

Министерство образования Республики Беларусь

Учреждение образования
БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет Информационных технологий и управления
Кафедра Интеллектуальных информационных технологий

К защите допустить:

Заведующий кафедрой

_____ Д. В. Шункевич

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

к расчетной работе

по дисциплине «Математические основы интеллектуальных систем»:

Нахождение хорд в неориентированном графе

БГУИР РР 1-40 03 01

Студент:

А. Ю. Гаврилович

Группа:

221703

Руководитель:

С. А. Никифоров

Минск 2023

СОДЕРЖАНИЕ

Введение	5
1 Список используемых понятий	6
2 Алгоритм решения задачи	7
3 Примеры и результаты	9
Заключение	11
Список использованных источников	12

ВВЕДЕНИЕ

Цель: получить базовые знания и представления о графе и научиться работать с графом, изучить строение ostis-агента и научиться его писать.

Задача: разработать ostis-агента, который найдет все хорды в неориентированном графе.

Хорды имеют широкий спектр применений в решении задач, связанных с теорией графов. Планирование маршрутов: если граф представляет собой дорожную сеть или сеть связи, знание о наличии хорд может помочь в выборе наиболее оптимального маршрута или распределении ресурсов.

Поиск циклов: хорды могут образовывать циклы в графе, что может быть полезно при анализе данных или поиске кратчайшего пути.

Алгоритмы оптимизации: хорды могут использоваться для определения метрик, таких как расстояние между вершинами или центральность графа, что помогает в разработке алгоритмов оптимизации.

Анализ социальных сетей: наличие хорд может указывать на наличие связей между вершинами, что полезно при анализе социальных сетей или графа друзей.

Графическая визуализация: хорды могут быть использованы для создания эффектов визуализации графа или улучшения его внешнего вида.

В целом, нахождение хорд в графе может предоставить дополнительную информацию о его структуре и связях между вершинами

1 СПИСОК ИСПОЛЬЗУЕМЫХ ПОНЯТИЙ

1. **Граф**[1] - математическая абстракция реальной системы любой природы, объекты которой обладают парными связями. Граф как математический объект есть совокупность двух множеств — множества самих объектов, называемого множеством вершин, и множества их парных связей, называемого множеством рёбер. Элемент множества рёбер есть пара элементов множества вершин

2. **Неориентированный граф**[1] - это граф, в котором узлы не имеют направления.

3. **Цикл**[1] — граф, состоящий из единственного цикла, или, другими словами, некоторого числа вершин, соединённых замкнутой цепью.

4. **Хорда**[2] - это ребро, соединяющее две вершины, не смежные в цикле. В неориентированном графе хорда также является неориентированной.

2 АЛГОРИТМ РЕШЕНИЯ ЗАДАЧИ

1.Обходим граф с помощью dfs [3] для нахождения всех циклов .Вот некоторые обенности данного поиска:

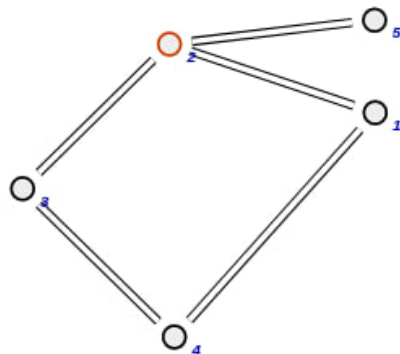
- а) Выбираем стартовую вершину
 - б) Переходим к смежной вершине,устанавливаем текущую вершину
 - в) Если стартовая вершина равна текущей вершине и цикл не пуст,цикл завершен;возвращаемся к предыдущей вершине, у которой еще остались непройденные смежные вершины,иначе выбираем новую стартовую вершину
 - г) Если текущая вершина уже записана в цикл,переходим к смежной вершине текущей вершины,переустанавливаем текущую вершину
 - д) Если смежные вершины закончились,а цикл не завершен,возвращаемся обратно до той вершины,у которой еще остались непройденные смежные вершины
 - е) Алгоритм выполняется до тех пор пока все вершины не побудут стартовыми и не будут пройдены все остальные вершины
- Если в графе есть циклы,это еще не гарант того,что в нем будут хорды[4] На рисунке 3.3 демонстрируется цикличный граф,но без хорд В ответе указаны все его характеристики 2.Ищем хорды[4] в каждом цикле

- а) Выбираем цикл
- б) Если цикл размером меньше трех,переход к другому циклу
- в) Определяем по формуле индекс последней стартовой вершины,чтобы избежать повторений хорд(на рисунке 3.1 количество стартовых вершин будет равно двум,например, 1 и 2,тк от вершины 4 идет та же хорда,что и от вершины 2),тк граф неориентированный(например: хорда 2-4 и 4-2 это одно и то же ребро).На рисунке 3.2 видно,что в ответе хорды 3-2 и 4-5,а не 3-2,2-3,4-5,5-4
- г) Определяем стартовую вершину
- д) Переходим вершине,смежной стартовой
- е) Если данная вершина не соседняя по отношению к стартовой,проверяем не записана ли данная хорда в вектор результатов,тк она могла уже быть в других циках
- ж) Если проверка пройдена успешно,записываем хорду в вектор хорд,иначе переход к пункту з)
- з) Если все смежные вершины пройдены,переход к пункту и),иначе к пункту д)
- и) Если все допустимые вершины графа были стартовыми,завершаем поиск хорд в данном цикле,переход к пункту к),иначе переход к пункту г)
- к) Если данный цикл последний,завершение,иначе - переход к пункту а)

Данный алгоритм работает только для неориентированного графа, поэтому в свой код я также включала проверку на неориентированность. На рисунке 3.4 демонстрируется попытка найти хорды в ориентированном графе, но в ответе мы видим только принадлежность исходного графа к классу ориентированных графов.

3 ПРИМЕРЫ И РЕЗУЛЬТАТЫ

Пример 1



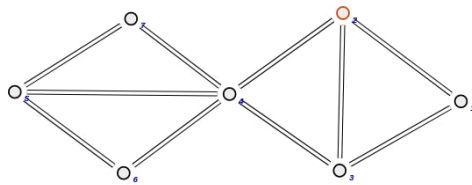
(a)

```
testGraph3
∈ UnchordalGraphClass
∈ CyclicGraphClass
∈ UndirectedGraphClass
```

(б)

Рисунок 3.1 – Граф 1 и результат вывода агента
В данном примере мы рассматриваем граф с циклом, но в нем отсутствуют хорды

Пример 2



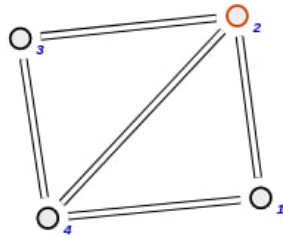
(a)

```
testGraph4
= (3 ↔ 2)
  ∈ HordClass
= (4 ↔ 5)
  ∈ HordClass
∈ ChordalGraphClass
∈ CyclicGraphClass
∈ UndirectedGraphClass
```

(б)

Рисунок 3.2 – Граф 2 и результат вывода агента
В данном примере мы рассматриваем граф с двумя хордами

Пример 3



```
testGraph6
∈ ChordalGraphClass
∈ CyclicGraphClass
∈ UndirectedGraphClass
= (4 = 2)
∈ HordClass
```

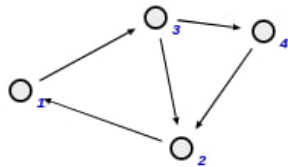
(a)

(б)

Рисунок 3.3 – Граф 3 и результат вывода агента

В данном примере мы рассматриваем граф с одной хордой

Пример 4



```
testGraph7
∈ DirectedGraphClass
```

(a)

(б)

Рисунок 3.4 – Граф 4 и результат вывода агента

В данном примере мы рассматриваем ориентированный граф

ЗАКЛЮЧЕНИЕ

В ходе данной расчётной работы мной изучены принципы создания *ostis-агента*. Я изучила его *структуру* и научилась писать *ostis-агента*, который способен выполнять определенную задачу. В данном случае я реализовала *ostis-агента*, который способен находить *хорды* в *неориентированном графе*.

СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ

- [1] Теория Графов. Часть 1 Введение и классификация графов. <https://habr.com/ru/articles/564594/>.
- [2] Хордальный граф. https://ru.wikipedia.org/wiki/РѢР«СТРѸРѸРѸСѸР,,СКРѸ_РѸСТРѸСн/.
- [3] Dfs. <https://habr.com/ru/articles/50437>.
- [4] Поиск хорд. https://nauchniestati.ru/spravka/hordy_grafa.