

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 «Представление и обработка информации в интеллектуальных системах»  
на тему  
**Найти граф замыкания неориентированного графа.**

Выполнил:

Е. В. Пшенов

Студент группы  
321702

Проверил:

Н. В. Малиновская

Минск 2024

# Содержание

<b>1</b>	<b>Введение</b>	<b>2</b>
<b>2</b>	<b>Список понятий</b>	<b>2</b>
<b>3</b>	<b>Тестовые примеры</b>	<b>4</b>
3.1	Тест 1 . . . . .	4
3.2	Тест 2 . . . . .	5
3.3	Тест 3 . . . . .	6
3.4	Тест 4 . . . . .	7
<b>4</b>	<b>Пример работы алгоритма в семантической памяти</b>	<b>8</b>
4.1	Краткое описание: . . . . .	8
4.2	Демонстрация на тесте 5: . . . . .	9
<b>5</b>	<b>Заключение</b>	<b>15</b>

# 1 Введение

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

**Задача:** Найти граф замыкания неориентированного графа.

## 2 Список понятий

1. **Неориентированный граф** (абсолютное понятие)-граф, в котором все ребра являются звеньями, то есть порядок двух концов ребра графа не существует

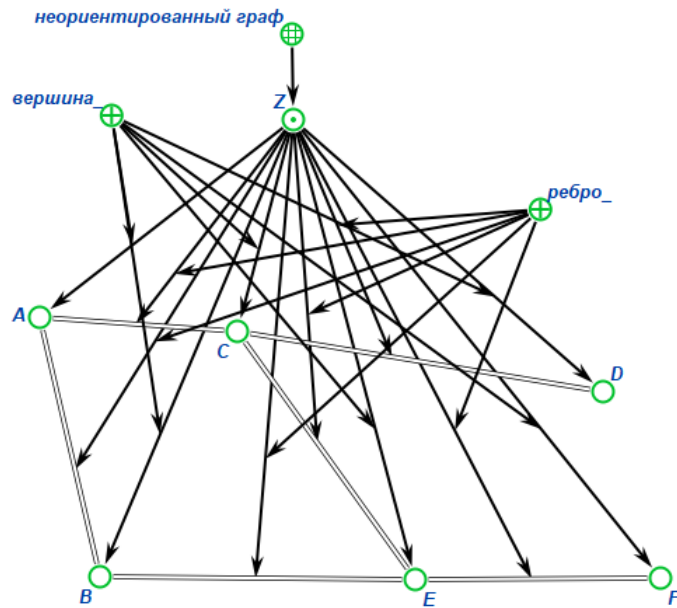


Рис. 1: Абсолютное понятие неориентированного графа

2. **Замыканием** графа называется такой набор рёбер, который содержит все возможные пути между всеми парами вершин.

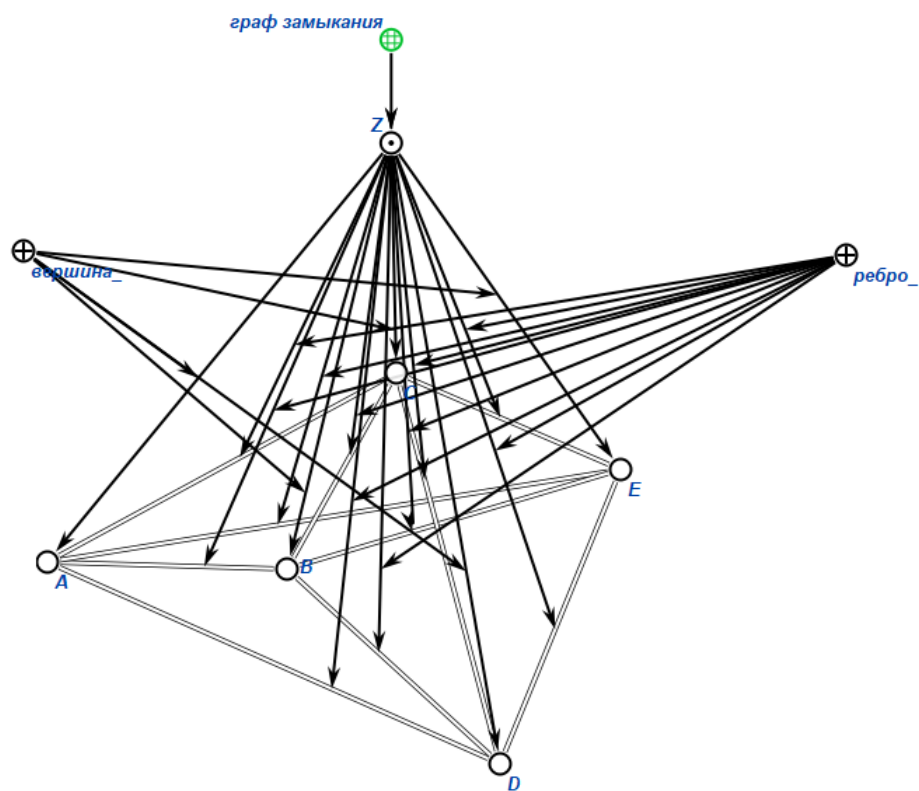


Рис. 2: Абсолютное понятие графа замыкания

### 3 Тестовые примеры

#### 3.1 Тест 1

**Вход:** Найти граф замыкания данного неориентированного графа.

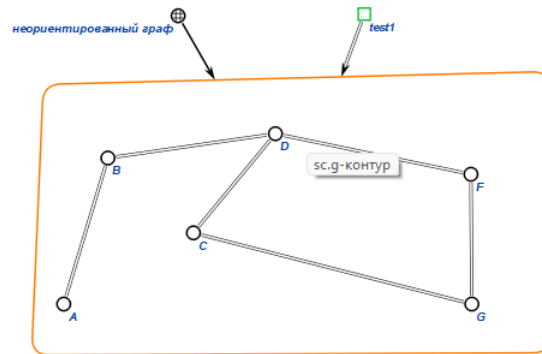


Рис. 3: Вход теста 1

**Выход:** Будет найдено замыкание заданного неориентированного графа.

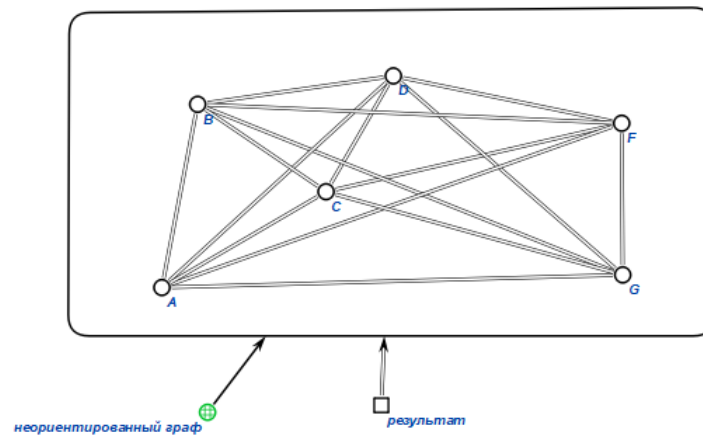


Рис. 4: Выход теста 1

## 3.2 Тест 2

**Вход:** Найти граф замыкания данного неориентированного графа.

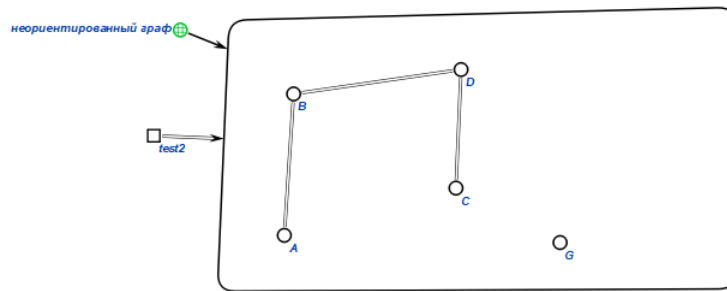


Рис. 5: Вход теста 2

**Выход:** Будет найдено замыкание заданного неориентированного графа.

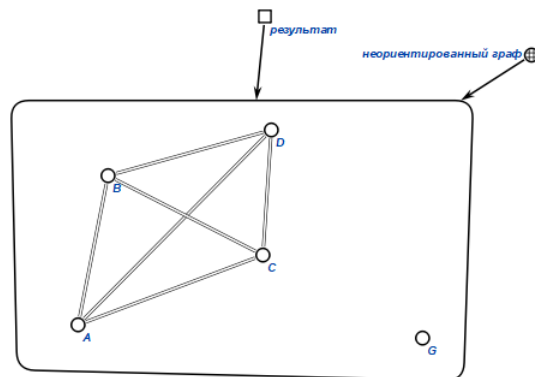


Рис. 6: Выход теста 2

### 3.3 Тест 3

**Вход:** Найти граф замыкания данного неориентированного графа.

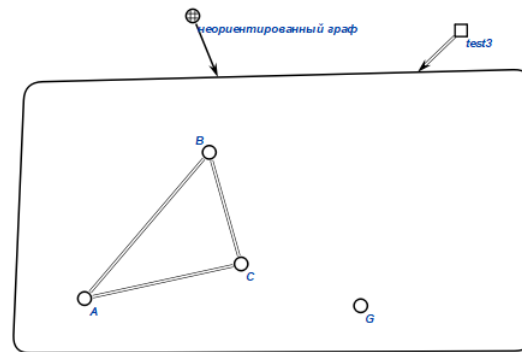


Рис. 7: Вход теста 3

**Выход:** Будет найдено замыкание заданного неориентированного графа.

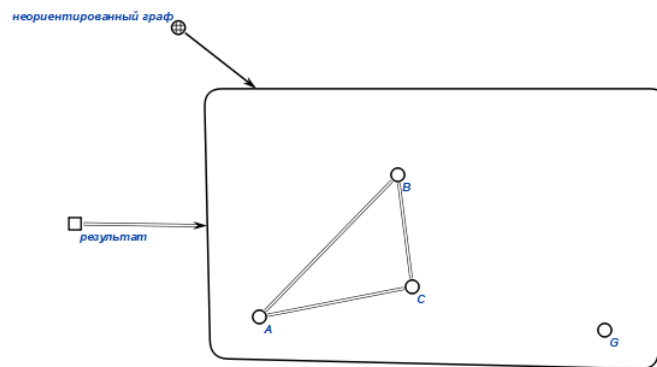


Рис. 8: Выход теста 3

### 3.4 Тест 4

**Вход:** Найти граф замыкания данного неориентированного графа.

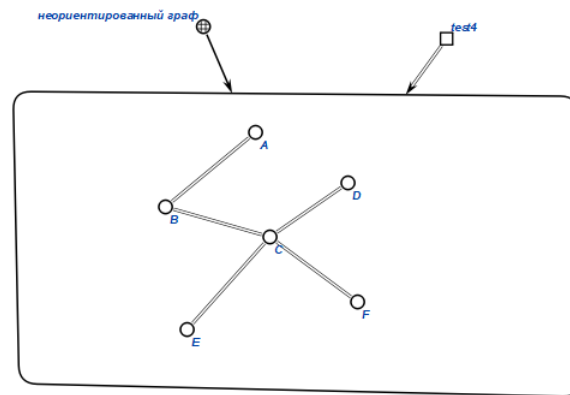


Рис. 9: Вход теста 4

**Выход:** Будет найдено замыкание заданного неориентированного графа.

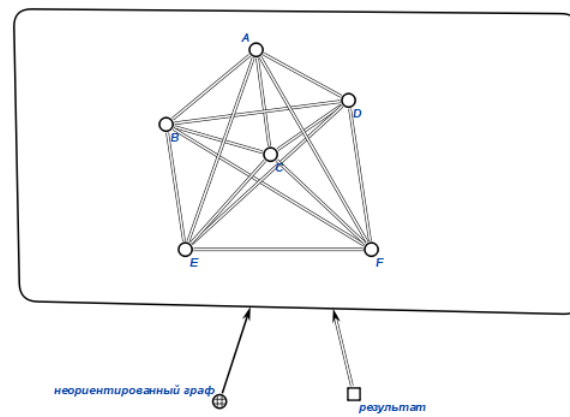


Рис. 10: Выход теста 4



## 4 Пример работы алгоритма в семантической памяти

### 4.1 Краткое описание:

1. Создание неориентированного графа(рис.11).
2. Создаём переменную **Count V**, которая будет хранить в себе количество вершин, имеющих хотя бы 1 ребро (рис.12).
3. Задаём начальную точку вхождения в граф и конечную (рис.13).
4. Создание всех возможных рёбер для вершины А (рис.14).
5. Просмотр добавленных рёбер для вершины А (рис.15).
6. Создание всех возможных рёбер для вершины В (рис. 16).
7. Просмотр добавленных рёбер для вершины В (рис.17).
8. Создание всех возможных рёбер для вершины С (рис.18).
9. Просмотр добавленных рёбер для вершины С (рис.19).
10. Создание всех возможных рёбер для вершины D
11. Просмотр добавленных рёбер для вершины D
12. Создание всех возможных рёбер для вершины E
13. Просмотр добавленных рёбер для вершины E
14. Создание всех возможных рёбер для вершины F
15. Просмотр добавленных рёбер для вершины F
16. Создаём переменную **Count E**, которая будет хранить в себе количество всех рёбер конечного графа (рис.20).
17. Проверка на замыкание вершин конечного графа (рис.21).
18. Выводим полученный результат (рис.22).

## 4.2 Демонстрация на тесте 5:

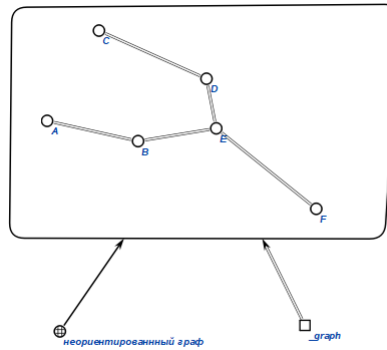


Рис. 11: Вход теста 5. Действие 1

1. Создаём переменную **Count V**, которая будет хранить в себе количество вершин, имеющих хотя бы 1 ребро. Если вершина не будет иметь ребро, в последующем алгоритме она использоваться не будет;

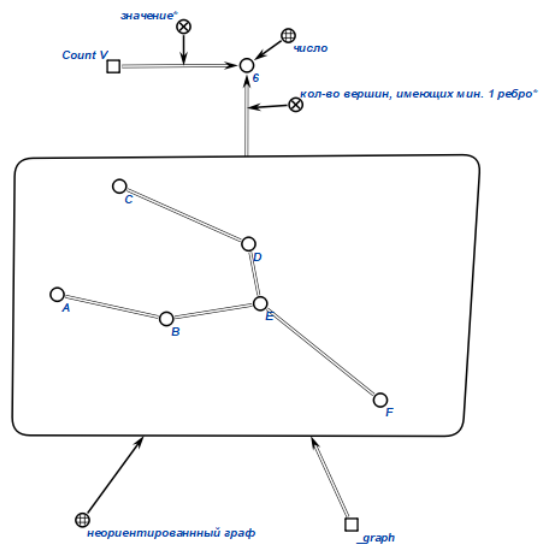


Рис. 12: Действие 2

2. Задаём начальную точку вхождения в граф и конечную, игнорируя вершины без рёбер;

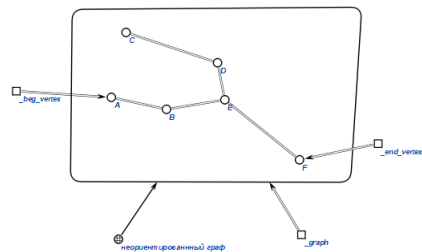


Рис. 13: Действие 3. Задаём начальную точку вхождения в граф и конечную

3. На данном этапе создаём все возможные рёбра для начальной вершины, то есть вершины A;

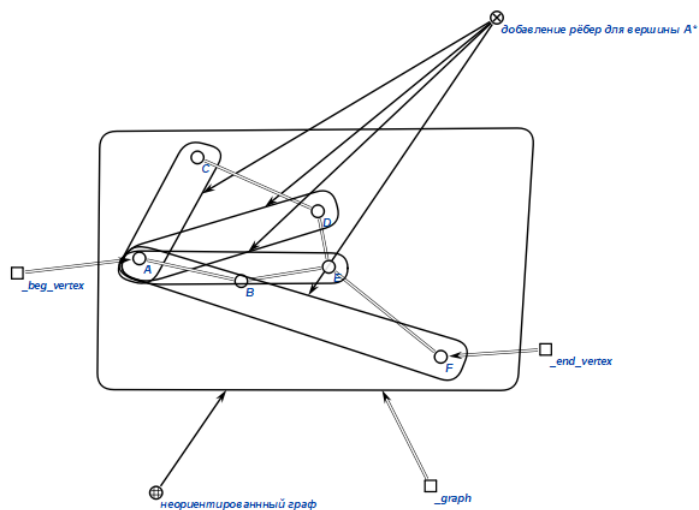


Рис. 14: Действие 4. Создание всех возможных рёбер для вершины A

4. После создания всех рёбер для вершины A просматриваем граф полностью ;

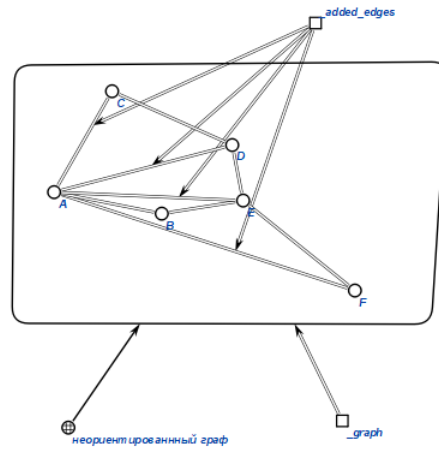


Рис. 15: Действие 5. Просмотр добавленных рёбер для вершины A

5. Продолжаем так до тех пор, пока не дойдём до конечной вершины, то есть до вершины F. Ниже представлены подробные пути перехода от одного действия к другому;

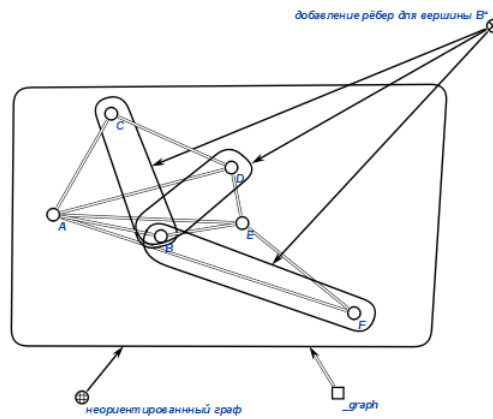


Рис. 16: Действие 6.Создание всех возможных рёбер для вершины B

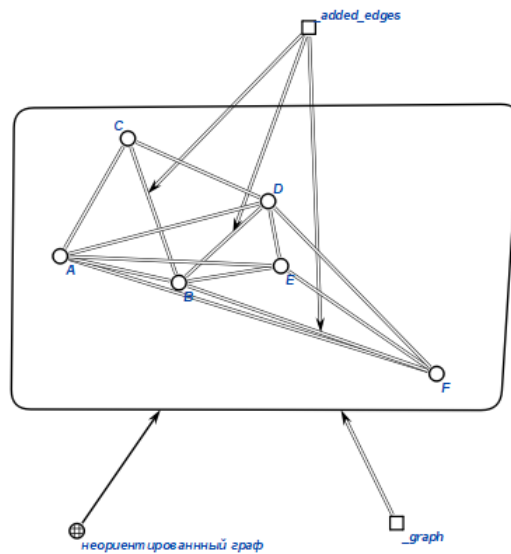


Рис. 17: Действие 7. Просмотр добавленных рёбер для вершины B

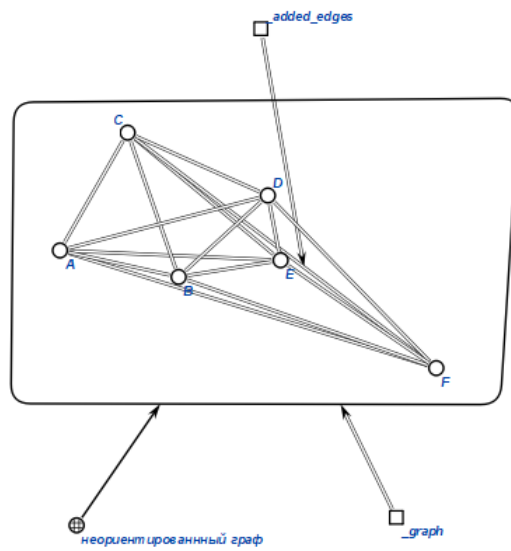


Рис. 18: Действие 8. Создание всех возможных рёбер для вершины C

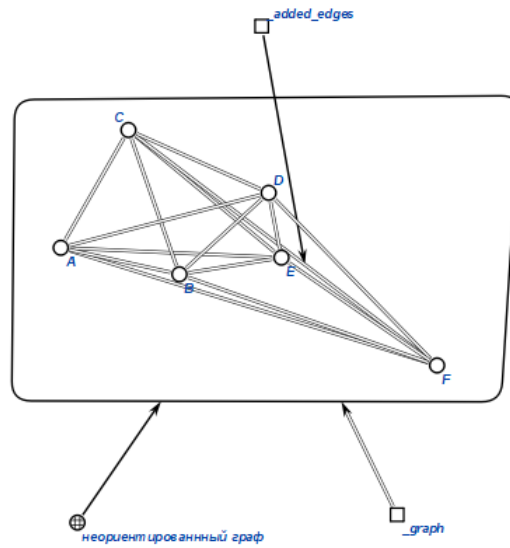


Рис. 19: Действие 9. Просмотр добавленных рёбер для вершины C

6. Так как после вершины C добавлять рёбра некуда из-за уже заполненного графа, то перейдём к созданию переменной **Count E**, которая будет хранить в себе количество всех рёбер конечного графа.

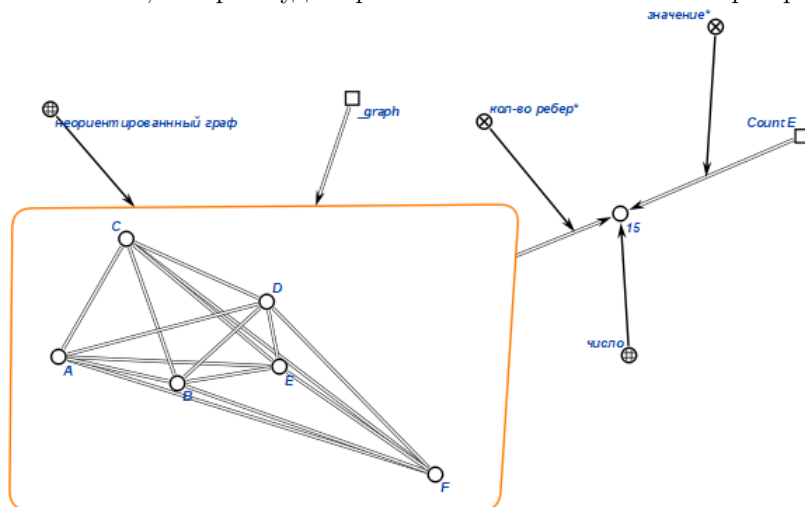


Рис. 20: Действие 10-16

7. Для того, чтобы правильно получить граф замыкания, необходимо, чтобы конечное количество рёбер было верным. Для этого воспользуемся формулой нахождения максимального количества рёбер для данного графа с количеством вершин  $\text{Count } E = \text{Count } V (\text{Count } v - 1)/2$

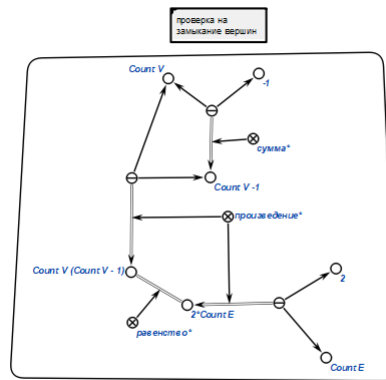


Рис. 21: Действие 17. Проверка на замыкание вершин конечного графа

8. Проверку на замыкание вершин наш граф прошёл, а это значит, что мы можем спокойно получить правильный граф по итогу выполненного алгоритма.

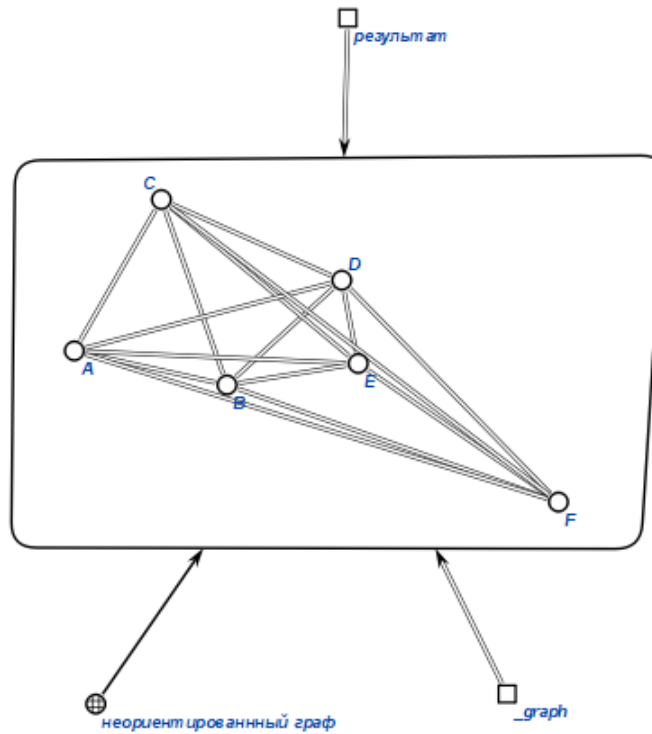


Рис. 22: Действие 18. Выводим полученный результат

## 5 Заключение

В рамках данной темы были рассмотрены ключевые аспекты формализации и обработки информации с использованием семантических сетей, а также задача нахождения графа замыкания для неориентированного графа. Освоение навыков формализации и обработки информации с помощью семантических сетей, а также методов работы с графами, является важным направлением в развитии компетенций, связанных с управлением и анализом данных. Полученные знания и умения могут найти применение в широком спектре прикладных задач, от информационных систем до интеллектуального анализа данных.



## Список литературы

- [1] Ф. Харарри. *Теория графов*. Эдиториал УРСС, Москва, 2018. 304 с.
- [2] Кофман А. *Введение в теорию нечетких множеств*. Радио и связь, Москва, 1982. 423с.
- [3] Нечипуренко, М. И. *Алгоритмы и программы решения задач на графах и сетях*. Наука. Сиб. отд-ние, Новосибирск, 1990. – 515 с.
- [4] Д. Кормен. *Алгоритмы. Построение и анализ*. Вильямс, Москва, 2015. 1328 с.
- [5] M. Wooldridge. *An introduction to multiagent systems*. 2nd ed. J. Wiley, Chichester, 2009. 484 p.