Пензенский государственный университет  
Кафедра «Вычислительной техники»

**Отчет**по лабораторной работе №4  
по курсу "ЛиОАвИЗ"

**Выполнили студент группы 20ВВ1:**

Вяльмисов М.О.

**Приняли:**

Юрова О.В.

Акифьев И.В.

Пенза 2021

**Лабораторные задания:**

**Задание 1**

1. Сгенерируйте (используя генератор случайных чисел) матрицу смежности для неориентированного графа *G*. Выведите матрицу на экран.
2. Для сгенерированного графа осуществите процедуру обхода в глубину, реализованную в соответствии с приведенным выше описанием.

**3.**\* Реализуйте процедуру обхода в глубину для графа, представленного списками смежности.

**Задание 2\***

1. Для матричной формы представления графов выполните преобразование рекурсивной реализации обхода графа к не рекурсивной.

**Листинг алгоритмов:**

Добавление в класс MatrixGraph:

public List<int> DeepWalk()

{

\_walkedList = new bool[\_size];

Walk(0);

return \_deepWalkList;

}

private void Walk(int v)

{

\_walkedList[v] = true;

\_deepWalkList.Add(v + 1);

for (int i = 0; i < \_size; i++)

{

if (\_matrix[v, i] == 1 && !\_walkedList[i])

{

Walk(i);

}

}

}

public List<int> DeepWalkNonRecursive()

{

\_savedV = new List<int>();

\_deepWalkList = new List<int>();

\_walkedList = new bool[\_size];

NonRecursiveWalk();

return \_deepWalkList;

}

private void NonRecursiveWalk()

{

\_savedV.Add(0);

\_deepWalkList.Add(1);

\_walkedList[0] = true;

for (int i = 0; i < \_size; i++)

{

if (\_savedV.Count == 0)

{

return;

}

if (\_matrix[\_savedV[^1], i] == 1 && !\_walkedList[i])

{

\_savedV.Add(i);

\_walkedList[i] = true;

\_deepWalkList.Add(i + 1);

i = 0;

}

if (i != \_size - 1)

{

continue;

}

\_savedV.RemoveAt(\_savedV.Count - 1);

i = 0;

}

}

Добавление в класс ListGraph:

public List<int> DeepWalk()

{

\_walkedList = new bool[\_list.Count];

for (int i = 0; i < \_list.Count; i++)

{

Walk(i);

}

return null;

}

private void Walk(int v)

{

\_walkedList[v] = true;

\_deepWalkList.Add(v + 1);

for (int i = 0; i < \_list[v].Count; i++)

{

if (!\_walkedList[\_list[v][i] - 1])

{

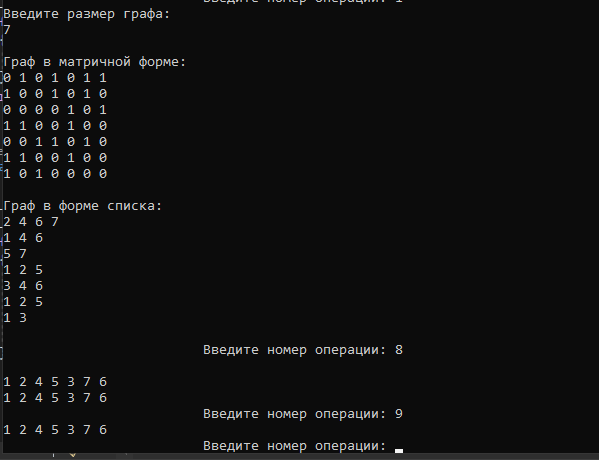
Walk(\_list[v][i] - 1);

}

}

}

**Результаты выполнения операций:**



**Вывод:** Научился реализовывать алгоритм обхода графа.