

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 минимальную степень ребра в неориентированном графе.

Выполнил:

В. Д. Семеняко

Студент группы
321701

Проверила:

Н. В. Малиновская

Минск 2024

Содержание

1	Введение	2
2	Список понятий	2
3	Тестовые примеры	3
3.1	Тест 1	3
3.2	Тест 2	4
3.3	Тест 3	5
3.4	Тест 4	6
4	Пример работы алгоритма в семантической памяти	7
4.1	Краткое описание:	7
4.2	Демонстрация на тесте 5:	7
5	Заключение	10
6	Список использованных источников	11

1 Введение

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

Задача: Найти минимальную степень ребра в неориентированном графе.

2 Список понятий

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

- (a) вершина (относительное понятие, ролевое отношение);
- (b) связка (относительное понятие, ролевое отношение).

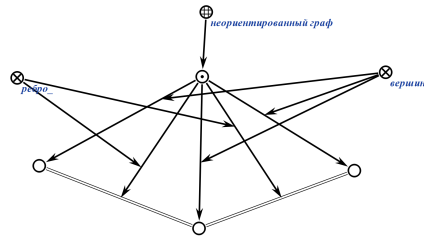


Рис. 1: Абсолютное понятие неориентированного графа

2. **Степень вершины** — это количество рёбер, инцидентных (подключённых) к этой вершине. В неориентированном графе степень вершины — это количество вершин, с которыми она связана.

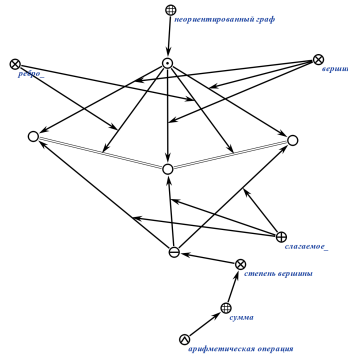


Рис. 2: Абсолютное понятие степени вершины

3. **Степень ребра** в неориентированном графе — это количество рёбер, инцидентных вершинам, которые это ребро соединяет.

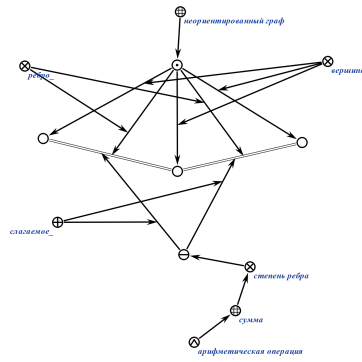


Рис. 3: Абсолютное понятие степени ребра

3 Тестовые примеры

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

3.1 Тест 1

Вход:

Необходимо найти минимальное значение степени ребра неориентированного графа.

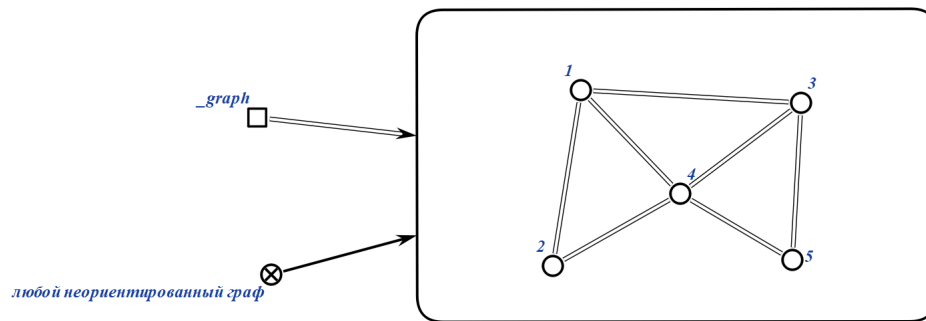


Рис. 4: Вход теста 1

Выход: Будет найдено значение: минимальная степень ребра неориентированного графа.

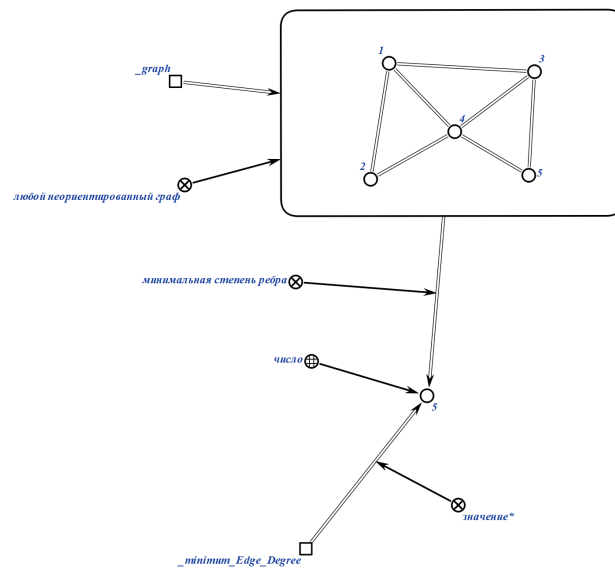


Рис. 5: Выход теста 1

3.2 Тест 2

Вход: Необходимо найти минимальное значение степени ребра неориентированного графа.

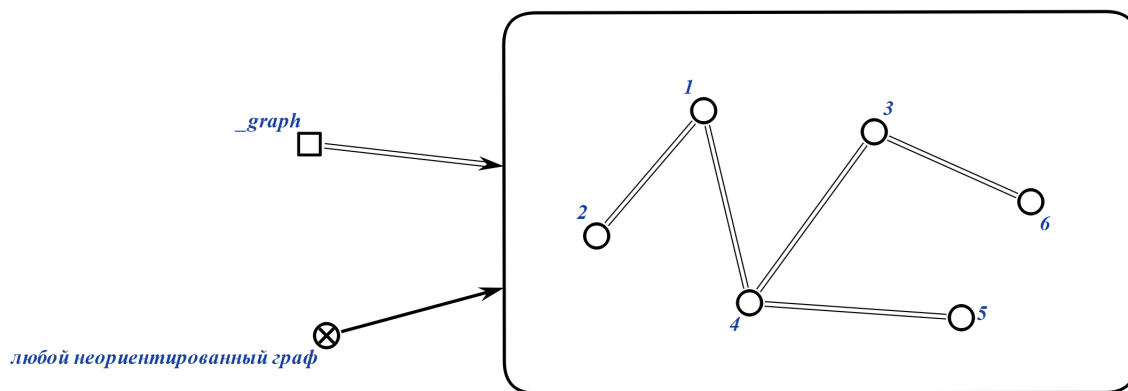


Рис. 6: Вход теста 2

Выход: Будет найдено значение: минимальная степень ребра неориентированного графа.

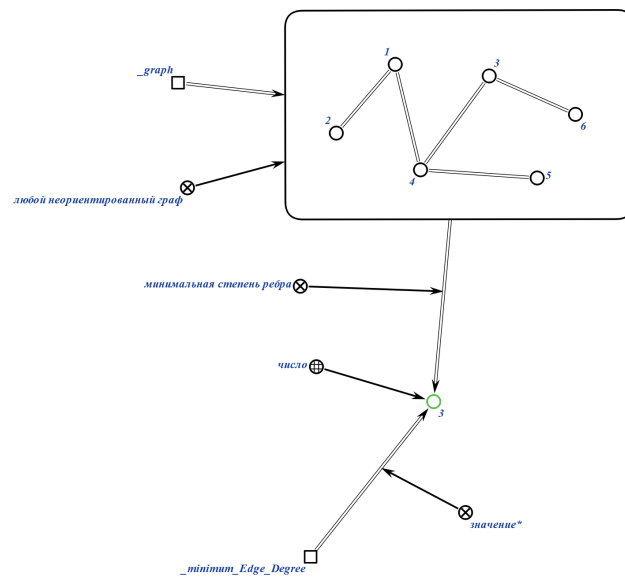


Рис. 7: Выход теста 2

3.3 Тест 3

Вход: Необходимо найти минимальное значение степени ребра неориентированного графа.

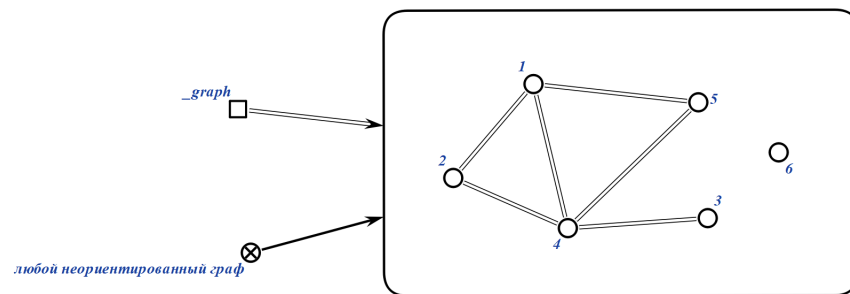


Рис. 8: Вход теста 3

Выход: Будет найдено значение: минимальная степень ребра неориентированного графа.

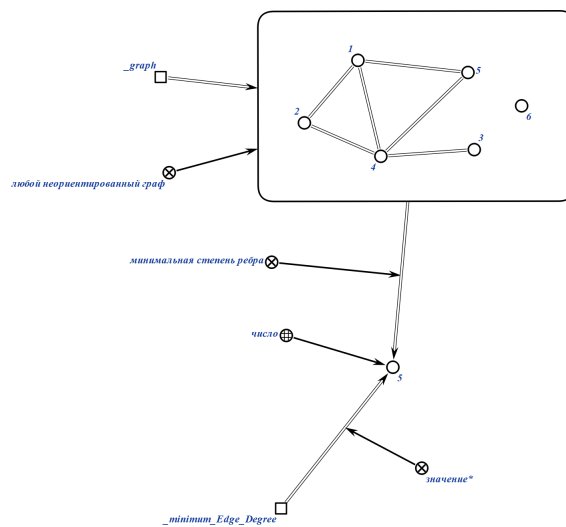


Рис. 9: Выход теста 3

3.4 Тест 4

Вход: Необходимо найти минимальное значение степени ребра неориентированного графа.

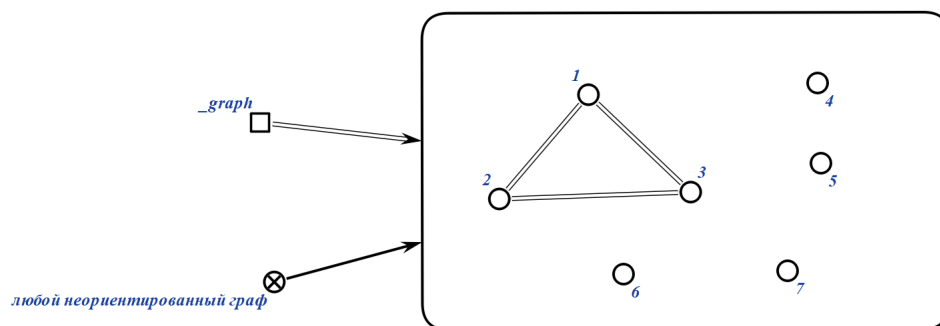


Рис. 10: Вход теста 4

Выход: Будет найдено значение: минимальная степень ребра неориентированного графа.

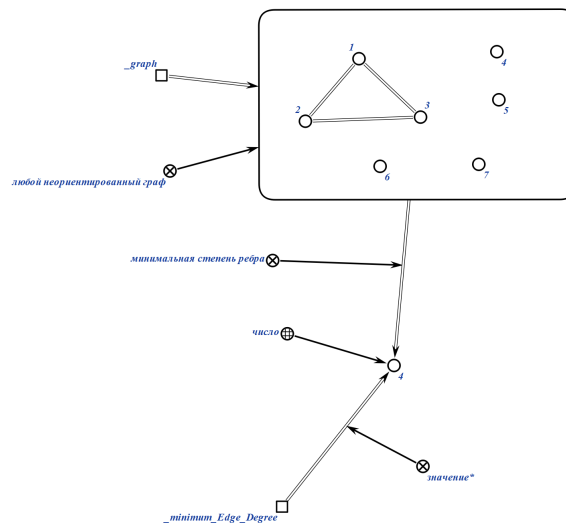


Рис. 11: Выход теста 4

4 Пример работы алгоритма в семантической памяти

4.1 Краткое описание:

1. Инициализация множества вершин и рёбер;
2. Создание переменной для хранения минимальной степени ребра;
3. Поиск смежных вершин для каждой вершины;
4. Инициализация поиска минимальной степени для каждого ребра;
5. Обход всех рёбер;
6. Завершение обхода всех ребер;
7. Вывод результата.

4.2 Демонстрация на тесте 5:

1. *graph* получит в качестве значения sc-узел неориентированного графа;

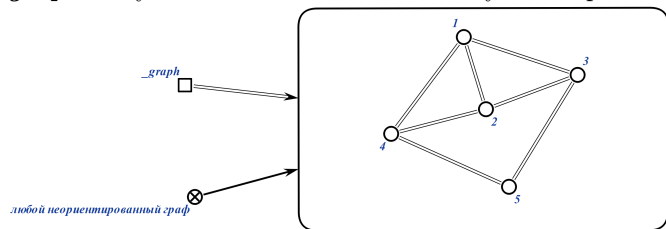


Рис. 12: Вход теста 5

2. Инициализируем переменную *minimum edge degree* для минимальной степени ребра.

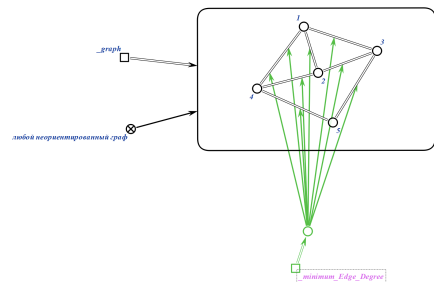


Рис. 13: Действие 1

3. Для вершины u создаём счётчик *degree U*, который будет хранить степень этой вершины, и аналогично для вершины v — переменная *degree V*. Мы будем вычислять эти степени для каждой вершины графа на основе матрицы смежности.

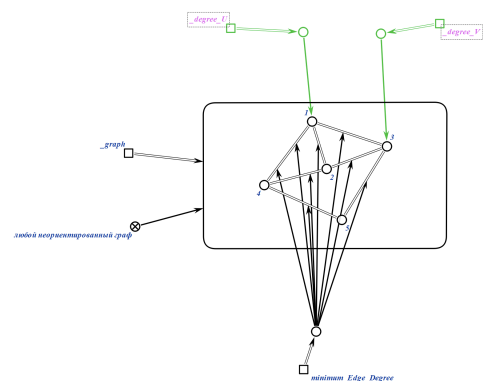


Рис. 14: Действие 2

4. Для каждой вершины u и v , между которыми есть ребро, считаем сумму степеней этих вершин и записываем в переменную *edge degree*

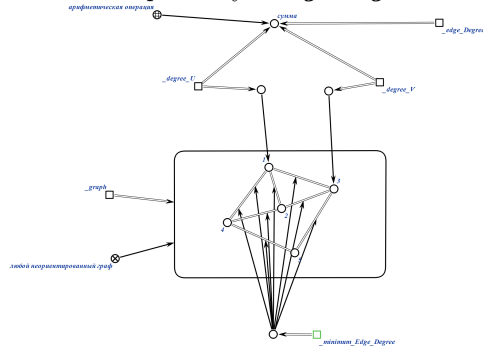


Рис. 15: Действие 3

5. Если значение *edge degree* меньше значения *minimum edge degree*, то присваиваем значение переменной *edge degree* переменной *minimum edge degree*

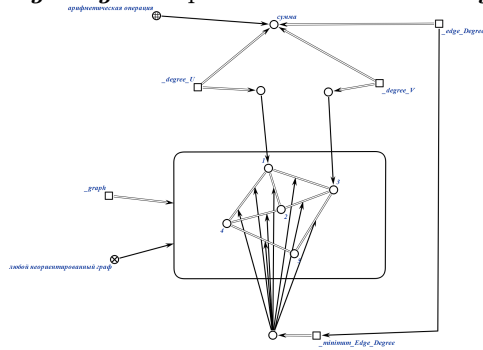


Рис. 16: Действие 4

6. Таким образом, мы получаем переменную *minimum Edge Degree*, которая содержит то, что мы искали

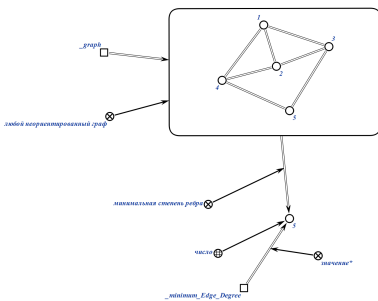


Рис. 17: Действие 5

5 Заключение

В заключении у нас получилось формализовать поставленную задачу. Мы нашли нужные нам числовые значения. Реализовали алгоритм их поиска, который работает на любом неориентированном связном графе.

6 Список использованных источников

- (a) Оре О. Теория графов. – 2-е изд.. – М.: Наука, 1980. – С. 336.
- (b) Кормен Т. Х. и др. Часть VI. Алгоритмы для работы с графами // Алгоритмы: построение и анализ = Introduction to Algorithms. – 2-е изд.. – М.: Вильямс, 2006. – С. 1296.
- (c) Харари, Ф. Теория графов / Ф. Харари / Пер. с англ. и предисл. В.П. Козырева. Под ред. Г.П. Гаврилова. Изд. 2-е. – М.: Едиториал УРСС, 2003. – 269 с.
- (d) Нечипуренко, М. И. Алгоритмы и программы решения задач на графах и сетях / М.И. Нечипуренко, В.К. Попков, С.М. Майнагашев и др. – Новосибирск: Наука. Сиб. отд-ние, 1990. – 515 с.