

{

if (reachabilityMatrix[i, k] == 1 && reachabilityMatrix[k, j] == 1)

{

reachabilityMatrix[i, j] = 1;

}

}

}

}

//вывод матрицы достижимости

Console.WriteLine("\nМатрица достижимости:\n");

for (int i = 0; i < 10; i++)

{

for (int j = 0; j < 10; j++)

{

Console.Write(reachabilityMatrix[i, j] + " ");

}

Console.WriteLine();

}

return reachabilityMatrix;

}

//вычисление компонент связности

static void calculatingConnectionComponents(int[,] arr)

{

var list = new List<int>();

var listRes = new List<int>();

int countComp = 0; //кол-во компонент связности

for (int i = 0; i < 10; i++)

{

for (int j = 0; j < 10; j++)

{

if ((arr[i, j] == 1) && (!list.Contains(j + 1)))

{

list.Add(j + 1);

listRes.Add(j + 1);

}

}

//вывод элементов компоненты

if (listRes.Count > 0)

{

countComp++;

Console.Write("{");

for (int j = 0; j < listRes.Count; j++)

Console.Write(" " + listRes[j] + " ");

Console.Write("}\n");

listRes.Clear();

}

}

//вывод компонент с 1 элементом

for (int i = 0; i < 10; i++)

{

if (!list.Contains(i + 1))

{

Console.WriteLine("{ " + (i + 1) + " }");

countComp++;

}

}

Console.WriteLine("\nКол-во компонент связности: " + countComp);

}

static void Main(string[] args)

{

string path = @"C:\Users\bayan\source\repos\Discrete-mathematics-2-course\ConnectionComponents\g14.txt";

if (!File.Exists(path))

{

// Создание файла

using (StreamWriter sw = File.CreateText(path))

{

Console.WriteLine("Введите количество столбцов: ");

int numRow = Int32.Parse(Console.ReadLine());

Console.WriteLine("Введите элементы матрицы: ");

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

sw.WriteLine(Console.ReadLine());

}

}

// Открытие файла для чтения

using (StreamReader sr = File.OpenText(path))

{

string s;

string matrix = "";

int k = 0;

//вывод матрицы + создание строки матрицы

Console.WriteLine("Ваша матрица:\n");

while ((s = sr.ReadLine()) != null)

{

Console.WriteLine(s);

k++;

matrix += s + " ";

}

int rows = k;

int columns = rows;

int count = 0;

//создание матрицы

int[,] arr = new int[rows, columns];

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

{

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

{

arr[i, j] = (int)Char.GetNumericValue(matrix[count]);

count += 2;

}

}

int[,] reachabilityMatrix = ReachabilityMatrix(arr);

calculatingConnectionComponents(reachabilityMatrix);

}

}

}

}