Министерство образования и науки Российской Федерации

Федеральное государственное автономное образовательное

учреждение высшего образования

«Национальный исследовательский

Томский политехнический Университет»



Инженерная школа ядерных технологий

Направление 01.04.02 «Прикладная математика и информатика»

**ОТЧЕТ**

по ИНДИВИДУАЛЬНОМУ ЗАДАНИЮ №5

**Основные понятия теории графов**

Вариант 1

по дисциплине:

**Дискретная математика и теория графов**

|  |  |  |  |
| --- | --- | --- | --- |
| **Исполнитель:** |  | Е. В. Петрович | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
| студент группы 0ВМ92 |  | Дата сдачи: | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
|  |  |  |  |
|  |  |  |  |
| **Руководитель:** |  | М. Л. Шинкеев | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
| доцент, |  | Дата проверки: | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |
| кандидат физико-математических наук |  |  |  |
|  |  |  |  |

Томск - 2019

Оглавление

[Задание варианта 1 2](#_Toc30182616)

[**Цель работы** 3](#_Toc30182617)

[**Теоретическая часть** 3](#_Toc30182618)

[**Основная часть** 4](#_Toc30182619)

[Задание 1. Построить матрицы смежности и инциденций 4](#_Toc30182620)

[Задание 2. Построить матрицу расстояний между вершинами графа 5](#_Toc30182621)

[Задание 3. Найти диаметр графа, радиус графа, центр графа 7](#_Toc30182622)

[Задание 4. Составить обход графа в ширину и глубину, используя в качестве начальной вершины - вершину v8 7](#_Toc30182623)

[**Заключение** 9](#_Toc30182624)

[**Приложение 1** 10](#_Toc30182625)

[Задание 2 10](#_Toc30182626)

[Задание 3 13](#_Toc30182627)

[**Приложение 2** 15](#_Toc30182628)

[Ссылка на проект 15](#_Toc30182629)

**Задание**

Задание варианта 1

Простой граф задан диаграммой (см. варианты задания). Требуется:

1. Построить матрицы смежности и инциденций.
2. Построить матрицу расстояний между вершинами графа.
3. Найти диаметр графа, радиус графа, центр графа.
4. Составить обход графа в ширину и глубину, используя в качестве начальной вершины - вершину v8.



**Цель работы**

Работа с графами.

**Теоретическая часть**

Граф – абстрактный математический объект, представляющий собой упорядоченную пару множества вершин V и множества соединяющих вершины ребер E – G(V,E).

Матрица смежности – способ задания графа, представляющий таблицу , где n – порядок графа, строки и колонки которой, представляют вершины. Если в графе существует ребро концами которого являются вершины  и , то в ячейке матрицы смежности ставится 1. В остальных ячейках ставится 0.

Матрица инциденций - способ задания графа, представляющий таблицу, где строки соответствуют вершинам графа, а столбцы соответствуют рёбрам. В ячейку матрицы на пересечении строки и столбца записывается 1, если ребро инцидентно вершине .

Матрица расстояний – в общем случае, это матрица взвешенной смежности. Как и в матрице смежности строки и колонки представляют вершины. В ячейках содержатся минимальные расстояния между вершинами и . Для невзвешенного графа расстояния измеряются в количестве ребер, для взвешенного – весах ребер.

Наибольшее расстояние от вершины до самой удаленной вершины называется эксцентриситетом вершины . Наименьший эксцентриситет графа – радиус графа, наибольший эксцентриситет – диаметр графа. Вершина, обладающая наименьшим эксцентриситетом, - центр графа.

Обход графа – посещение всех вершин графа. При обходе в ширину сначала посещаются все соседние вершины от исходной, далее посещаются вершины находящиеся на расстоянии двух ребер от исходной. То есть на каждом шаге k посещаются вершины, находящиеся на расстоянии k ребер от начальной вершины. При обхрде в глубину проход осуществляется вдоль цепи ребер до тупиковой вершины. Тупиковая вершина – вершина, смежные вершины которой уже посещены. Далее происходит возврат по цепи ребер до вершины , которая имеет непосещенные смежные вершины и происходит проход по цепи ребер отходящей от вершины до достижения тупиковой вершины и т.д.

**Основная часть**

Задание 1. Построить матрицы смежности и инциденций

Таблица 1. Матрица смежности

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
| 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 2 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 3 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 4 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| 5 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 6 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 1 |
| 7 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 1 |
| 8 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 |
| 9 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 10 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 11 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 12 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 13 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 14 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| 15 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 16 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 17 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| 18 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 1 | 0 |
| 19 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| 20 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |

Таблица 2. Матрица инциденций

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 1-4 | 1-17 | 2-3 | 2-4 | 2-15 | 3-5 | 3-8 | 4-9 | 4-18 | 5-12 | 6-16 | 6-19 | 6-20 | 7-13 |
| 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 3 | 0 | 0 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 4 | 1 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 |
| 5 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 6 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 0 |
| 7 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 8 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 9 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 10 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 11 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 12 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 13 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 14 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 15 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 16 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 17 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 18 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 19 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| 20 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |

Таблица 2. Матрица инциденций (продолжение)

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 7-16 | 7-19 | 7-20 | 8-19 | 8-20 | 9-12 | 10-11 | 10-16 | 11-20 | 12-17 | 14-18 | 15-20 | 17-18 | 18-19 |
| 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 3 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 5 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 6 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 7 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 8 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 9 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 10 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 11 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |
| 12 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| 13 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 14 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 15 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
| 16 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 17 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 |
| 18 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 |
| 19 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 |
| 20 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 |

Задание 2. Построить матрицу расстояний между вершинами графа

Для построения матрицы расстояний применяем алгоритм Флойда-Уоршелла. В алгоритме выполняется последовательный проход по всем парам вершин графа |V| раз. Каждый k-й проход выполняется проверка является ли расстояние через k-ую вершину меньше, чем текущее расстояние между парами. Если условие выполняется, то расстояние в ячейке пары в матрице расстояний обновляется. Первоначальная матрица представляет собой матрицу смежности, в которой нули, означающие отсутствие смежности, заменены на число большее, чем максимально возможное расстояние между двумя любыми вершинами.

Работа программы в пакете Wolfram Mathematica представлена на Рис. 1. Последняя колонка «Е» - эксцентриситет. Код и вывод представлен в Приложении 1, Задание 2.

Таблица 2. Матрица расстояний

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
| 1 | 2 | 2 | 3 | 1 | 3 | 4 | 4 | 4 | 2 | 6 | 5 | 2 | 5 | 3 | 3 | 5 | 1 | 2 | 3 | 4 |
| 2 | 2 | 2 | 1 | 1 | 2 | 4 | 3 | 2 | 2 | 4 | 3 | 3 | 4 | 3 | 1 | 4 | 3 | 2 | 3 | 2 |
| 3 | 3 | 1 | 2 | 2 | 1 | 3 | 3 | 1 | 3 | 4 | 3 | 2 | 4 | 4 | 2 | 4 | 3 | 3 | 2 | 2 |
| 4 | 1 | 1 | 2 | 2 | 3 | 3 | 3 | 3 | 1 | 5 | 4 | 2 | 4 | 2 | 2 | 4 | 2 | 1 | 2 | 3 |
| 5 | 3 | 2 | 1 | 3 | 2 | 4 | 4 | 2 | 2 | 5 | 4 | 1 | 5 | 4 | 3 | 5 | 2 | 3 | 3 | 3 |
| 6 | 4 | 3 | 3 | 3 | 4 | 2 | 2 | 2 | 4 | 2 | 2 | 4 | 3 | 3 | 2 | 1 | 3 | 2 | 1 | 1 |
| 7 | 4 | 3 | 3 | 3 | 4 | 2 | 2 | 2 | 4 | 2 | 2 | 4 | 1 | 3 | 2 | 1 | 3 | 2 | 1 | 1 |
| 8 | 4 | 2 | 1 | 3 | 2 | 2 | 2 | 2 | 4 | 3 | 2 | 3 | 3 | 3 | 2 | 3 | 3 | 2 | 1 | 1 |
| 9 | 2 | 2 | 3 | 1 | 2 | 4 | 4 | 4 | 2 | 6 | 5 | 1 | 5 | 3 | 3 | 5 | 2 | 2 | 3 | 4 |
| 10 | 6 | 4 | 4 | 5 | 5 | 2 | 2 | 3 | 6 | 2 | 1 | 6 | 3 | 5 | 3 | 1 | 5 | 4 | 3 | 2 |
| 11 | 5 | 3 | 3 | 4 | 4 | 3 | 2 | 2 | 5 | 1 | 2 | 5 | 3 | 5 | 2 | 2 | 5 | 4 | 3 | 1 |
| 12 | 2 | 3 | 2 | 2 | 1 | 4 | 4 | 3 | 1 | 6 | 5 | 2 | 5 | 3 | 4 | 5 | 1 | 2 | 3 | 4 |
| 13 | 5 | 4 | 4 | 4 | 5 | 3 | 1 | 3 | 5 | 3 | 3 | 5 | 2 | 4 | 3 | 2 | 4 | 3 | 2 | 2 |
| 14 | 3 | 3 | 4 | 2 | 4 | 3 | 3 | 3 | 3 | 5 | 5 | 3 | 4 | 2 | 4 | 4 | 2 | 1 | 2 | 4 |
| 15 | 3 | 1 | 2 | 2 | 3 | 4 | 2 | 2 | 3 | 3 | 2 | 4 | 3 | 4 | 2 | 3 | 4 | 3 | 3 | 1 |
| 16 | 5 | 4 | 4 | 4 | 5 | 1 | 1 | 3 | 5 | 1 | 2 | 5 | 2 | 4 | 3 | 2 | 4 | 3 | 2 | 2 |
| 17 | 1 | 3 | 3 | 2 | 2 | 3 | 3 | 3 | 2 | 5 | 5 | 1 | 4 | 2 | 4 | 4 | 2 | 1 | 2 | 4 |
| 18 | 2 | 2 | 3 | 1 | 3 | 2 | 2 | 2 | 2 | 4 | 4 | 2 | 3 | 1 | 3 | 3 | 1 | 2 | 1 | 3 |
| 19 | 3 | 3 | 2 | 2 | 3 | 1 | 1 | 1 | 3 | 3 | 3 | 3 | 2 | 2 | 3 | 2 | 2 | 1 | 2 | 2 |
| 20 | 4 | 2 | 2 | 3 | 3 | 3 | 1 | 1 | 4 | 2 | 1 | 4 | 2 | 4 | 1 | 2 | 4 | 3 | 2 | 2 |

A screenshot of a computer

Description automatically generated

Рис. 1

Задание 3. Найти диаметр графа, радиус графа, центр графа

Радиус, диаметр и центры графа найдены программой из задания 2 (Рис.1).

Радиус графа: 3

Диаметр графа: 6

Центр(ы) графа: {19}

Задание 4. Составить обход графа в ширину и глубину, используя в качестве начальной вершины - вершину v8

Для обхода графа в глубину использовался алгоритм с рекурсивным вызовом функции поиска для каждой вершины. Для обхода графа в ширину использовался алгоритм с циклом по каждому ярусу каждой вершины.

Работа программы в пакете Wolfram Mathematica представлена на Рис. 2. Код и вывод представлен в Приложении 1, Задание 4.

A screenshot of a social media post

Description automatically generated

Рис. 2

**Заключение**

В работе проведены операции над графами в пакете Wolfram Mathematica. Представлены скриншоты работы программы, исходные коды и ответы.

**Приложение 1**

Задание 2

In[59]:=

(\*РАСЧЕТ МАТРИЦЫ РАССТОЯНИЙ (Агоритм Флойда)\*)

Unprotect[D];

(\*Матрица смежности\*)

D={

{0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0},

{0,0,1,1,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0},

{0,1,0,0,1,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0},

{1,1,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,1,0,0},

{0,0,1,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0},

{0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,1,1},

{0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,1,0,0,1,1},

{0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1},

{0,0,0,1,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0},

{0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,1,0,0,0,0},

{0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,1},

{0,0,0,0,1,0,0,0,1,0,0,0,0,0,0,0,1,0,0,0},

{0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0},

{0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0},

{0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1},

{0,0,0,0,0,1,1,0,0,1,0,0,0,0,0,0,0,0,0,0},

{1,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,1,0,0},

{0,0,0,1,0,0,0,0,0,0,0,0,0,1,0,0,1,0,1,0},

{0,0,0,0,0,1,1,1,0,0,0,0,0,0,0,0,0,1,0,0},

{0,0,0,0,0,0,1,1,0,0,1,0,0,0,1,0,0,0,0,0}

};

nodeCount=Length[D];

max=nodeCount\*(nodeCount-1)/2+1;

For[i=1, i<=nodeCount,i++,

For[j=1, j<=nodeCount,j++,

If[D[[i]][[j]]==0,D[[i]][[j]]=max];

]

]

radius=max;

diameter=0;

centers={};

For[k=1, k<=nodeCount,k++,

For[i=1, i<=nodeCount,i++,

For[j=1, j<=nodeCount,j++,

D[[i]][[j]]=Min[D[[i]][[j]],D[[i]][[k]]+D[[k]][[j]]];

]

If[k==nodeCount, eps=Max[D[[i]]];

AppendTo[D[[i]],eps];

If[radius==eps,AppendTo[centers,i];];

diameter=Max[eps, diameter];

If[radius>eps,

radius = eps;

centers={};

AppendTo[centers,i];

];

];

]

]

header={""};

For[i=1,i<=nodeCount,i++,

PrependTo[D[[i]],i];

AppendTo[header,i];

]

AppendTo[header,"\[CapitalEpsilon]"];

PrependTo[D,header];

Print["Матрица расстояний"];

TableForm[D]

Print["Радиус графа:"];

radius

Print["Диаметр графа:"];

diameter

Print["Центр(ы) графа:"];

centers

During evaluation of In[59]:= Матрица расстояний

Out[69]//TableForm=

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | E |
| 1 | 2 | 2 | 3 | 1 | 3 | 4 | 4 | 4 | 2 | 6 | 5 | 2 | 5 | 3 | 3 | 5 | 1 | 2 | 3 | 4 | 6 |
| 2 | 2 | 2 | 1 | 1 | 2 | 4 | 3 | 2 | 2 | 4 | 3 | 3 | 4 | 3 | 1 | 4 | 3 | 2 | 3 | 2 | 4 |
| 3 | 3 | 1 | 2 | 2 | 1 | 3 | 3 | 1 | 3 | 4 | 3 | 2 | 4 | 4 | 2 | 4 | 3 | 3 | 2 | 2 | 4 |
| 4 | 1 | 1 | 2 | 2 | 3 | 3 | 3 | 3 | 1 | 5 | 4 | 2 | 4 | 2 | 2 | 4 | 2 | 1 | 2 | 3 | 5 |
| 5 | 3 | 2 | 1 | 3 | 2 | 4 | 4 | 2 | 2 | 5 | 4 | 1 | 5 | 4 | 3 | 5 | 2 | 3 | 3 | 3 | 5 |
| 6 | 4 | 3 | 3 | 3 | 4 | 2 | 2 | 2 | 4 | 2 | 2 | 4 | 3 | 3 | 2 | 1 | 3 | 2 | 1 | 1 | 4 |
| 7 | 4 | 3 | 3 | 3 | 4 | 2 | 2 | 2 | 4 | 2 | 2 | 4 | 1 | 3 | 2 | 1 | 3 | 2 | 1 | 1 | 4 |
| 8 | 4 | 2 | 1 | 3 | 2 | 2 | 2 | 2 | 4 | 3 | 2 | 3 | 3 | 3 | 2 | 3 | 3 | 2 | 1 | 1 | 4 |
| 9 | 2 | 2 | 3 | 1 | 2 | 4 | 4 | 4 | 2 | 6 | 5 | 1 | 5 | 3 | 3 | 5 | 2 | 2 | 3 | 4 | 6 |
| 10 | 6 | 4 | 4 | 5 | 5 | 2 | 2 | 3 | 6 | 2 | 1 | 6 | 3 | 5 | 3 | 1 | 5 | 4 | 3 | 2 | 6 |
| 11 | 5 | 3 | 3 | 4 | 4 | 3 | 2 | 2 | 5 | 1 | 2 | 5 | 3 | 5 | 2 | 2 | 5 | 4 | 3 | 1 | 5 |
| 12 | 2 | 3 | 2 | 2 | 1 | 4 | 4 | 3 | 1 | 6 | 5 | 2 | 5 | 3 | 4 | 5 | 1 | 2 | 3 | 4 | 6 |
| 13 | 5 | 4 | 4 | 4 | 5 | 3 | 1 | 3 | 5 | 3 | 3 | 5 | 2 | 4 | 3 | 2 | 4 | 3 | 2 | 2 | 5 |
| 14 | 3 | 3 | 4 | 2 | 4 | 3 | 3 | 3 | 3 | 5 | 5 | 3 | 4 | 2 | 4 | 4 | 2 | 1 | 2 | 4 | 5 |
| 15 | 3 | 1 | 2 | 2 | 3 | 4 | 2 | 2 | 3 | 3 | 2 | 4 | 3 | 4 | 2 | 3 | 4 | 3 | 3 | 1 | 4 |
| 16 | 5 | 4 | 4 | 4 | 5 | 1 | 1 | 3 | 5 | 1 | 2 | 5 | 2 | 4 | 3 | 2 | 4 | 3 | 2 | 2 | 5 |
| 17 | 1 | 3 | 3 | 2 | 2 | 3 | 3 | 3 | 2 | 5 | 5 | 1 | 4 | 2 | 4 | 4 | 2 | 1 | 2 | 4 | 5 |
| 18 | 2 | 2 | 3 | 1 | 3 | 2 | 2 | 2 | 2 | 4 | 4 | 2 | 3 | 1 | 3 | 3 | 1 | 2 | 1 | 3 | 4 |
| 19 | 3 | 3 | 2 | 2 | 3 | 1 | 1 | 1 | 3 | 3 | 3 | 3 | 2 | 2 | 3 | 2 | 2 | 1 | 2 | 2 | 3 |
| 20 | 4 | 2 | 2 | 3 | 3 | 3 | 1 | 1 | 4 | 2 | 1 | 4 | 2 | 4 | 1 | 2 | 4 | 3 | 2 | 2 | 4 |

During evaluation of In[314]:= Радиус графа:

Out[330]= 3

During evaluation of In[314]:= Диаметр графа:

Out[332]= 6

During evaluation of In[314]:= Центр(ы) графа:

Out[334]= {19}

Задание 3

Unprotect[D];

(\*матрица смежности\*)

D={

{0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,0},

{0,0,1,1,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0},

{0,1,0,0,1,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0},

{1,1,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,1,0,0},

{0,0,1,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0},

{0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,1,1},

{0,0,0,0,0,0,0,0,0,0,0,0,1,0,0,1,0,0,1,1},

{0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,1},

{0,0,0,1,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0},

{0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,1,0,0,0,0},

{0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,1},

{0,0,0,0,1,0,0,0,1,0,0,0,0,0,0,0,1,0,0,0},

{0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0},

{0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0},

{0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1},

{0,0,0,0,0,1,1,0,0,1,0,0,0,0,0,0,0,0,0,0},

{1,0,0,0,0,0,0,0,0,0,0,1,0,0,0,0,0,1,0,0},

{0,0,0,1,0,0,0,0,0,0,0,0,0,1,0,0,1,0,1,0},

{0,0,0,0,0,1,1,1,0,0,0,0,0,0,0,0,0,1,0,0},

{0,0,0,0,0,0,1,1,0,0,1,0,0,0,1,0,0,0,0,0}

};

vertexCount=Length[D];

G=UndirectedGraph[AdjacencyGraph[D],VertexLabels->Table[i->Subscript["",i],{i,vertexCount}]];

GetAdjacencyList[v\_]:=

(

adjacencyList={};

For[i=1,i<=vertexCount,i++,

If[D[[v]][[i]]!=0,AppendTo[adjacencyList,i]]];

Return[adjacencyList];

)

DepthFirstSearch[vList\_List:{},v\_]:=Module[{i,adjacencyList},

locList={};

locList=vList;

AppendTo[locList,v];

adjacencyList=GetAdjacencyList[v];

For[i=1,i<=Length[adjacencyList],i++,

If[MemberQ[locList,adjacencyList[[i]]],Continue[]];

locList=DepthFirstSearch[locList,adjacencyList[[i]]];

];

Return[locList];

]

BreadthFirstSearch[v\_]:=Module[{i,adjacencyList},

vList={};

AppendTo[vList,v];

For[i=1,i<=Length[vList],i++,

adjacencyList=GetAdjacencyList[vList[[i]]];

For[j=1,j<=Length[adjacencyList],j++,

If[!MemberQ[vList,adjacencyList[[j]]],AppendTo[vList,adjacencyList[[j]]]];

];

];

Return[vList];

]

G=UndirectedGraph[AdjacencyGraph[D],VertexLabels->Table[i->Subscript["",i],{i,vertexCount}]];

startVertex=8;

dfsBypassList={};

dfsBypassList=DepthFirstSearch[startVertex];

bfsBypassList={};

bfsBypassList=BreadthFirstSearch[startVertex];

G

Print["Проход в глубину: ",dfsBypassList];

Print["Проход в ширину: ",bfsBypassList];

Проход в глубину: {8,3,2,4,1,17,12,5,9,18,14,19,6,16,7,13,20,11,10,15}

Проход в ширину: {8,3,19,20,2,5,6,7,18,11,15,4,12,16,13,14,17,10,1,9}

**Приложение 2**

Ссылка на проект

[https://](https://github.com/petrovicheugene/MobAppLabs.git)github.com/[petrovicheugene](https://github.com/petrovicheugene/DiscreteMathAndGraphs.git)/[DiscreteMathAndGraphs](https://github.com/petrovicheugene/DiscreteMathAndGraphs.git).git