Министерство образования Республики Беларусь

Учреждение образования

“Белорусский государственный университет информатики и радиоэлектроник”

Факультет информационных технологий и управления

Кафедра интеллектуальных информационных технологий

Расчетная работа

По дисциплине "Представление и обработка информации в интеллектуальных системах"

на тему

"Задача нахождения гамильтонова цикла в неориентированнном графе"

Выполнил
Студент группы 121702 Смольник В.А.

Проверил Загорский А.Г.

Минск 2022

Содержание

Цель 2

Постановка задач 2

1 Список понятий 2

1. Графовая структура . . . 2
2. Графовая структура с ориентированными связками . . . 2
3. Графовая структура с неориентированными связками . . . 3
3. Гиперграф . . . 4
5. Псевдограф . . . 4
6. Мультиграф . . . 5
7. Граф . . . 6
8. Неориентированный граф . . . 6
9. Ориентированный граф . . . 7
10. Маршрут . . . 7
11. Цепь . . . 7
12. Простая цепь . . . 8
13. Цикл . . . 8
14. Гамильтонов цикл . . . 9

2 Алгоритм (Гамильтонов цикл для неор. графа) 10

3 Тестовые примеры 11

3.1 Тест 1 . . . 11
3.2 Тест 2 . . . 15
3.3 Тест 3 . . . 16
3.4 Тест 4 . . . 17

Вывод 18

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

Постановка задача: Найти эксцентриситет каждой вершины неориентированного графа

1 Список понятий

1. Графовая структура - это такая одноуровневая реляционная структура, объекты которой могут играть роль либо вершины, либо связки:
   a. Вершина (относительное понятие, ролевое отношение);
   b. Связка (относительное понятие, ролевое отношение).

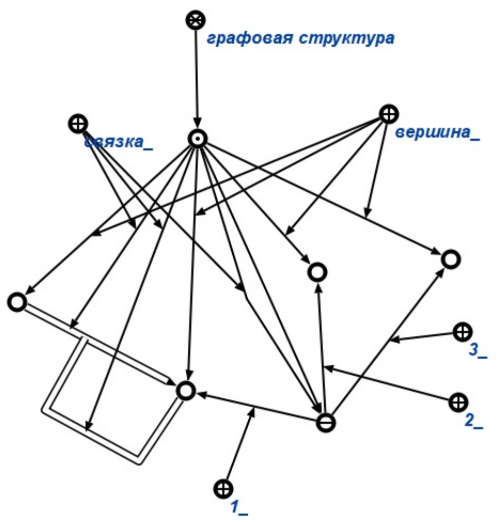


Рис.1 Графовая структура

2. Графовая структура с ориентированными связками (абсолютное понятие)
   a. Ориентированная связка (относительное понятие, ролевое отношение) –связка, которая задается ориентированным множеством.

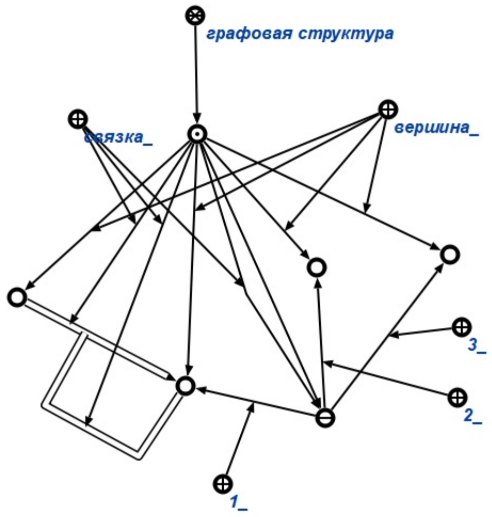


Рис.2 Графовая структура с ориентированными связками

3. Графовая структура с неориентированными связками(абсолютное понятие)
   a. Неориентированная связка (относительное понятие, ролевое отношение) –связка, которая задается неориентированным множеством.

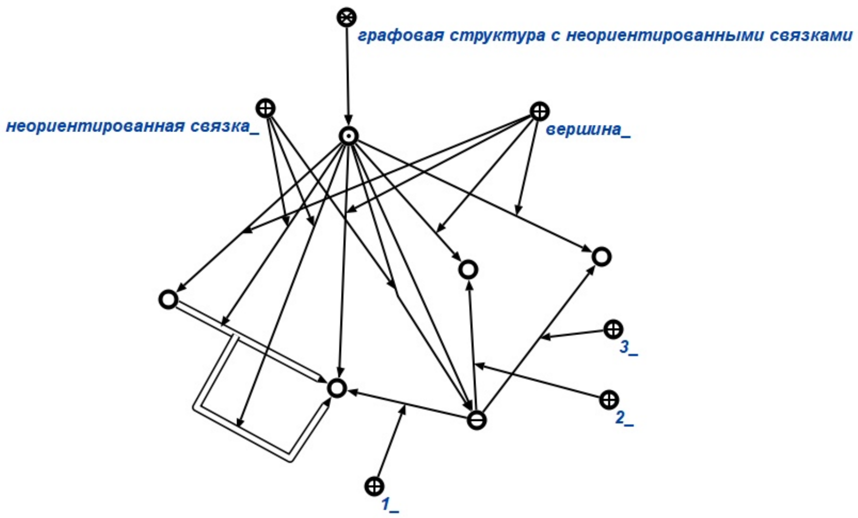


Рис.3 Графовая структура с неориентированными связками

4. Гиперграф (абсолютное понятие) – это такая графовая структура, в которой связки могут связывать только вершины:
   a. Гиперсвязка (относительное понятие, ролевое отношение);
   b. Гипердуга (относительное понятие, ролевое отношение) – ориентированная гиперсвязка;
   c. Гиперребро (относительное понятие, ролевое отношение) – неориентированная гиперсвязка.

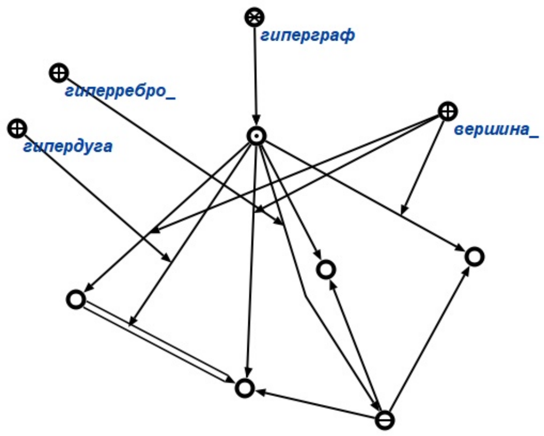


Рис.4 Гиперграф

5. Псевдограф (абсолютное понятие) – это такой гиперграф, в котором все связки должны быть бинарными:
   a. Бинарная связка (относительное понятие, ролевое отношение) - гиперсвязка арности 2;
   b. Ребро (относительное понятие, ролевое отношение) - неориентированна ягиперсвязка;
   c. Дуга (относительное понятие, ролевое отношение) – ориентированная гиперсвязка;
   d. Петля (относительное понятие, ролевое отношение) – бинарная связка, у которой первый и второй компоненты совпадают.

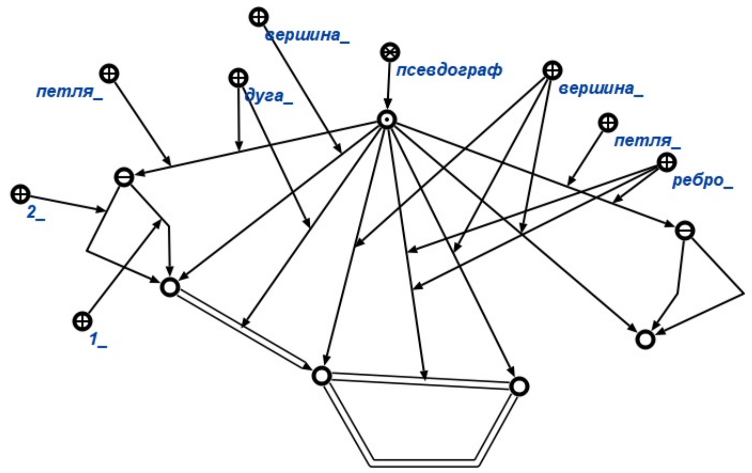


Рис.5 Псевдограф

6. Мультиграф (абсолютное понятие) – это такой псевдограф, в котором не может быть петель:

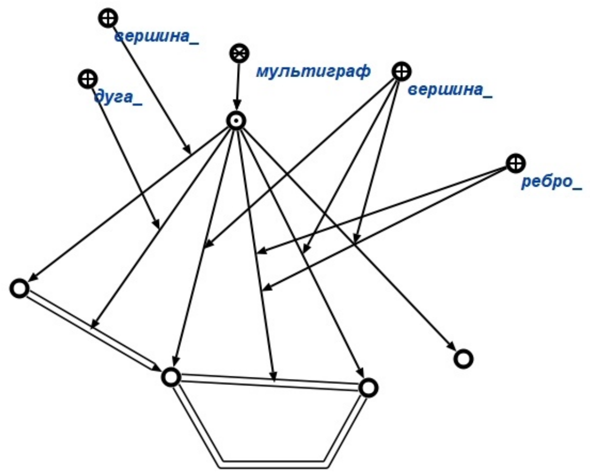


Рис.6 Мультиграф

7. Граф (абсолютное понятие) – это такой мультиграф, в котором не может быть кратных связок, т.е. связок у которых первый и второй компоненты совпадают:

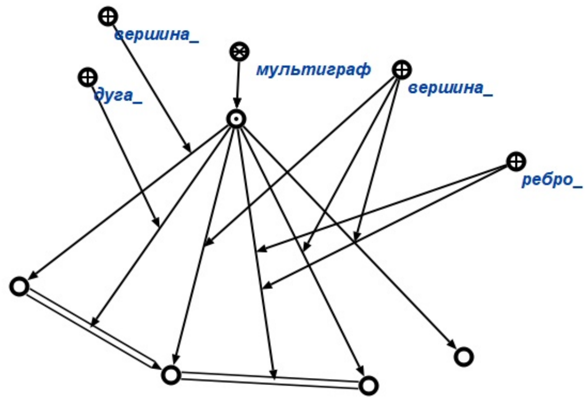


Рис.7 Граф

8. Неориентированный граф (абсолютное понятие) –это такой граф, в котором все связки являются ребрами:

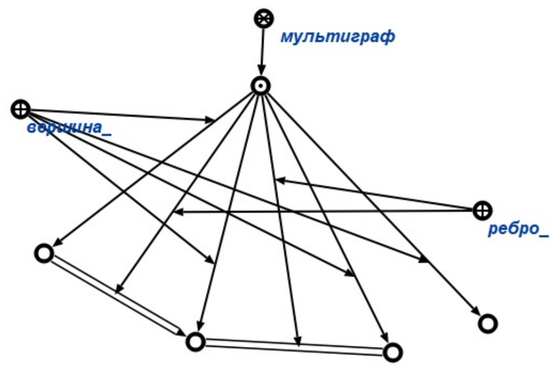


Рис.8 Неориентированный граф

9. Ориентированный граф (абсолютное понятие) - это такой граф, в котором все связки являются дугами:

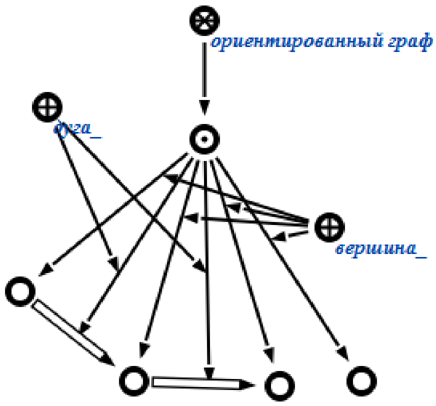


Рис.9 Ориентированный граф

10. Маршрут (относительное понятие, бинарное ориентированное отношение) – это чередующаяся последовательность вершин и гиперсвязок в гиперграфе, которая начинается и кончается вершиной, и каждая гиперсвязка последовательности инцидентна двум вершинам, одна из которых непосредственно предшествует ей, а другая непосредственно следует за ней.

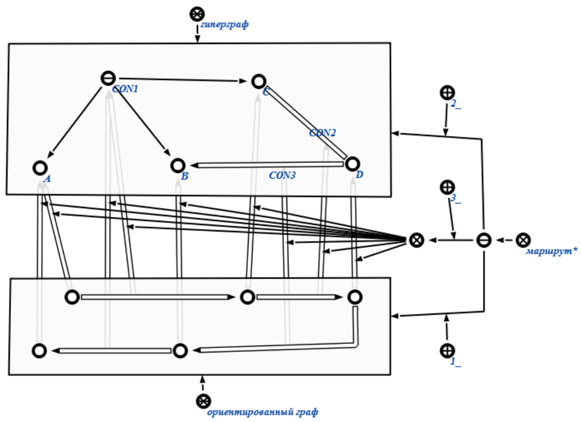


Рис.10 Маршрут

11. Цепь (относительное понятие, бинарное ориентированное отношение) – это маршрут, все гиперсвязки которого различны. В примере ниже показана цепь A, CON1, C, CON2, D, CON3, B, CON4, A в гиперграфе.

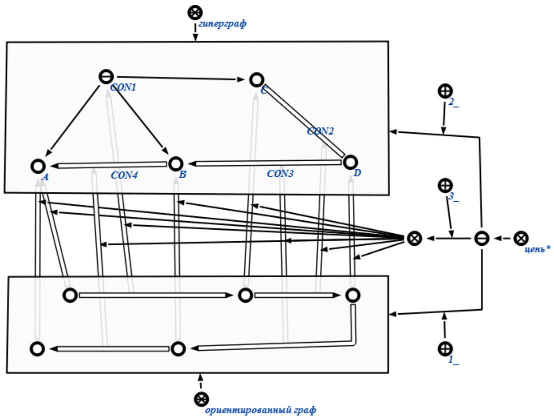


Рис.11 Цепь

12. Простая цепь, путь (относительное понятие, бинарное ориентированное отношение) – это цепь, в которой все вершины различны. В примере ниже показан путь A, CON1, C, CON2, D, CON3, B в гиперграфе.

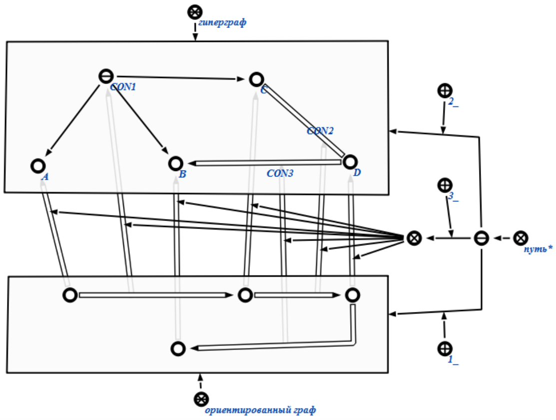


Рис.12 Простая цепь, путь

13. Цикл - цепь, которая начинается и заканчивается одной вершиной. При этом длиной цикла называют число составляющих его рёбер. В примере ниже показан цикл A, CON1, C, CON2, D, CON3, B

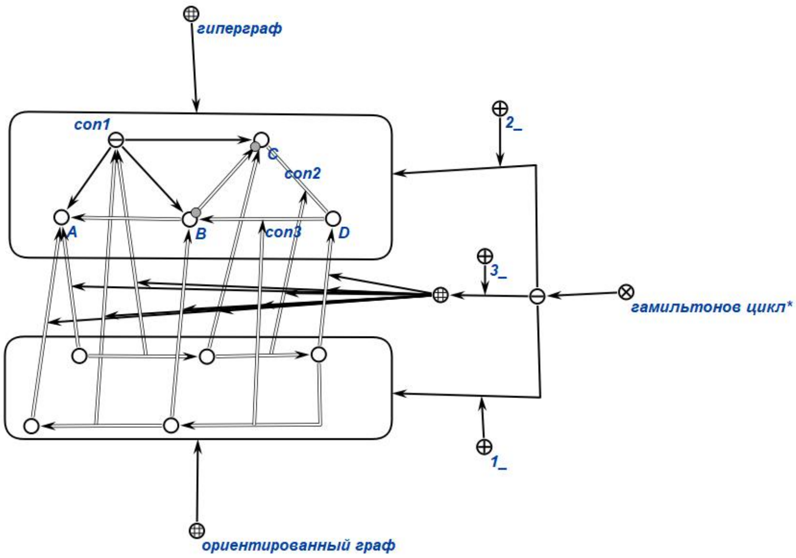


Рис.13 Цикл

14. Гамильтонов цикл - такой цикл, который проходит через каждую вершину данного графа ровно по одному разу, то есть цикл, в который входит все вершины графа. В примере ниже показан гамильтонов цикл A, CON1, C, CON2, D, CON3, B.

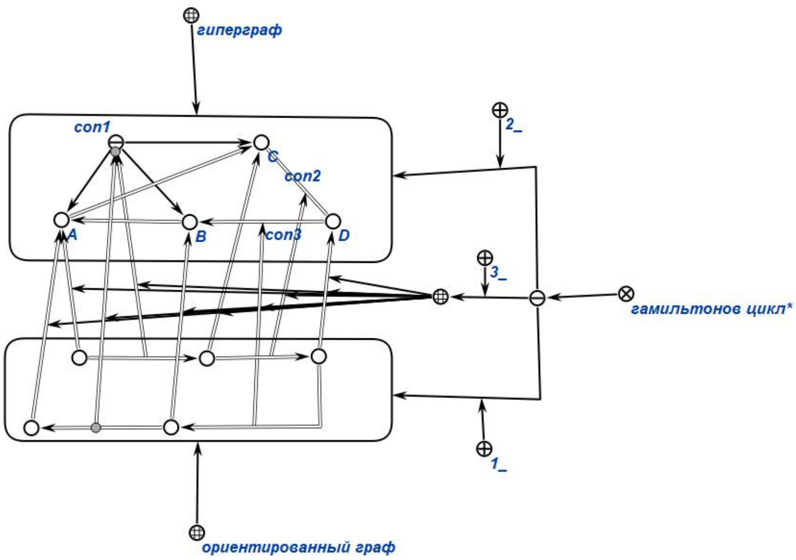


Рис.14 Гамильтонов Цикл

2 Алгоритм (Гамильтонов цикл для неор. графа)

1. Задаем ориентированное множество path, изначально равное пустому множесту;
2. Произвольным образом берем вершину графа.
3. В множество path добавляем номер текущей вершины графа;
4. Задаем множество children и добавляем в него номера всех вершин, смежных текущей вершине, начинаем рассмотрение с первого элемента множества children;
5. Рассмотрим несколько случаев:
   (a) Если значение просмативаемого элемента множетва children равно first и мощность множества path равна количеству вершин, то мы нашли Гамильтонов цикл, в множество path добавляем первый элемент множества path (для заыкания цепи). Цикл будут образовывать вершины , номера которых являются элементами множества path в определенном порядке
   (b) Если значение просматриваемого элемента множества children равно first и размер множества path не равен количеству вершин, то переходим к пункту e
   (c) Если значение просматриваемого элемента множества children принадлежит множесту path, то переходим к пункту e;
   (d) Переходим к вершине с номером, равным значению просматриваемого элемента множества children и переходим к пункту 3.
   (e) Проверяем наличие непроверенных вершин, смежных текущей вершине:
      i. Если текущий элемент не является последним элементом в множестве children, то берем следующий элемент (после текущего) множества children и переходим к пункту 5;
      ii. Рассмотрим два варианта:

A. Если мощность множества path не равна 1, то возвращаемся к предыдущей рассматриваемой вершине (к вершине с номером, равнам значению предпоследнего элемента множества path), при этом из множества path удаляем последний элемент, переходим к следующему элементу из множества children (при вершине с номером, равным уже последнему элементу множества path). Переходим к пункту 4.

B. Если мощность множества path равна 1, то в данном графе нету Гамильтонова цикла, выходим из программы.

3 Тестовые примеры

3.1 Тест 1

Вход:

Определить наличие Гамильтонова цикла и вывести его, если он существует.

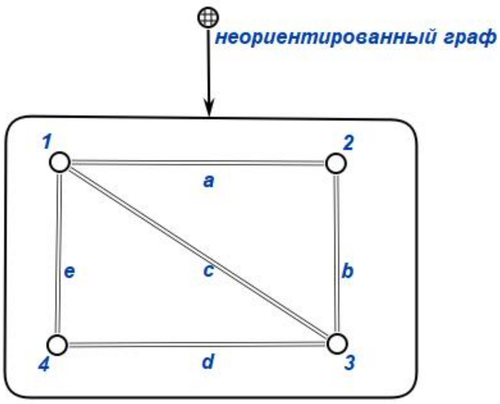


Рис. 15 Неориентированный граф для поиска Гамильтонова цикла

Шаг 1:

Задаем ориентированное множество path=<>. Произвольным образом возьмем вершину графа, пусть вершиной графа будет вершина с номером 1. В множество path добавляем значение переменной first (1). Задаем множество children, элементами которого будут номера всех вершин, смежных текущей вершине.

children = 2, 3, 4.

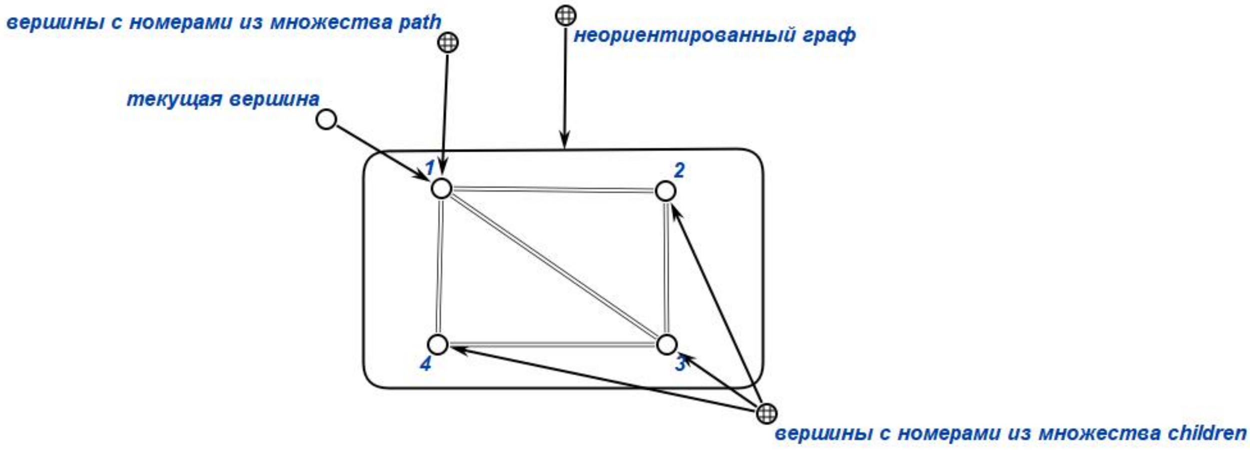


Рис.16 Шаг 1

Шаг 2:
Рассматриваем случай с 1-ым элементом множества children (2). Поскольку 2 не равно значению первого элемента множества path и не принадлежит множеству path, то мы переходим к вершине с номером 2, а в множество path=<1> добавляем новый элемент: номер текущей вершины(2). Задаем множество children, элементами которого будут номера всех вершин, смежных текущей вершине. children={1, 3}.

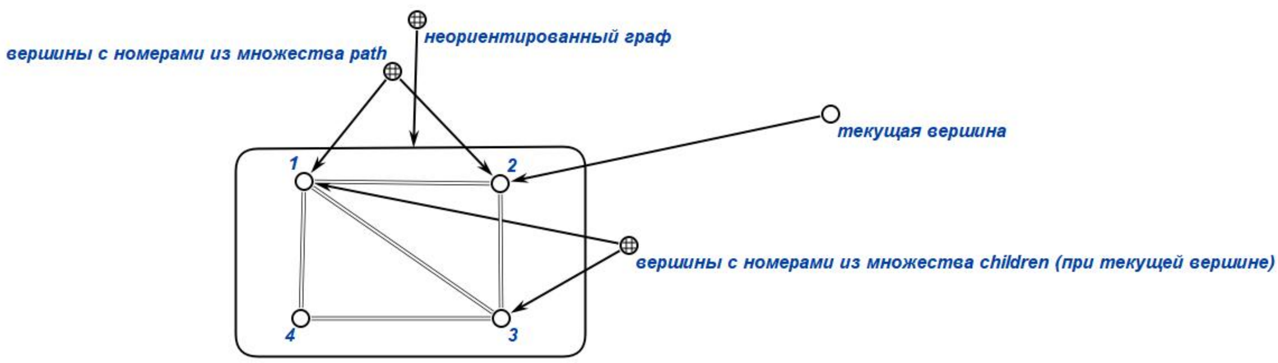


Рис.17 Шаг 2

Шаг 3:
Рассмативам случай с 1-ым элементом множества children (1). Поскольку значение 1-го элемента (1) равно значению первого

элемента множества path, а мощность множества path (2) не равна количеству вершин графа, то берем 2-ой элемент множества children (3). Поскольку 3 не равно значению first и не принадлежит множеству path=<1, 2>, то мы переходим к вершине с номером 3, а в множество path добавляем новый элемент: номер текущей вершины(3). Задаем множество children, элементами которого будут номера всех вершин, смежных текущей вершине. children={1, 2, 4}.

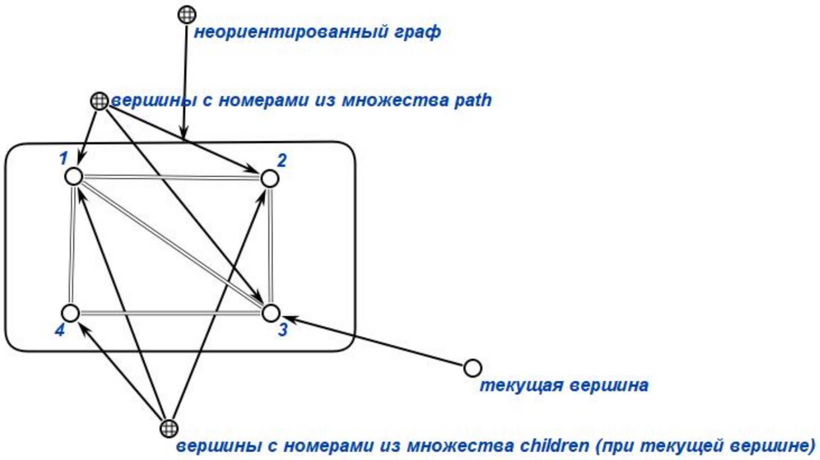


Рис.18 Шаг 3

Шаг 4:

Рассматривам случай с 1-ым элементом множества children (1). Поскольку значение 1-го элемента (1) равно значению первого элемента множества path, а мощность множества path (3) не равна количеству вершин графа, то берем второй элемент множества children(2). Поскольку значение 2-го элемента принадлежит множеству path=<1, 2, 3>, то переходим к третьему элементу множества children (4). Поскольку 4 не равно значению первого элемента множества path и не принадлежит множеству path=<1, 2, 3>, то мы переходим к вершине с номером 4, а в множество path добавляем новый элемент: номер текущей вершины(4). Задаем множество children, элементами которого будут номера всех вершин, смежных текущей вершине. children={1, 3}.

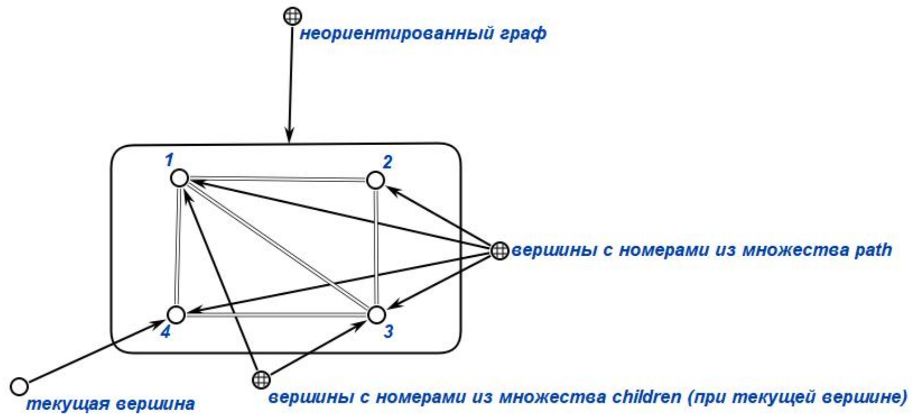
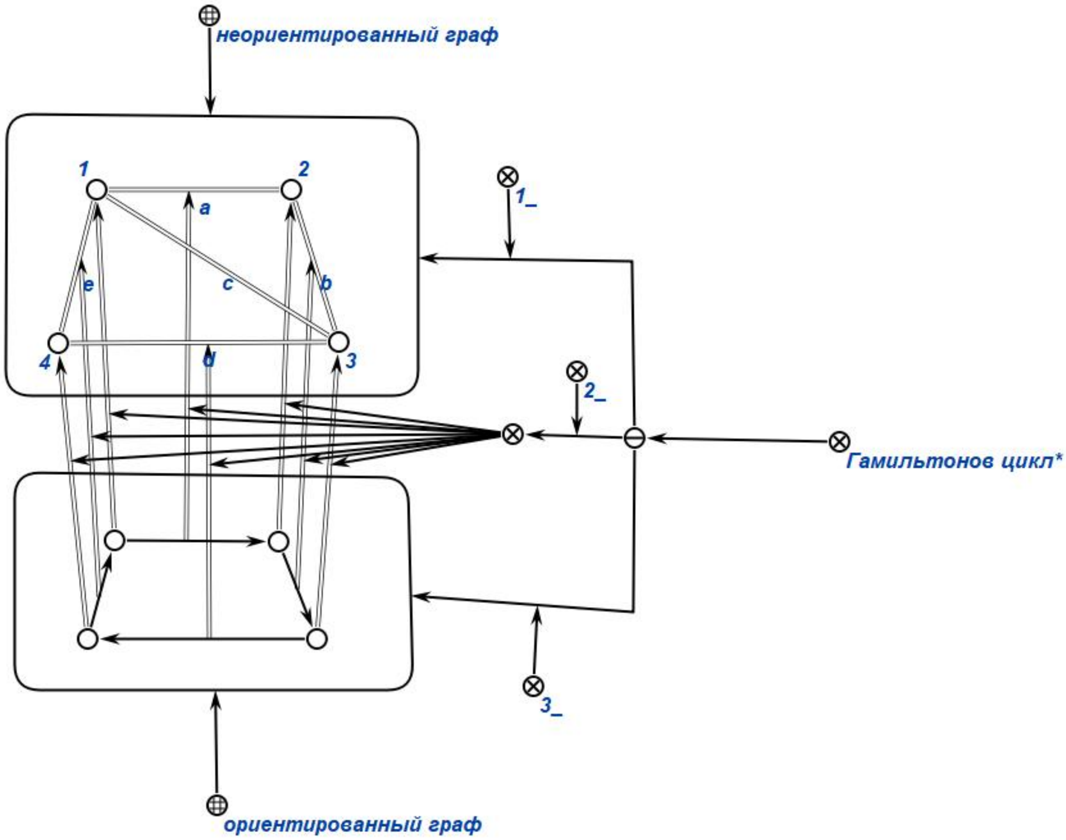


Рис.16 Шаг 4

Шаг 5:
Рассматривам случай с 1-ым элементом множества children (1). Поскольку значение 1-го элемента (1) равно значению первого элемента множества path, и мощность множества path (4) равна количеству вершин графа, то мы нашли Гамильтонов цикл. В множество path=<1, 2, 3, 4> добавляем значение первого элемента множества path (1). Найденный Гамильтонов цикл будет образован вершинами графа в оперделенном порядке, номера которых являются элементами ориентированного множества path.

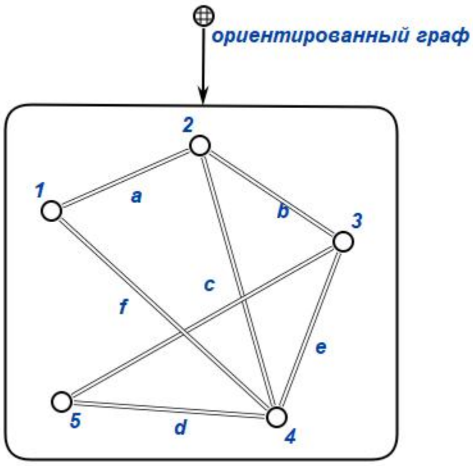
Выход:
Гамильтонов цикл найден: 1->2->3->4->1.



3.2 Тест 2

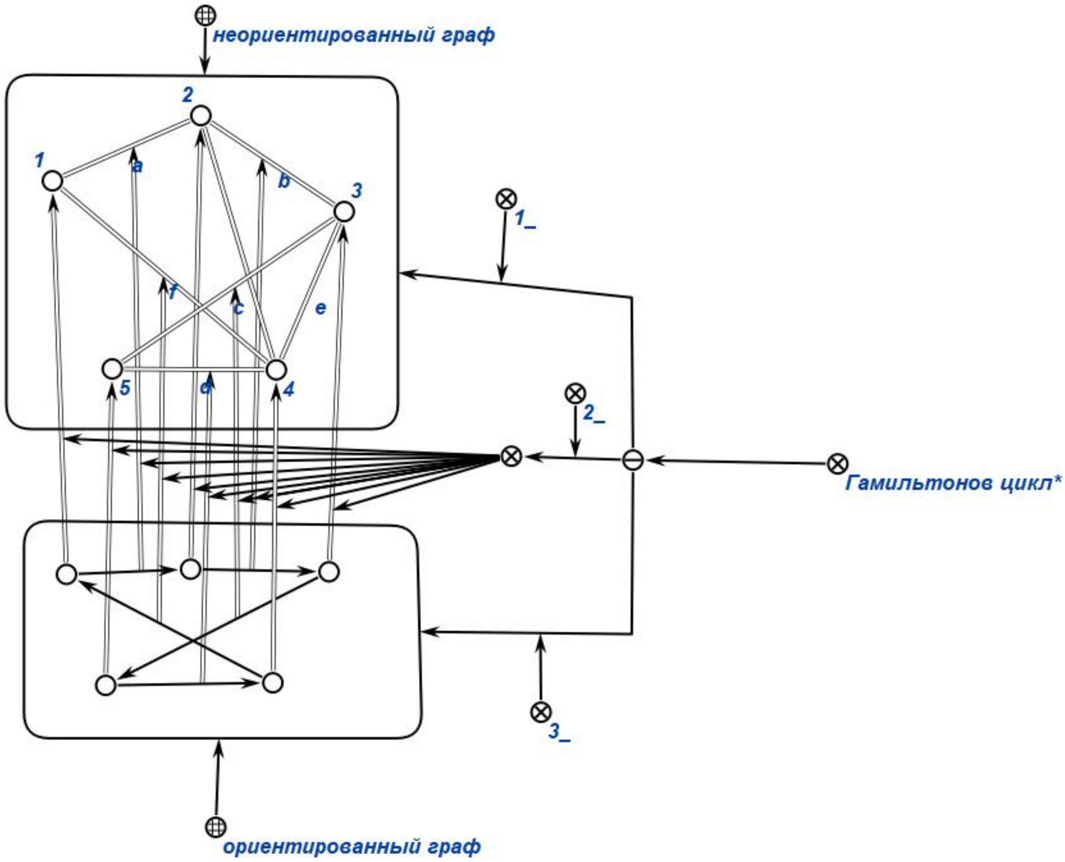
Вход:

Определить наличие Гамильтонова цикла и вывести его, если он существует.



Выход:

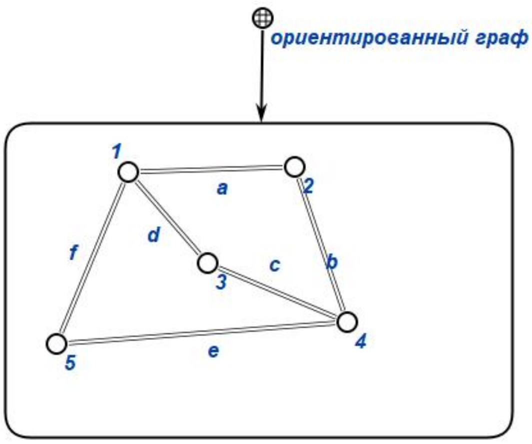
Гамильтонов цикл найден: 1->2->3->5->4->1.



3.3 Тест 3

Вход:

Определить наличие Гамильтонова цикла и вывести его, если он существует.



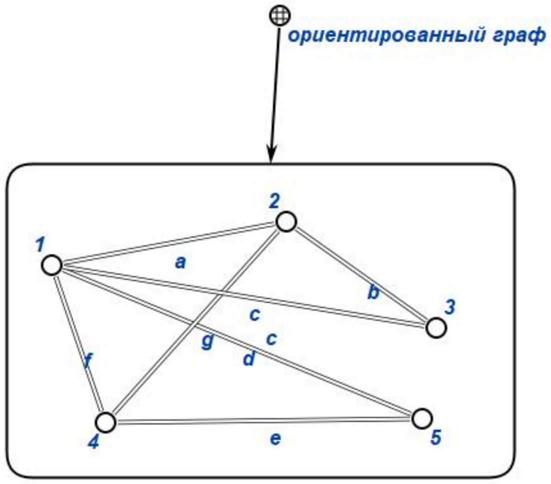
Выход:

Гамильтонов цикл не был обнаружен.

3.4 Тест 4

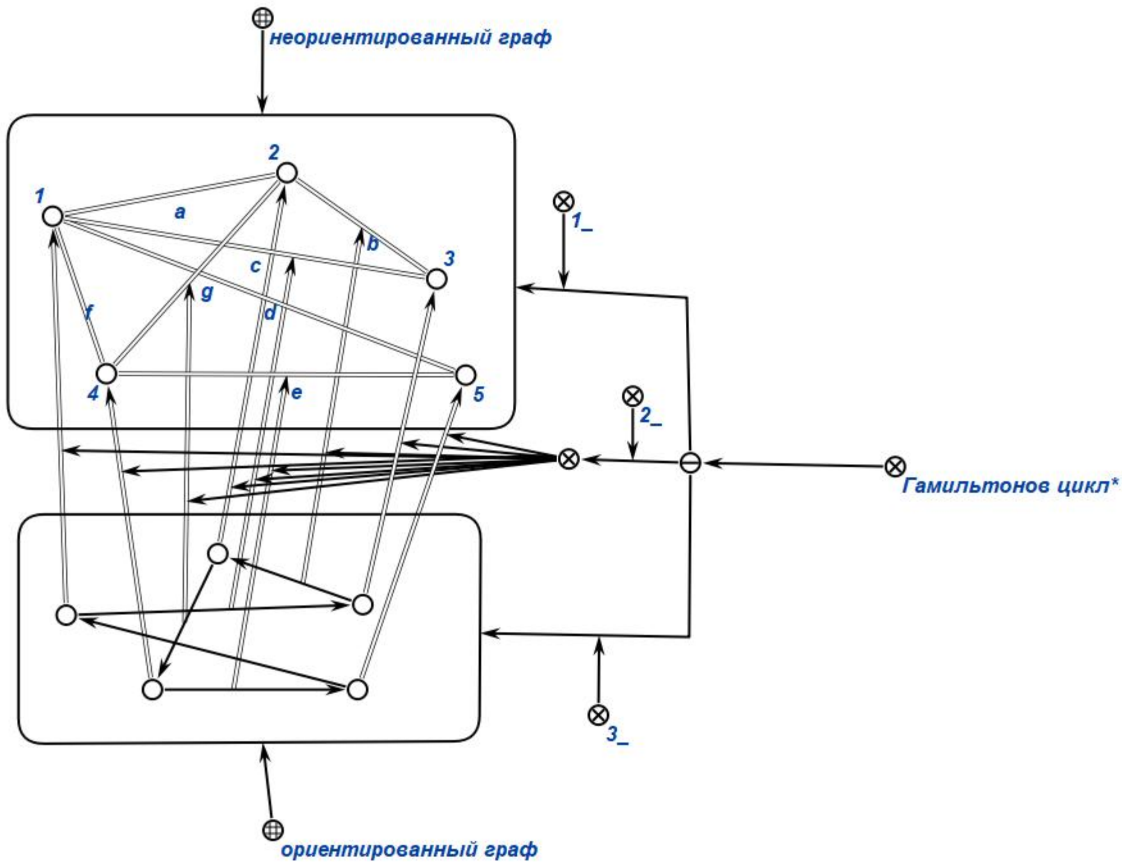
Вход:

Определить наличие Гамильтонова цикла и вывести его, если он существует.



Выход:

Гамильтонов цикл найден: 1->3->2->4->5->1.



Вывод:

В ходе выполнения работы был изучен алгоритм поиска Гамильтонова цикла и применение его в конкретной ситуации. Были изучены понятия графа, мультиграфа,взвешенного графа, псевдографа, гиперграфа,графовой структуры, графовой структуры с ориентированными связками, графовой структуры с неориентированными связками, неориентированного графа, цепи, маршрута, цикла, Гамильтонова цикла.

Список использованных источников

OSTIS GT. База знаний по теории графов OSTIS GT. - 2011. [Электронный ресурс] - Режим доступа: http://ostisgraphstheo.sourceforge.net/index.php. Дата доступа - 28.03.2022

Гладков Л.А., Курейчик В. В., Курейчик В.М. Дискретная математика. Под ред. В.М. Курейчика. — М.: ФИЗМАТЛИТ, 2014. - 325с.