Министерство науки и высшего образования РФ

Федеральное государственное бюджетное образовательное учреждение

высшего образования

«Курский государственный университет»

Кафедра программного

обеспечения и администрирования

информационных систем

Направление подготовки

математическое обеспечение и

администрирование

информационных систем

Форма обучения очная

**Отчет**

**по лабораторной работе №7**

«Реализация алгоритмов на графах, представленных с помощью матриц»

Выполнил:

студент группы 213 Тихонов Е.Е.

Проверил:

профессор кафедры ПОиАИС Кудинов В.А.

Курск, 2022

**Цель работы**: Изучить основные алгоритмы теории графов.

**Задание**

Определите, верно ли, что для любой пары вершин заданного ориентированного графа одна из этих вершин достижима от другой.

**Алгоритм решения задачи**

Связный граф — граф, содержащий ровно одну компоненту связности. Это означает, что между любой парой вершин этого графа существует как минимум один путь. Алгоритм решения задачи строится на поиске матрицы достижимости графа. Матрица достижимости показывает, сколько существует путей из одной вершины в другую, следовательно, в связном графе, где все вершины взаимно достижимы, на всех позициях матрицы будут значения большие нуля. Если на какой-либо позиции в матрице достижимости присутствует ноль, то пути между соответствующими вершинами нет. Матрица достижимости вычисляется по формуле: . Где n - количество вершин графа, S - матрица смежности графа, E – единичная матрица, где единицы расположены по главной диагонали.

**Код программы**

import numpy

# функция вычисления связности

def graph\_connectivity(S):

# матрица достижимости вычисляется по формуле

# С = E + S + S^2 + S^3 + S^4 + ... + S^n

# где n - количество вершин графа

# E - единичная матрица (единицы по диагонале)

n = S.shape[0]

E = numpy.eye(n)

С = E + S

S\_i = S # S^i

for i in range(1, n):

S\_i = numpy.dot(S\_i, S)

С = С + S\_i

print('матрица достижимости')

print(С) # показывает количество путей длины от 1 до 4 между вершинами орграфа

elem\_prod = numpy.prod(С)

if elem\_prod != 0:

print('граф связный')

else:

print('граф не связный')

# матрица смежности S1

S1 = numpy.array([

[0, 1, 0, 0],

[0, 0, 1, 0],

[0, 0, 0, 1],

[1, 0, 0, 0]])

# матрица смежности S2

S2 = numpy.array([

[0, 1, 1, 0],

[0, 0, 0, 0],

[0, 1, 0, 1],

[0, 0, 1, 0]])

graph\_connectivity(S1)

graph\_connectivity(S2)

**Тест программы**

Тест программы представлен на рисунке 1. Внешний вид графов для тестирования программы представлен на рисунках 2-3.

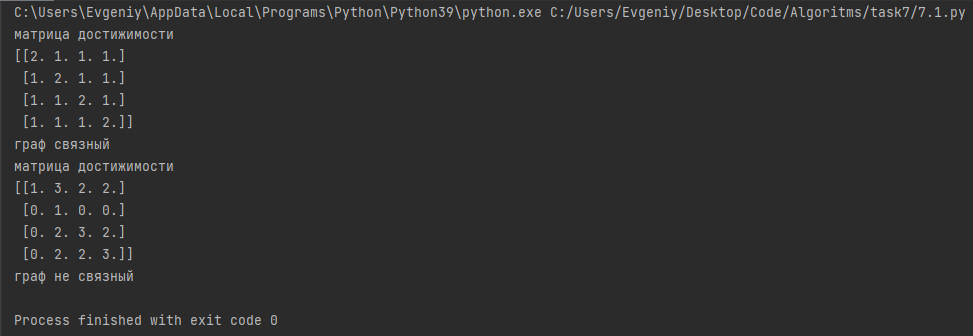


Рисунок 1 - Тест программы 1

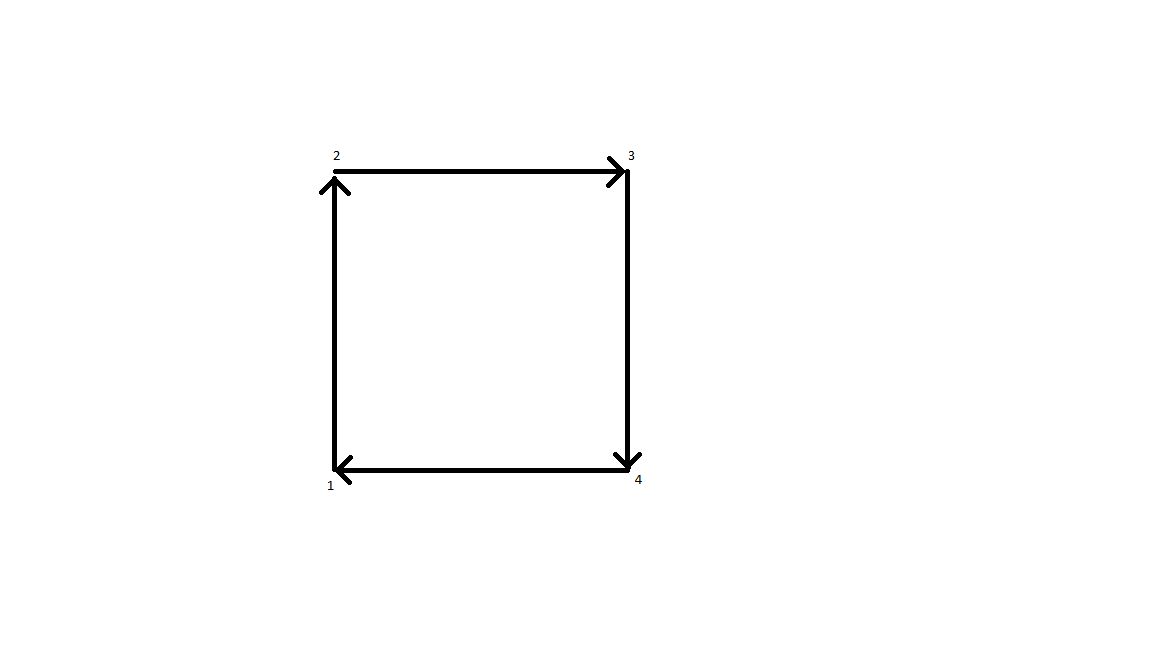


Рисунок 2 – Внешний вид связанного графа для тестирования программы

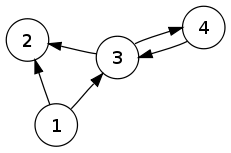


Рисунок 3 - Внешний вид не связанного графа для тестирования программы