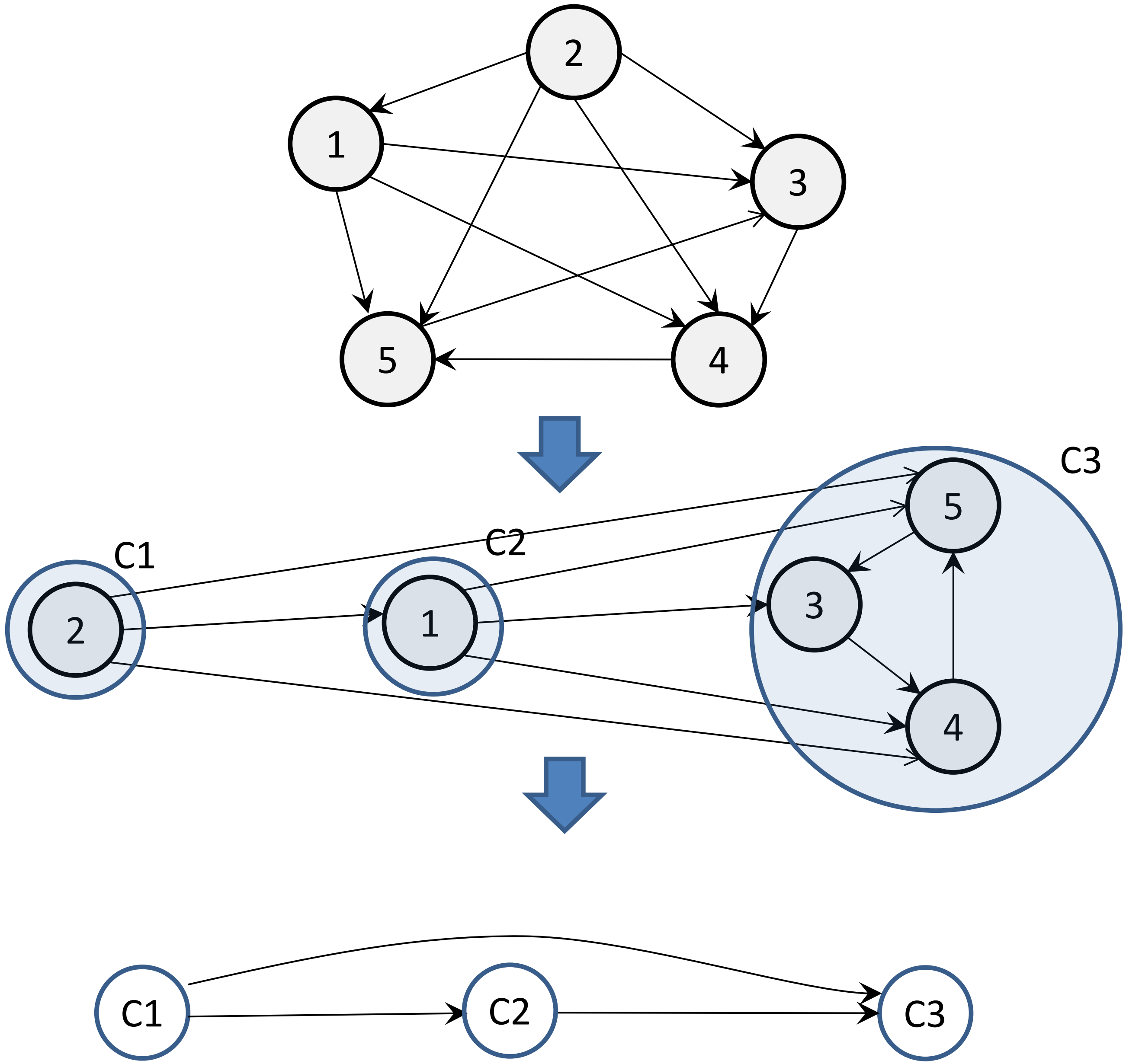
Задача 8 – Магические порталы

1. Найдём компоненты сильной связности графа в порядке топологической сортировки и построим конденсацию графа.



Алгоритм поиска компонентов сильной связности*

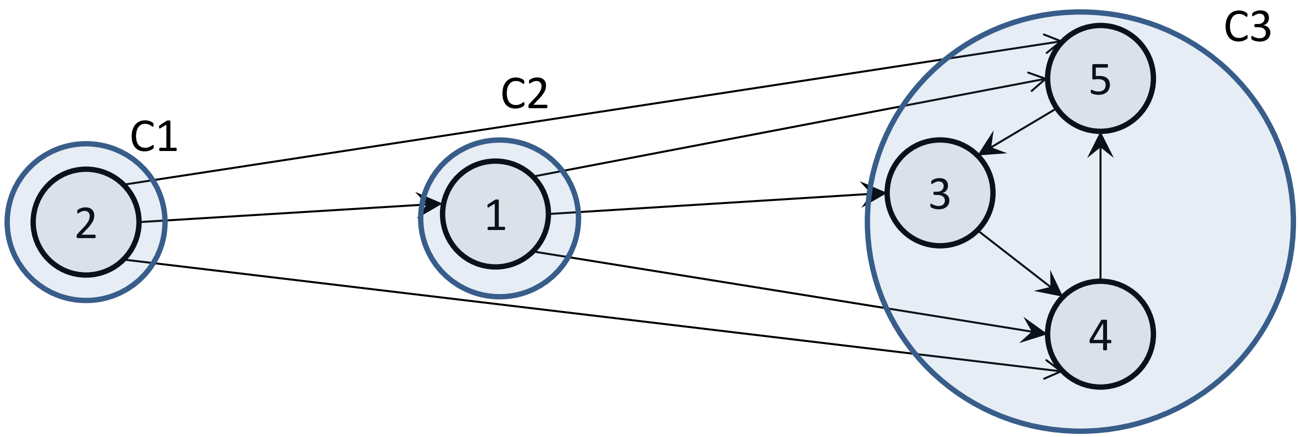
1). Запустить серию обходов в глубину исходного графа , которая возвращает вершины в порядке увеличения времени выхода *tout*, т.е. некоторый список *order*.

2). Построить транспонированный граф. Запустить серию обходов в глубину этого графа в порядке, определяемом списком (т.е. в порядке уменьшения времени выхода).
Каждое множество вершин, достигнутое в результате очередного запуска обхода, и будет очередной компонентой сильной связности.

Компоненты автоматически получаются в порядке топологической сортировки

* http://e-maxx.ru/algo/strong_connected_components

Решение подзадач 1 - 3



Пусть c_1, c_2, … – размеры компонент C_1, C_2, …
Начальное количество совершенных городов равно c_1.

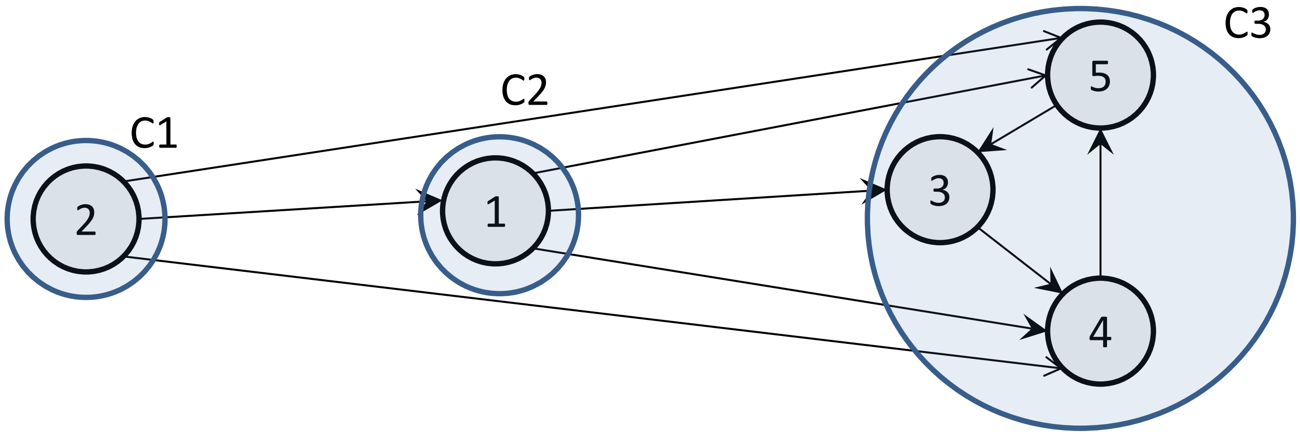
Это количество может увеличиться только при развороте ребра, начинающегося в C_1 и заканчивающегося в C_i, где $i > 1$.

- **Частный случай:** $i = 2$, в компонентах C_1 и C_2 по одной вершине. Компоненты «меняются ролями», остаётся одна совершенная вершина
- **Общий случай.** Совершенными стали все вершины в компонентах от 1 до i-й.

  Их количество = $c_1 + c_2 + \ldots + c_i$

  Число вариантов рёбер = $c_1 \times c_i$

Решение подзадачи 4



Число совершенных вершин может уменьшиться только при развороте ребра, начало и конец которого лежат в C_1

Исходный граф является **турниром** (существует ребро между любыми двумя вершинами). Тогда и компонента C_1 тоже будет турниром.

Известно, что в любом полносвязном турнире существует **гамильтонов цикл** (теорема Редеи - Камиона). Нужно его найти.

Построение гамильтонова цикла для решения подзадачи 4 *

Доказательство теоремы Редеи – Камиона является конструктивным – то есть, по сути, даёт алгоритм поиска гамильтонова цикла.

Основные этапы:

1. Строим цикл длины 3
2. На каждом шаге цикл длины k превращаем в цикл длины (k + 1)

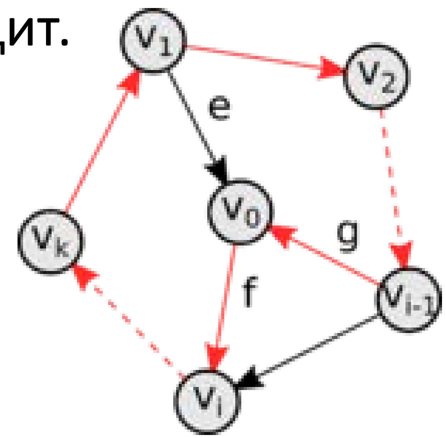
Этап 1. Как построить цикл длины 3? Берём любую вершину u. Всегда найдутся такие вершины w_2, w_1, что $u–w_2–w_1–u$ - цикл.

Этап 2. Возможны два отдельных случая. Рассмотрим их подробнее…

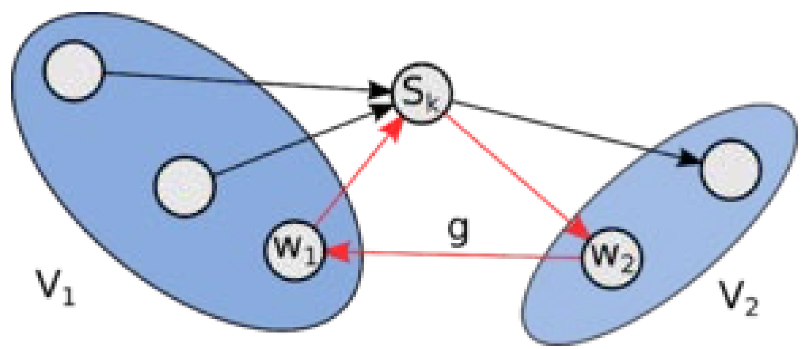
* *Возможны и другие алгоритмы поиска гамильтонова цикла – например, модифицированный dfs с использованием стека и последующими преобразованиями результатов*

Увеличение цикла длины k до длины (k + 1)

Первый случай – найдётся такая вершина v_0, что существует входящее в неё ребро из какой-то вершины цикла и, аналогично, исходящее. Тогда найдутся и две **соседних** вершины в цикле, такие что из одной ребро выходит в v_0, а в другую входит.



Второй случай



$$S_{k+1} = v_1 \to w_2 \to w_1 \to v_2 \to \cdots \to vk \to v_1$$

*Рисунки с http://neerc.ifmo.ru/wiki/index.php?title=Теорема_Редеи-Камиона

Структуры данных, позволяющие получить сложность $O(n^2)$ *

- Целочисленный массив cycle[n] – список вершин цикла
- Булевский массив c[n] – множество вершин цикла

- Булевский массив v1[n] – множество вершин, не принадлежащих циклу, из которых выходят рёбра в вершины цикла

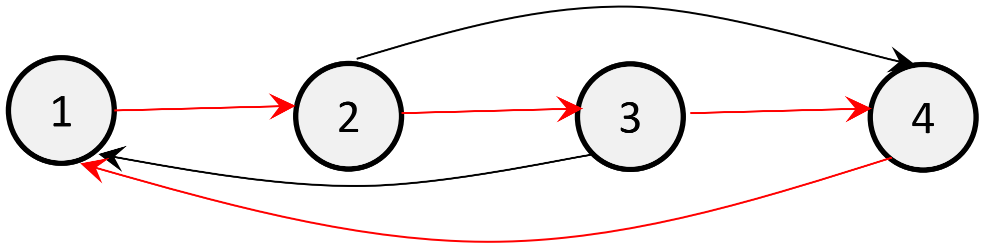
- Булевский массив v2[n] – множество вершин, не принадлежащих циклу, в которые приходят рёбра из вершин цикла

- Целочисленный массив from_v2[n]. Здесь from_v2[i] – это количество дуг, которые приходят в вершину i из вершин из множества v2.

После каждого увеличения цикла все массивы обновляются

* *Вполне возможны и другие эффективные реализации*

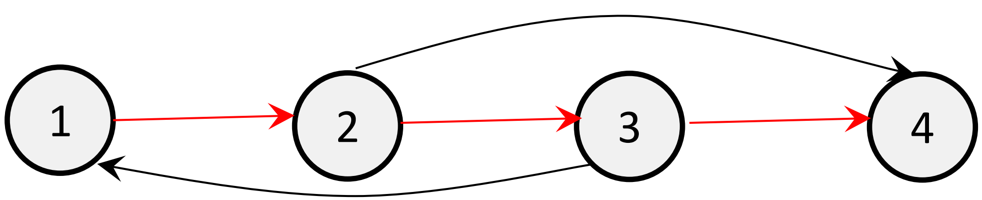
Решение подзадачи 4 - продолжение



- Разворот ребра, не принадлежащего циклу, сохраняет сильную связность компоненты и не меняет числа совершенных городов

- Разворот любого ребра $u \to v$, принадлежащего циклу, сохраняет возможность дойти из v в u, поэтому разворот можно заменить на удаление ребра.

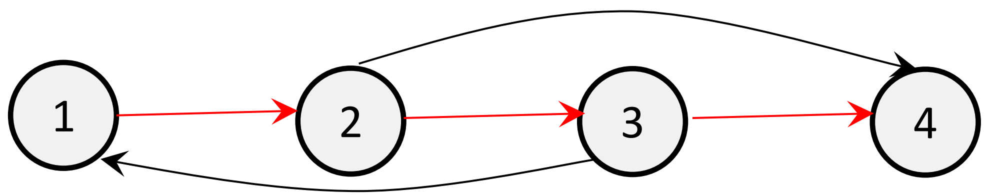
Попробуем удалить ребро $v_k \to v_1$:



Как видим, на рисунке только три вершины остались совершенными. Как находить их количество?...

Решение подзадачи 4 – продолжение

Подсчёт числа совершенных вершин при удалении ребра $v_k \to v_1$



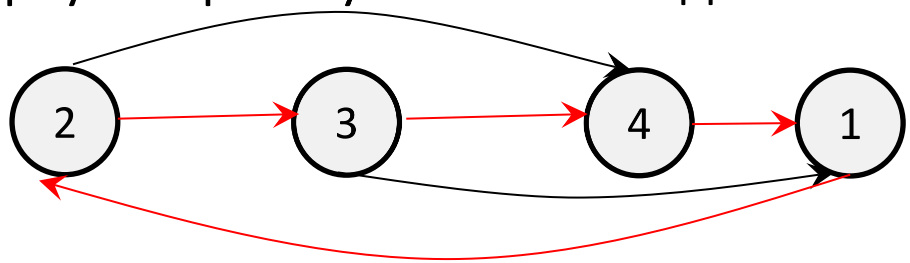
Создадим массив x[n], где x[i] – максимальный номер вершины j, такой что существует ребро $j \to i$.
Для нашего примера x[1] = 3, x[2] = 1, x[3] = 2, x[4] = 3.

Будем идти по x слева направо. Пусть right – самое большое значение в x, встреченное нами к i-му шагу.
Если right <= y, то цикл останавливается, а количество совершенных вершин равно i.

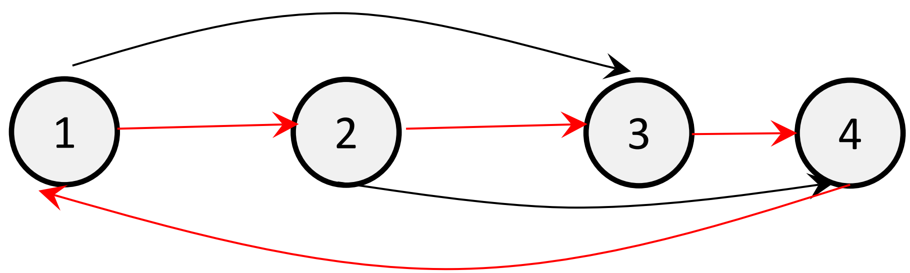
Решение подзадачи 4 – окончание

Вернём удалённое ребро на место.

Передвинем первую вершину после последней.



Перенумеруем вершины, чтобы номера снова шли слева направо:



Заметим, что массив *x* не нужно считать заново за квадрат, а можно получить из предыдущего с линейной сложностью. Для этого:

1. Сдвигаем элементы в x на единицу влево и вычитаем из них 1
2. Перебираем ребра, выходящие из новой последней вершины, и при необходимости обновляем значения в x.

Делаем такие сдвиги столько раз, чтобы каждая вершина гамильтонова цикла побывала последнем месте