

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
Задача поиска гамильтонова цикла в ориентированном графе

Выполнил
студент группы
021704

Степанов Н.Ю.

Проверил

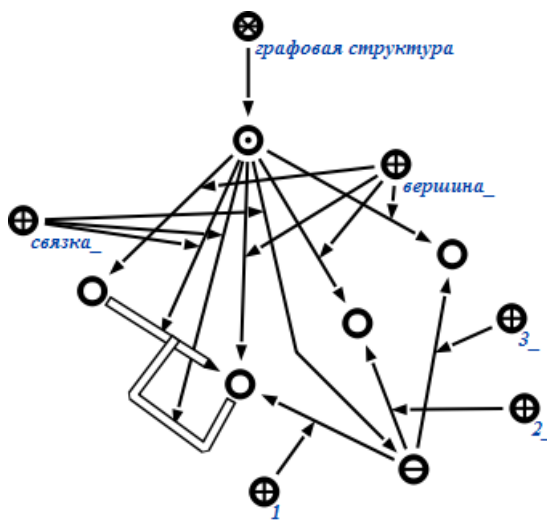
Витязь В.С

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

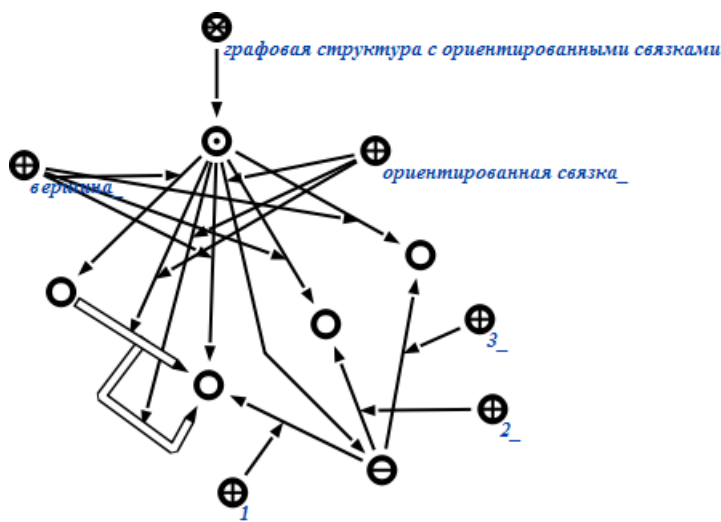
Задача: поиск гамильтонова цикла в ориентированном графе

1 Список понятий

1. Графовая структура (абсолютное понятие) - 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).

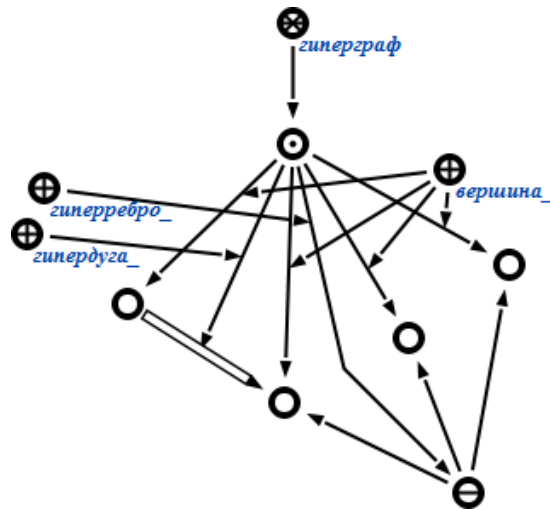


2. Графовая структура с ориентированными связками (абсолютное понятие)
 - a. Ориентированная связка (относительное понятие, ролевое отношение) – связка, которая задается ориентированным множеством.



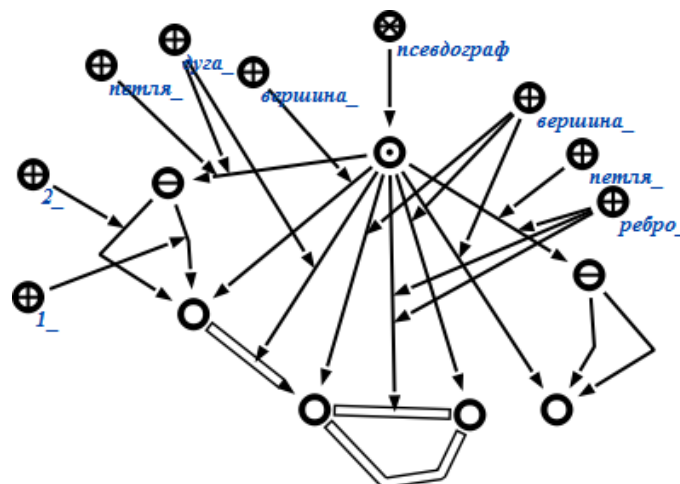
3. Гиперграф (абсолютное понятие) – это такая графовая структура, в которой связки могут связывать только вершины:
 - a. Гиперсвязка (относительное понятие, ролевое отношение);

- b. Гипердуга (относительное понятие, ролевое отношение) – ориентированная гиперсвязка;
- c. Гиперребро (относительное понятие, ролевое отношение) – неориентированная гиперсвязка.

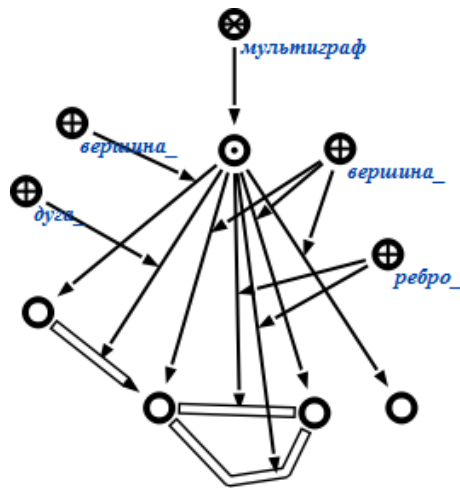


4. Псевдограф (абсолютное понятие) – это такой гиперграф, в котором все связи должны быть бинарными:

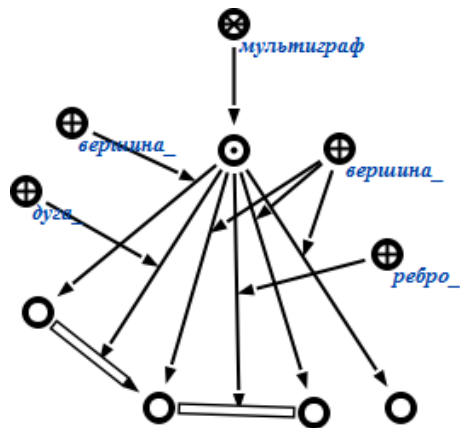
- a. Бинарная связка (относительное понятие, ролевое отношение) – гиперсвязка арности 2;
- b. Ребро (относительное понятие, ролевое отношение) – неориентированная гиперсвязка;
- c. Дуга (относительное понятие, ролевое отношение) – ориентированная гиперсвязка;
- d. Петля (относительное понятие, ролевое отношение) – бинарная связка, у которой первый и второй компоненты совпадают.



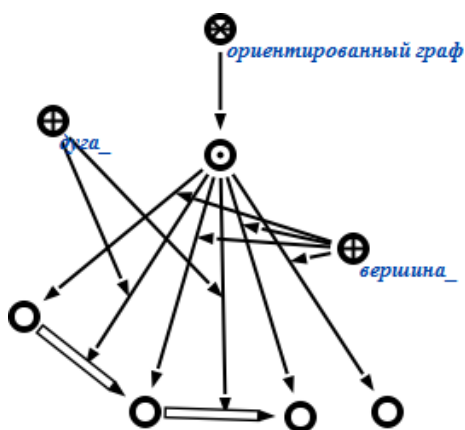
5. Мультиграф (абсолютное понятие) – это такой псевдограф, в котором не может быть петель:



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

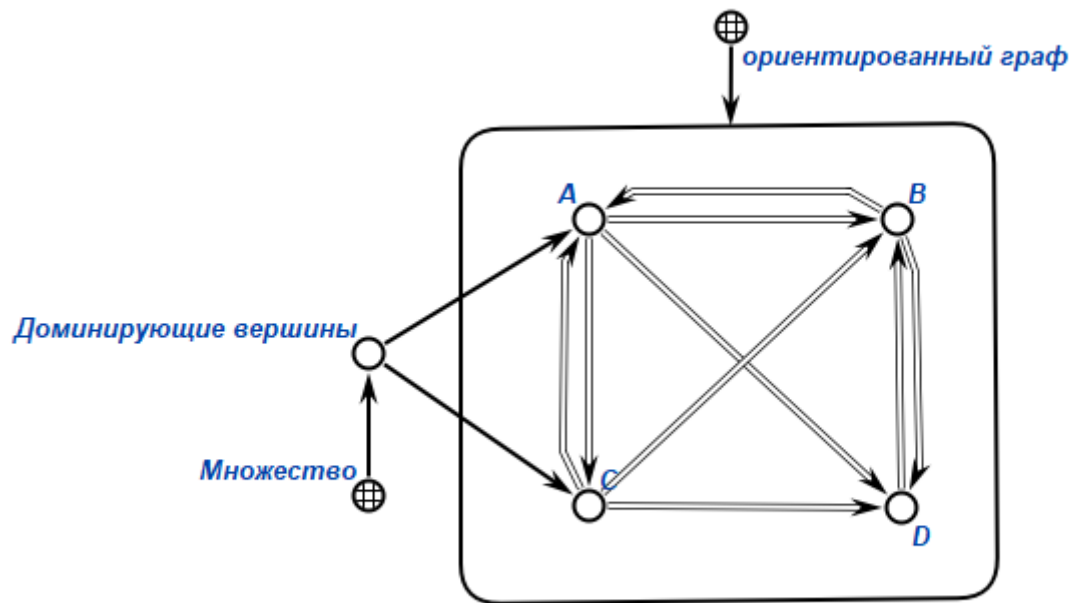


7. Ориентированный граф (абсолютное понятие) - это такой граф, в котором все связи являются дугами:



8. Доминирующая вершина (относительное понятие) – это вершина степени выхода $n-1$ графа порядка n , где порядок графа – количество вершин в граф, степень выхода вершины – количество ребер графа, для которых вершина является начальной.

В примере доминирующие вершины: А, С.



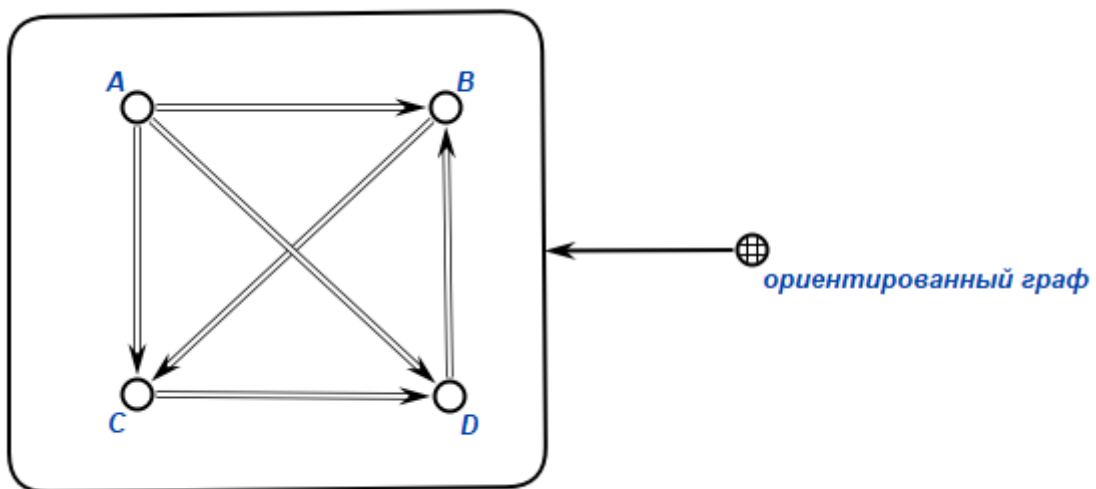
2 Тестовые примеры

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

2.1 Тест 1

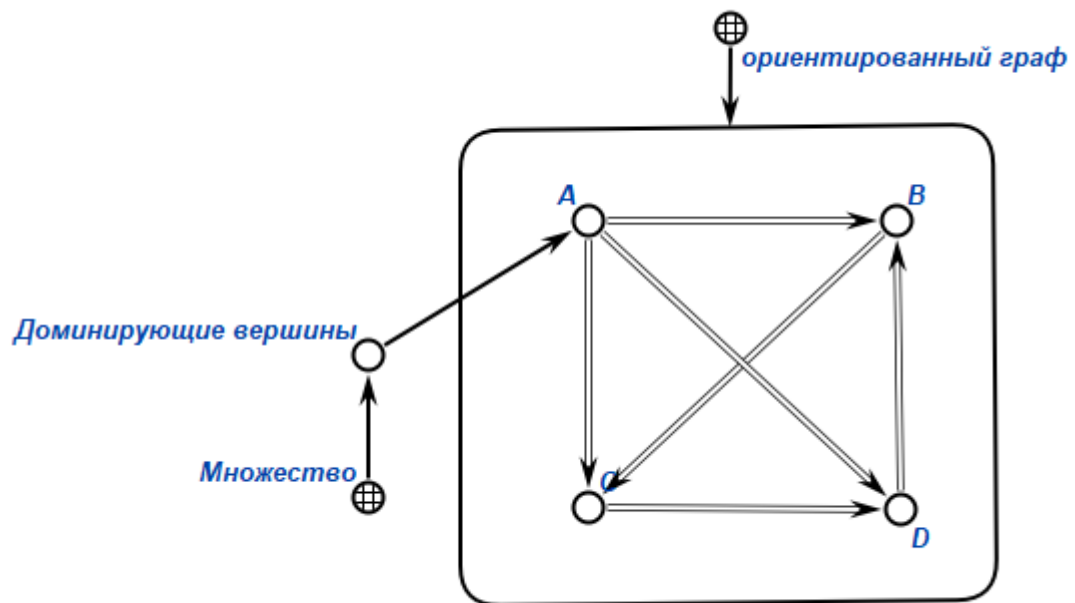
Вход:

Необходимо найти все доминирующие вершины ориентированного графа.



Выход:

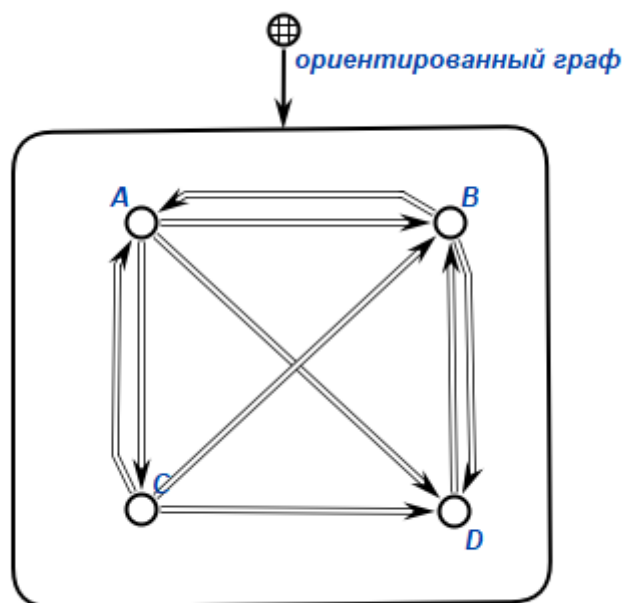
Будет найдена доминирующая вершина A.



2.2 Тест 2

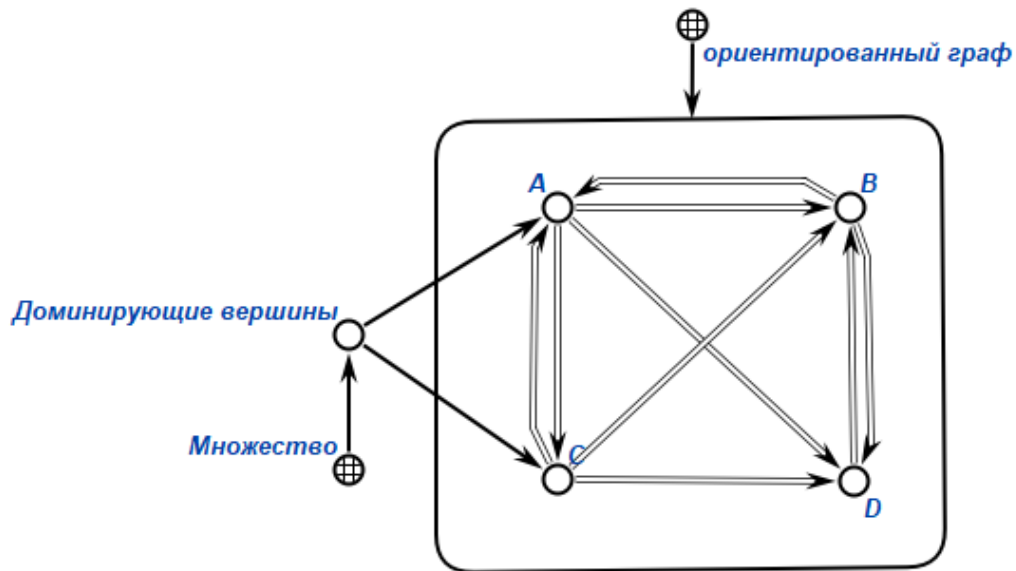
Вход:

Необходимо найти все доминирующие вершины ориентированного графа.



Выход:

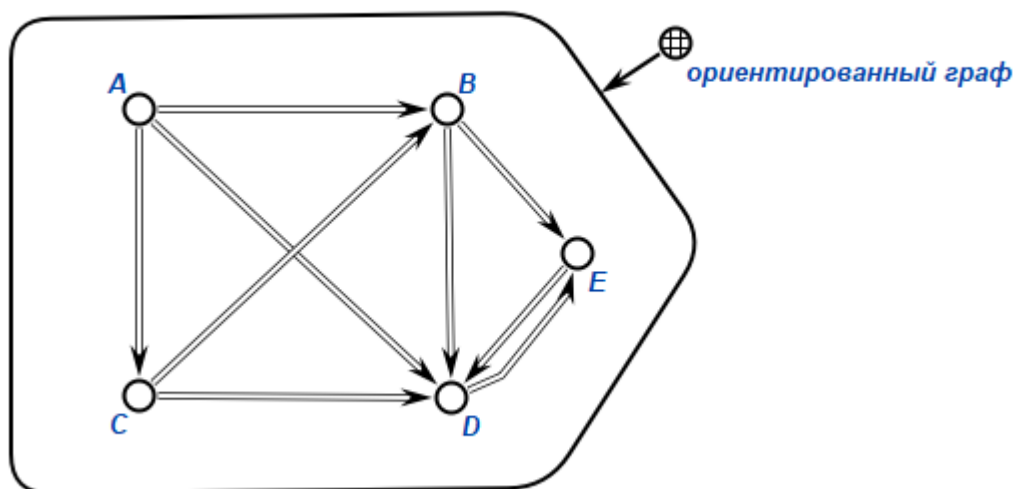
Будут найдены доминирующие вершины A и C.



2.3 Тест 3

Вход:

Необходимо найти все доминирующие вершины ориентированного графа.



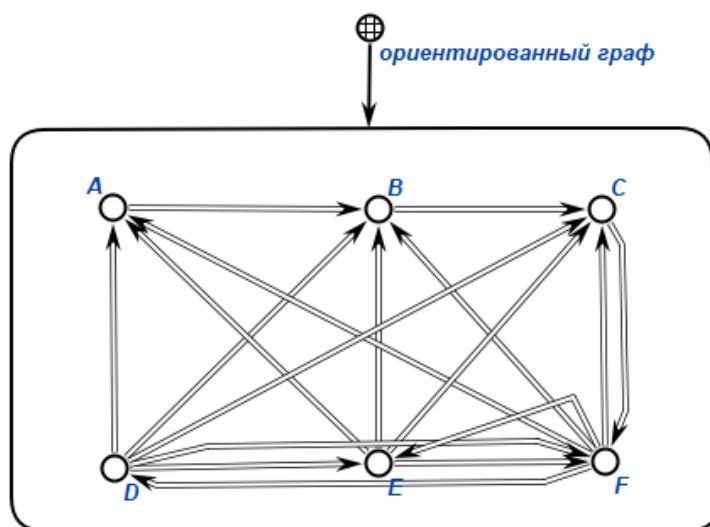
Выход:

Доминирующих вершин не существует. Программа должна вернуть ошибку вызывающему контексту.

2.4 Тест 4

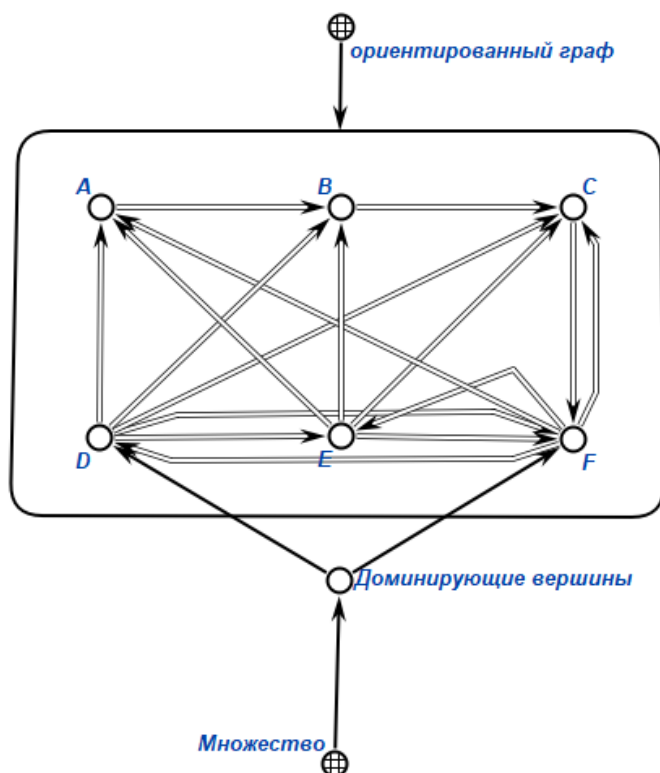
Вход:

Необходимо найти все доминирующие вершины ориентированного графа.



Выход:

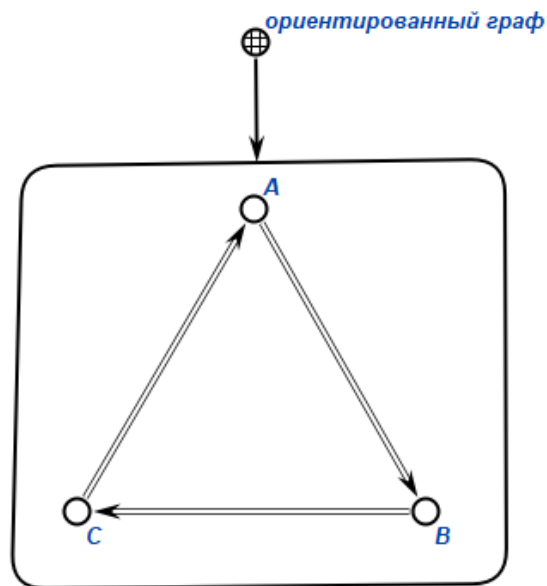
Будут найдены доминирующие вершины D и F.



2.5 Тест 5

Вход:

Необходимо найти все доминирующие вершины ориентированного графа.

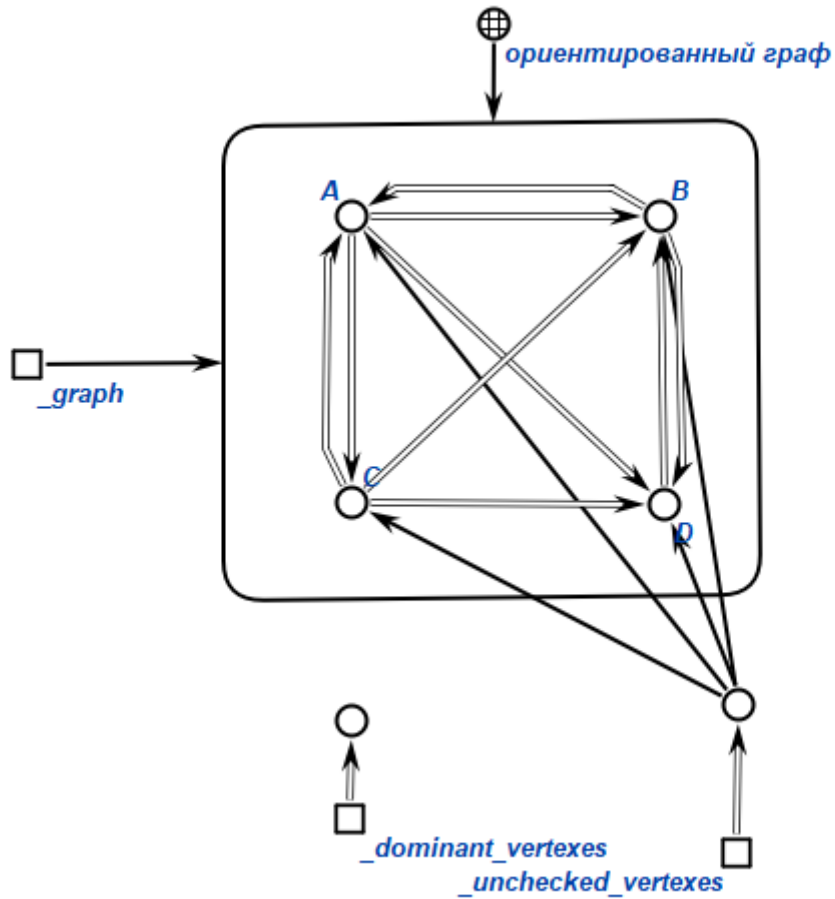


Выход:

Доминирующих вершин не существует. Программа должна вернуть ошибку вызывающему контексту.

3 Описание алгоритма

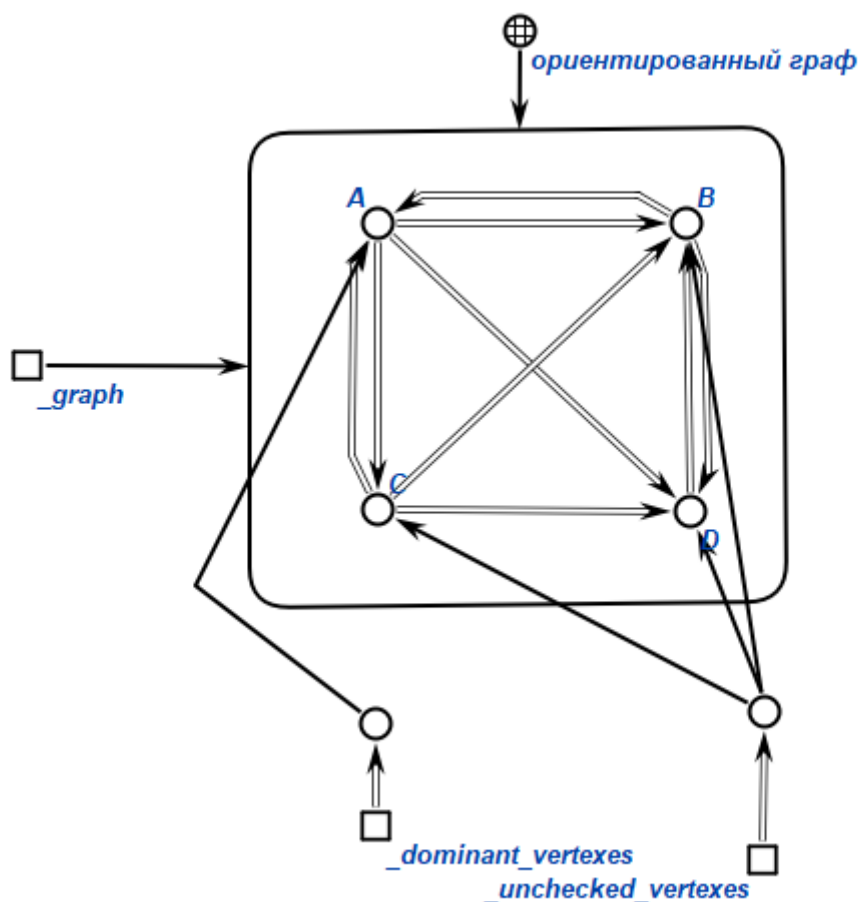
1. Задание входного графа, множества непроверенных вершин и множества доминирующих вершин



Переменные изменятся следующим образом:

- *_graph* получит в качестве значения sc-узел ориентированного графа;
- *_unchecked_vertexes* получит в качестве значения множество непроверенных вершин обрабатываемого графа
- *_dominant_vertexes* получит в качестве значения множество доминирующих вершин, которые будут добавляться по мере работы алгоритма

2. Проверка вершины A



По мере работы алгоритма необходимо проверить все вершины графа: пока множество непроверенных вершин не станет пустым. Порядок проверки вершин не важен.

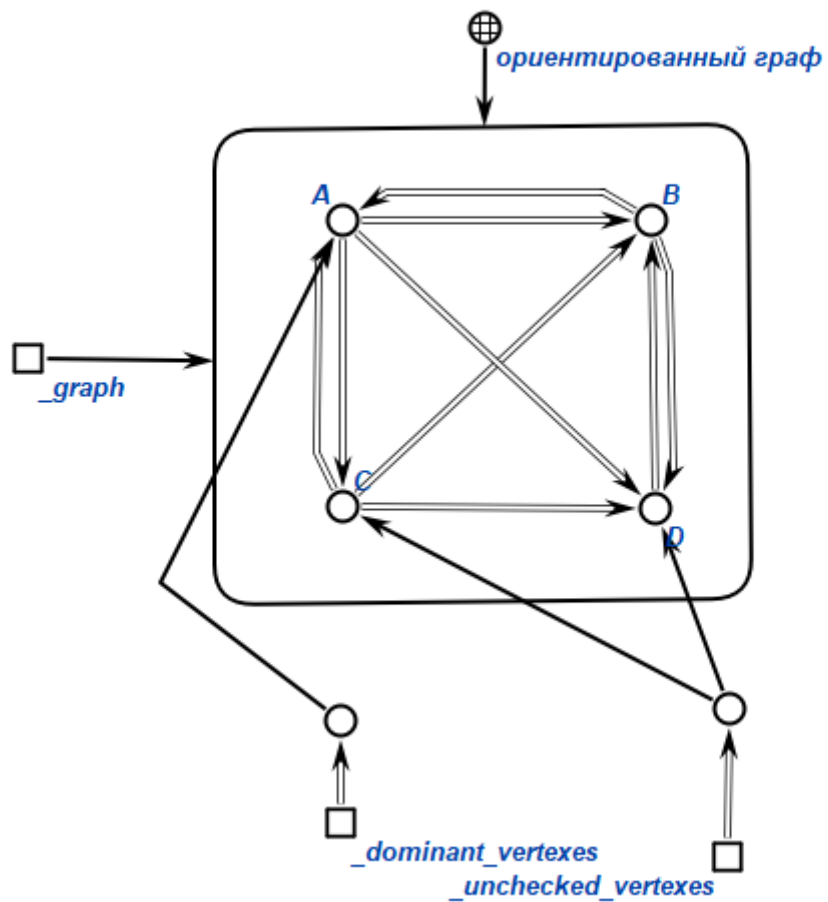
Проверим первой вершину A. Нас интересует количество ребер, для которых она является начальной: таких ребер 3.

По определению вершина является доминирующей, если ее степень равна $n-1$, если n – порядок графа. Порядок проверяемого графа: 4. По определению вершина A является доминирующей.

Добавляем вершину A во множество доминирующих вершин.

Удаляем вершину A из множества непроверенных вершин.

3. Проверка вершины В



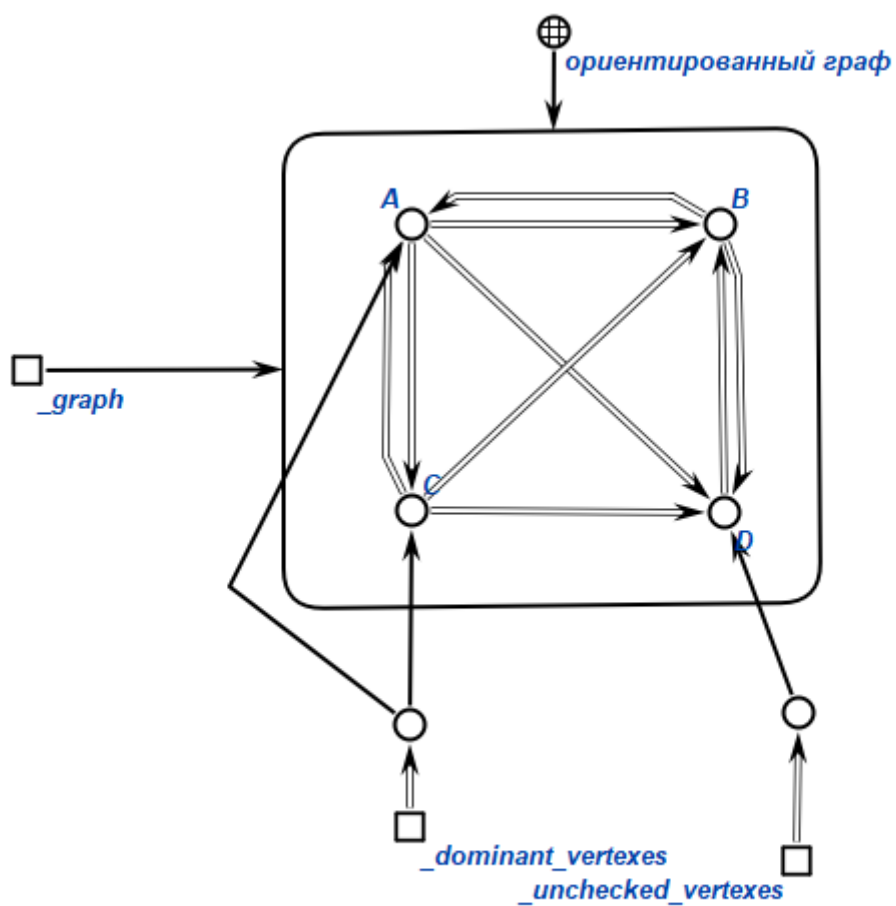
Вершина В является начальной для 2 ребер.

Порядок проверяемого графа 4.

По определению вершина В не является доминирующей.

Удаляем вершину В из множества непроверенных вершин.

4. Проверка вершины С



Вершина С является начальной для 3 ребер.

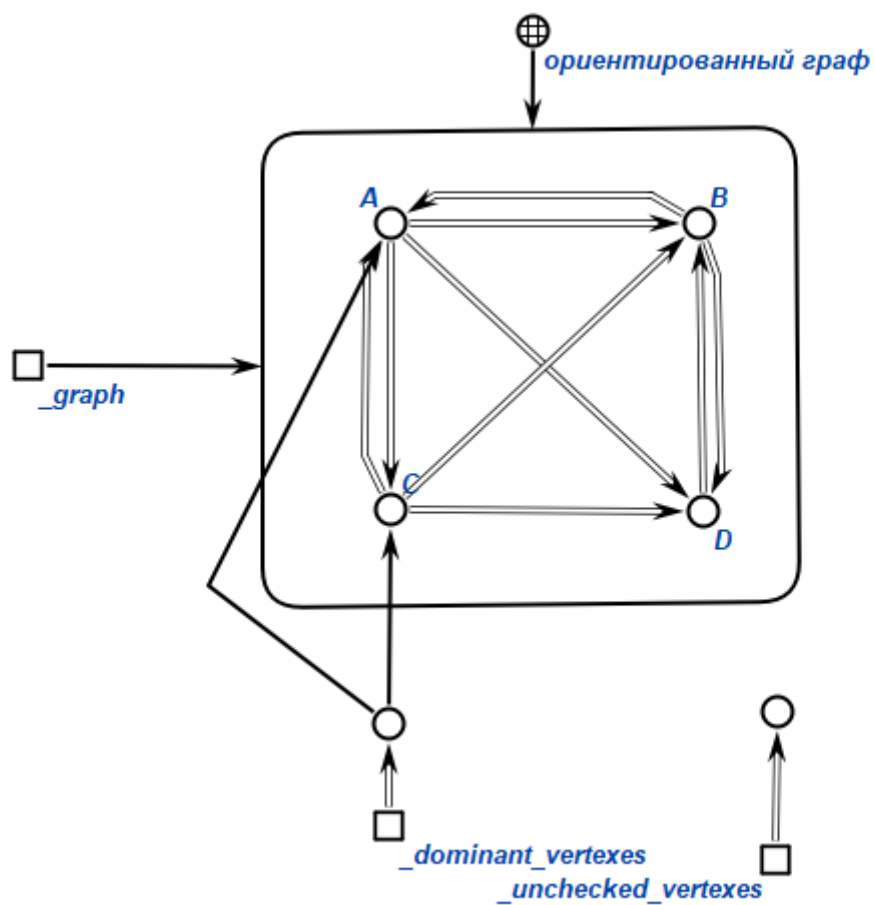
Порядок проверяемого графа 4.

По определению вершина С является доминирующей.

Удаляем вершину С из множества непроверенных вершин.

Добавляем вершину С во множество доминирующих вершин.

5. Проверка вершины D



Вершина D является начальной для 1 ребра.

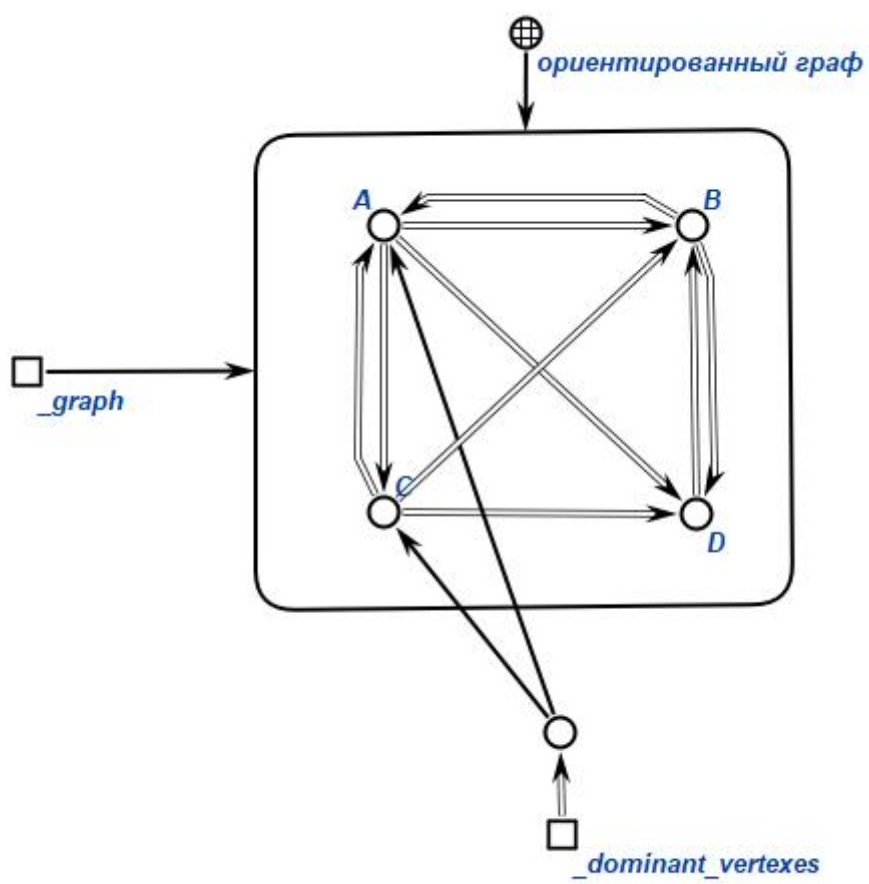
Порядок проверяемого графа 4.

По определению вершина D не является доминирующей.

Удаляем вершину D из множества непроверенных вершин.

Множество `_unchecked_vertexes` пустое, значит, мы проверили все вершины в графе.

6. Удаление множества `_unchecked_vertexes`. Результат работы алгоритма



4 Список литературы

OSTIS GT [В Интернете] // База знаний по теории графов OSTIS GT. - 2011 г.. - http://ostisgraphstheo.sourceforge.net/index.php/Заглавная_страница.

Харарри Ф. Теория графов [Книга]. - Москва : Едиториал УРСС, 2003.