

Министерство образования Республики Беларусь  
Учреждение образования  
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет Информационных технологий и управления  
Кафедра Интеллектуальных информационных технологий

**РАСЧЕТНАЯ РАБОТА**  
по дисциплине «Традиционные и интеллектуальные информационные технологии»  
на тему  
**Найти количество хорд неориентированного графа**

Выполнила:

И. Д. Пикта

Студент группы  
321702

Проверил:

Н. В. Малиновская

Минск 2024

# 1 Введение

**Цель:** Получить навыки формализации и обработки информации с использованием семантических сетей

**Задача:** Найти количество хорд неориентированного графа.

## 2 Список понятий

1. **Граф** (Рис.1) (абсолютное понятие) - совокупность непустого множества вершин и наборов пар вершин (связей между вершинами).

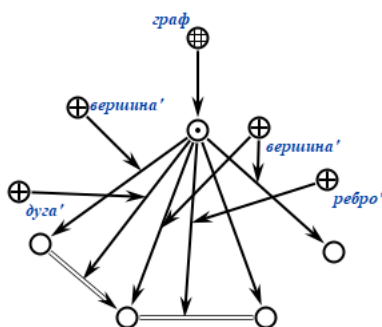


Рис. 1: Абсолютное понятие Графа

2. **Неориентированный граф** (Рис.2) (абсолютное понятие) – граф, в котором все связи-ребра.

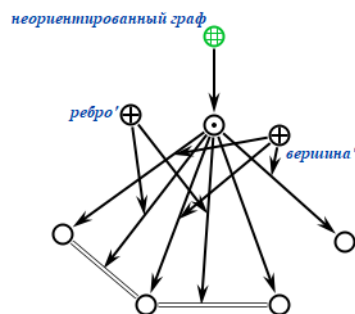


Рис. 2: Неориентированный граф

3. **Степень вершины** (Рис.3) (абсолютное понятие) – это число рёбер, инцидентных данной вершине.

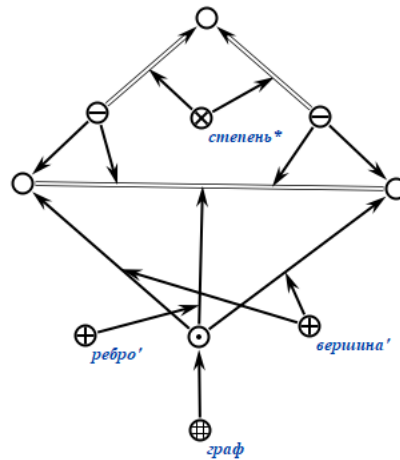


Рис. 3: Степень вершины

4. **(Реберно)Регулярный** (Рис.4) - (абсолютное понятие) – это граф, степени всех вершин которого равны, то есть каждая вершина имеет одинаковое количество соседей.

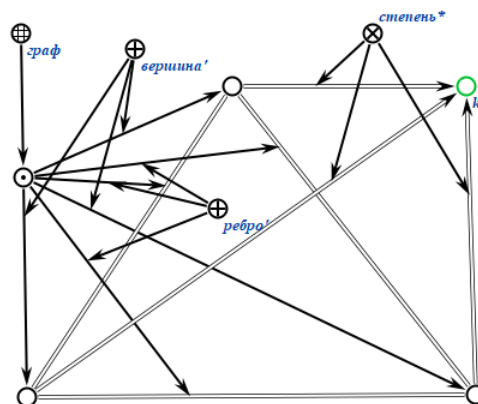


Рис. 4: Неориентированный граф

### 3 Тестовые примеры

Во всех тестах графы будут приведены в сокращенной форме со скрытыми ролями элементов графа.

#### 3.1 Тест 1 (Рис.5, Рис.6)

**Вход:** Необходимо определить, является ли граф регулярным.

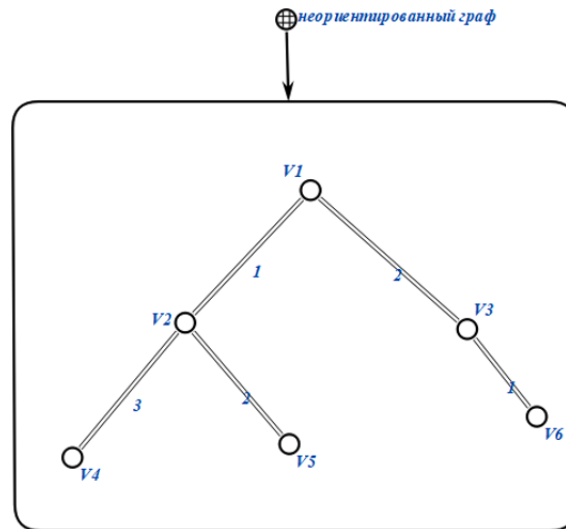


Рис. 5: Вход теста 1

**Выход:** Да, граф является регулярным.

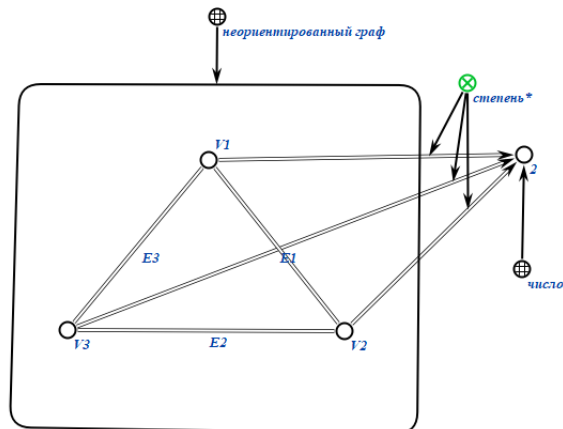


Рис. 6: Вход теста 1

### 3.2 Тест 2 (Рис.7, Рис.8)

**Вход:** Необходимо определить, является ли граф регулярным.

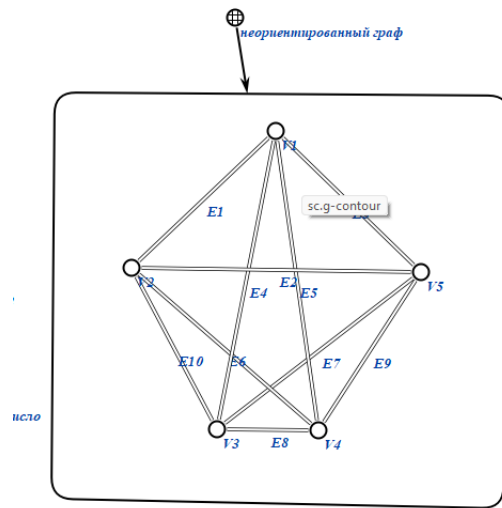


Рис. 7: Вход теста 2

**Выход:** Да, граф является регулярным.

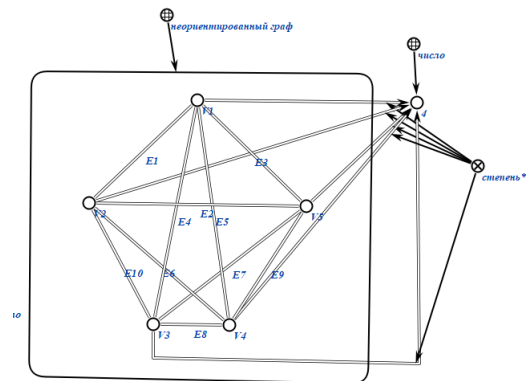


Рис. 8: Вход теста 2

### 3.3 Тест 3 (Рис.9, Рис.10)

**Вход:** Необходимо определить, является ли граф регулярным.

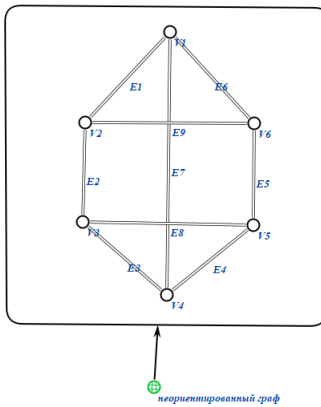


Рис. 9: Вход теста 3

**Выход:** Да, граф является регулярным.

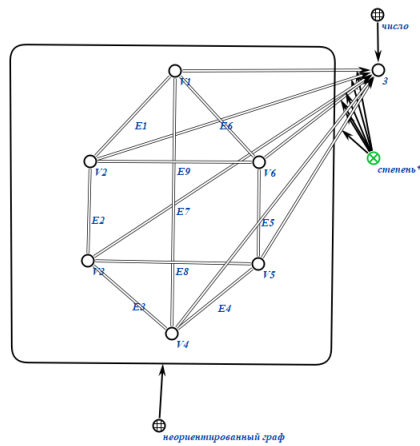


Рис. 10: Вход теста 3

### 3.4 Тест 4 (Рис.11, Рис.12)

**Вход:** Необходимо определить, является ли граф регулярным.

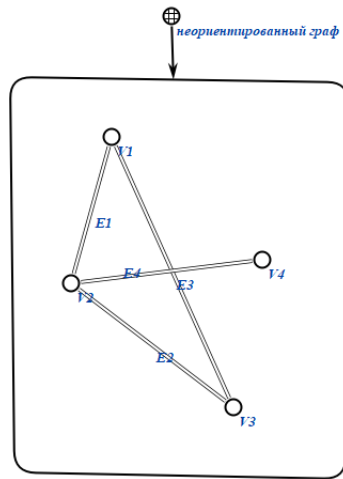


Рис. 11: Вход теста 4

**Выход:** Нет, граф не является регулярным, т.к. его вершины имеют разную степень.

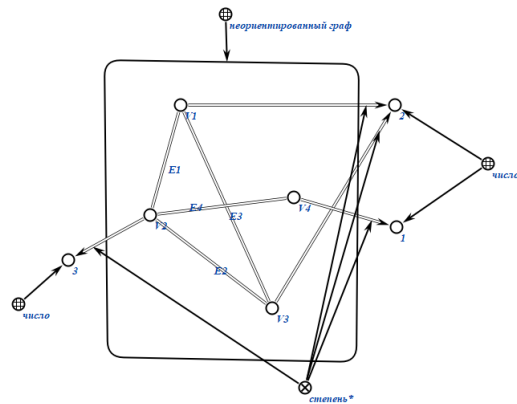


Рис. 12: Вход теста 4

### 3.5 Тест 5 (Рис.13, Рис.14)

**Вход:** Необходимо определить, является ли граф регулярным.

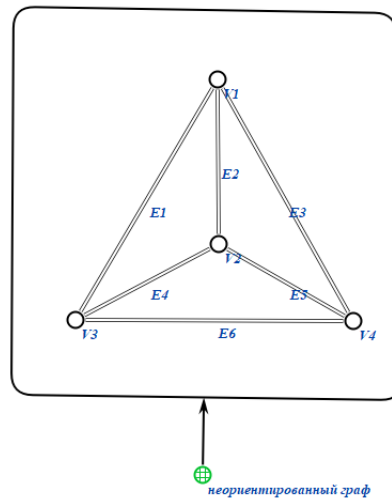


Рис. 13: Вход теста 4

**Выход:** Да, граф является регулярным.

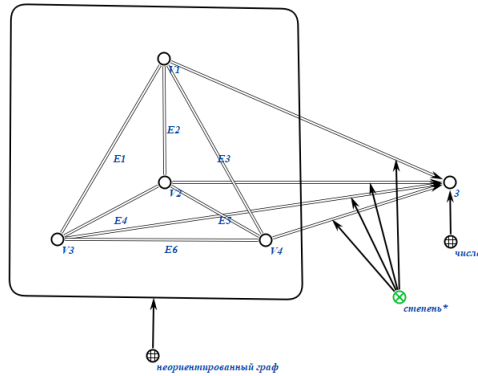


Рис. 14: Вход теста 4



## 4 Пример работы алгоритма в семантической памяти

### 4.1 Краткое описание:

1. Создаем список, который будет хранить степень каждой вершины.
2. Проверяем степень первой вершины и полученное значение записываем в список.
3. Проверяем степень следующей вершины и полученное значение записываем в список. Повторяем пункт 3 до обозревания всех вершин.
4. Если пройдены все вершины, то переходим к проверки на эквивалентность степеней вершин из полученного списка.
5. Если степени всех вершин графа равны, то делаем вывод, что граф является регулярным. В противном случае граф не является регулярным, сообщаем об этом пользователю

### 4.2 Демонстрация на тесте 5:

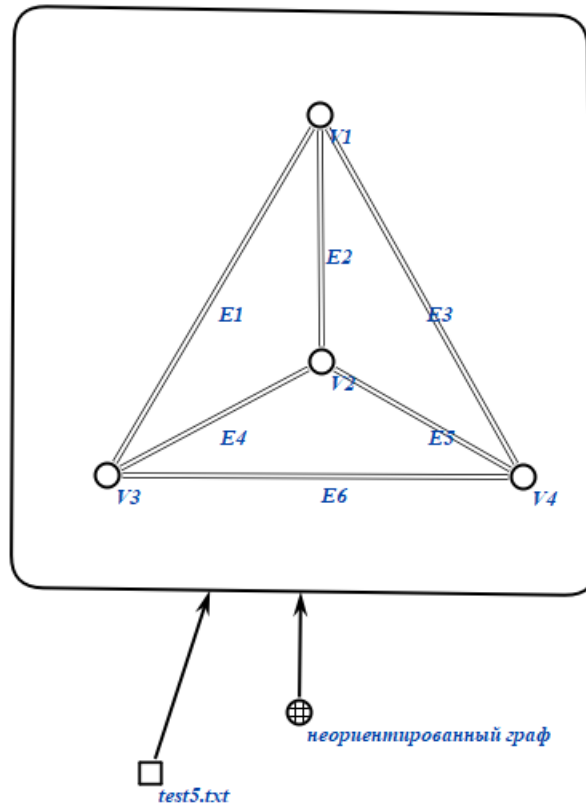


Рис. 15: Вход теста 5.

1. `_graph` получит в качестве значения sc-узел неориентированного графа;

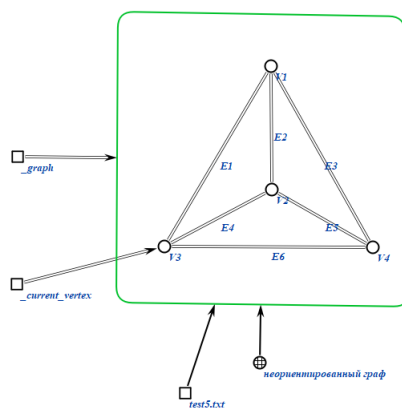


Рис. 16: Действие 1.

2. Создается список степеней вершин, которая сразу включает степень первой вершины `V3`.

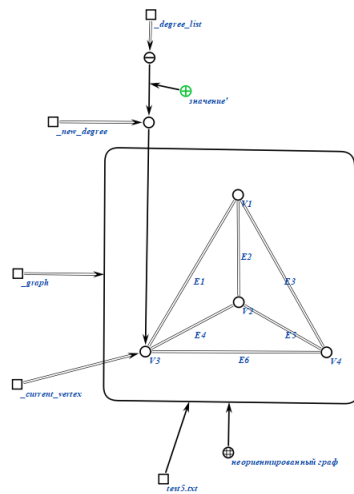


Рис. 17: Действие 2.

3. На этом этапе добавляем в список степень вершины  $V1$ .

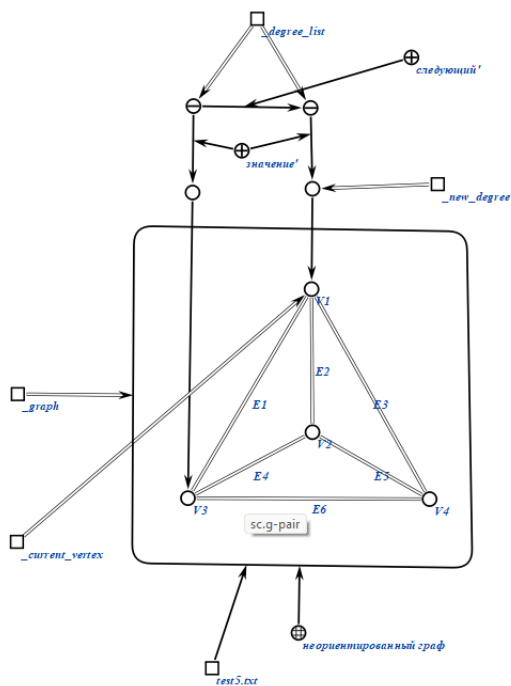


Рис. 18: Действие 3.

4. На этом этапе добавляем в список степень вершины  $V2$

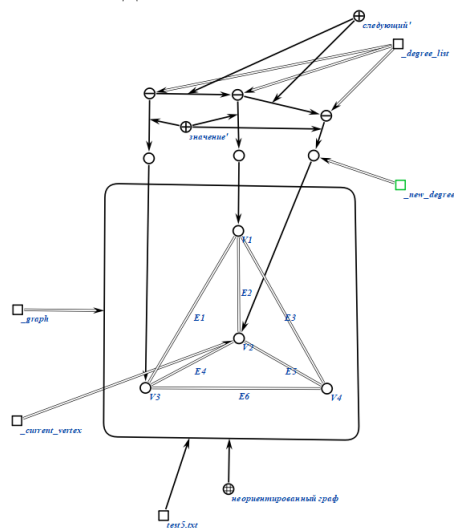


Рис. 19: Действие 4.

5. На этом этапе добавляем в список степеней вершины  $V_4$

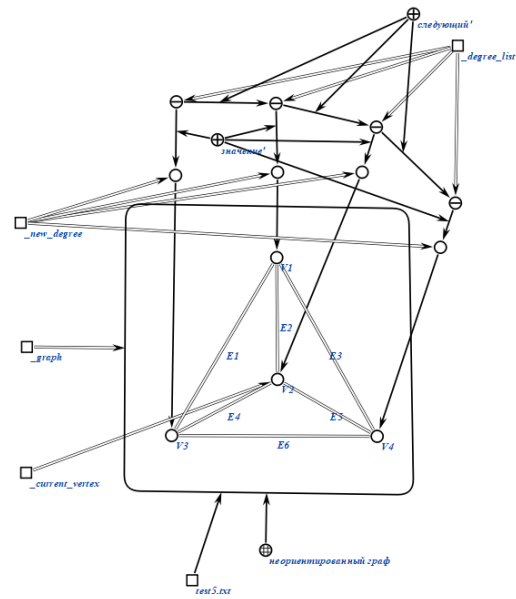


Рис. 20: Действие 5.

6. Выводим полученный список степеней вершин графа для дальнейшей проверки.

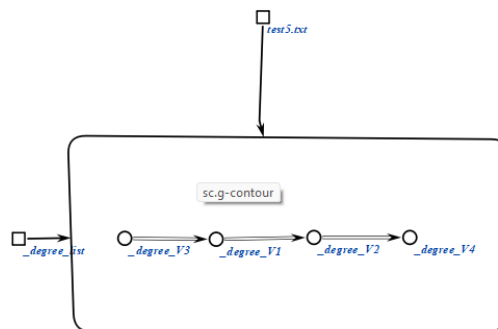


Рис. 21: Действие 6.

7. Сравниваем степени вершин графа - и делаем вывод, что они все равны.

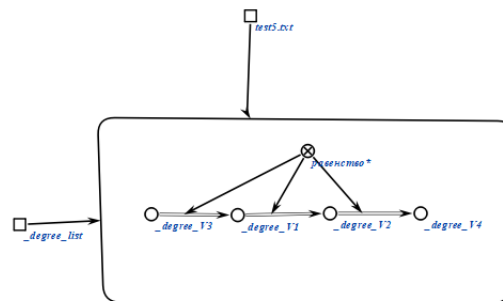


Рис. 22: Действие 7.

8. На основе того, что все вершины имеют одинаковую степень, то, исходя из определения, можем сделать вывод, что данный граф является регулярным.

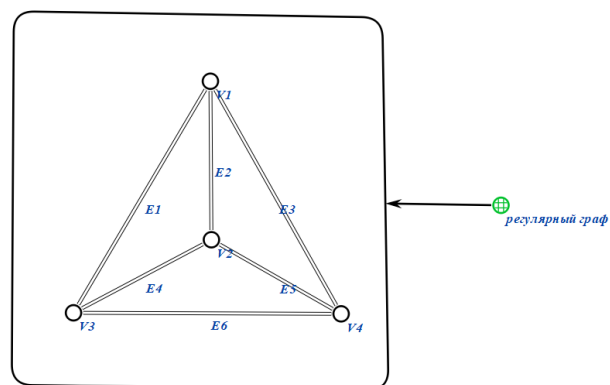


Рис. 23: Действие 8.

## 5 Заключение

В рамках данной лабораторной работы был разработан алгоритм проверки неориентированного графа на регулярность. Для начала, необходимо выбрать произвольную вершину в графе в качестве стартовой. Далее, осуществляется проверка степени каждой вершины, то есть количества ребер, связанных с данной вершиной. Если все вершины имеют одинаковую степень, граф считается регулярным.

Однако, для корректной работы алгоритма, необходимо учесть, что граф должен быть связным, то есть должен существовать путь между любыми двумя вершинами. Если граф не является связным, алгоритм может быть применен для каждой компоненты связности отдельно.

Таким образом, предложенный алгоритм позволяет эффективно проверить регулярность неориентированного графа, основываясь на сравнении степеней его вершин. Он может быть полезен для анализа и классификации графов в различных приложениях, таких как социальные сети, транспортные сети, и другие сетевые структуры.