Задачи стоит решать не в порядке их перечисления ниже, а по своему выбору: сначала те, которые кажутся более лёгкими. Параллельно нужно усваивать определения и смотреть ссылку http://lpcs.math.msu.su/~lyubetsky/2019/, файлы: Lyubetsky_графы**.**

Определения. Рассматриваем ориентированные графы («*структуры*»), которые состоят из цепей и циклов (включая петли, но без изолированных вершин, называемых *точками*). Дугам (=рёбрам) приписаны имена – натуральные числа. Два графа *a* и *b* имеют *равный состав*, если множества имён у *a* и *b* совпадают; иначе они имеют «неравный состав» – дуги, присутствующие лишь в *a* (лишь в *b*), называются *a-особыми* (*b-особыми*), а прочие дуги – *общими*.
Операции над графом (подробнее на той же ссылке): *двойная переклейка* – разрезать две вершины и по-новому отождествить (=склеить) четыре образовавшихся края; *полуторная переклейка* – разрезать вершину и по-новому склеить один образовавшийся край с каким-то свободным краем в графе; *одинарная переклейка* – склеить два свободных края и *разрез* – разрезать вершину. Операции обратимы. Для *неравного состава* добавляются операции: *удалить* из графа связный отрезок из *a*-особых дуг и *вставить* в граф связный отрезок из *b*-особых дуг («отрезки»); если у удаляемого отрезка были соседние дуги, то их края склеиваются. Максимальный по включению отрезок называют *блоком*. Первые четыре операции называются *стандартными*, последние две – *дополнительными*.
Общий граф a+*b* структур *a* и *b* с равным составом – неориентированный граф без петель, вершины которого – имена краёв всех дуг; например, начало дуги с номером 3 имеет имя 3_1, конец – имя 3_2. Ребро общего графа соединяет две вершины, если соответствующие им края склеены в *a или* в *b*; оно помечается как *a*- или *b*-ребро (*обычное* ребро в *a*+*b*).
В случае неравного состава *общий граф a*+*b* имеет следующие вершины: *обычные* – имена краёв общих дуг; и *сингулярные* – *блоки*. Блок помечается как *a*- или *b*-вершина в *a*+*b* и ей приписывается последовательность дуг, составляющих блок. *Сингулярное* ребро в *a*+*b* соединяет обычную вершину с сингулярной, если в *a* или *b* край блока склеен с обычной вершиной (*a*- или *b*-ребро). *Петля* в *a*+*b* соответствует циклу, который является блоком. *Висячим* называется сингулярное ребро, инцидентное сингулярной вершине степени 1.

<u>Задача критическая.</u> Долгим рисованием понять, что сказано выше.
<u>Задача простейшая.</u> Доказать: граф *a*+*b* состоит из циклов, петель и цепей (точки считаем цепями), а если в *a* и *b* нет цепей, то только из циклов и петель. Как выглядит *a*+*b*, если *a*=*b*?
<u>Задача простая.</u> Определить над *a*+*b* аналоги операций, кроме вставки, над графами *a* и *b*, так, чтобы для любой такой операции *o* над *a* и её аналога *o** выполнялось *o**(*a*+*b*)=*o*(*a*)+*b*; и аналогично для *b*. Почему трудно определить аналог вставки? Таким образом, *над общим графом определяются 5 операций*: стандартные и удаления сингулярной вершины.
<u>Задача простая.</u> Рассмотрим задачу преобразования: найти *кратчайшую* по числу операций цепочку операций, преобразующую данные графы *a* в *b*. Для равного состава свести её к задаче приведения: привести *a*+*b* к *финальному* виду (= виду *c*+*c*) кратчайшей цепочкой операций. Отсюда получить линейный по времени алгоритм решения задачи преобразования для равного состава. Какие трудности встретятся при обобщении этого алгоритма на неравный состав?
<u>Задачи простые.</u> Привести общий граф к финальному виду операциями двойной переклейки и удаления сингулярной вершины при одном из следующих <u>условий на цены</u>:
1) цены операций равны 1 (= все цены равны, т.е. без цен);
пусть цена двойной переклейки равна 1 и
2) цена удаления равна 0.5;
3) цена удаления 1.5;
4) цена удаления 2.5 – все перечисленные варианты называют «стационарными»;
5) Цены двойной переклейки и удаления *a*-вершины 1, цена удаления *b*-вершины равна 1.5.

Граф задан ниже последовательностью *a*- и *b*-рёбер в каждой компоненте и состоит только из циклов и петель (циклический случай).
В пунктах 1–5 один цикл:
1. (*abab*),
2. (*aabab*),
3. (*aabbab*),
4. (*aabbaab*),
5. (*aabbaabb*).
Последняя задача для третьего и четвёртого вариантов цен более сложная.
6. цикл (*abb*) и *b*-петля.
7. два цикла (*baa*).
8. два цикла (*abb*) и (*aabb*).
9. два цикла (*aabb*).
10. цикл (*aabbaabbaabb*).

Задачи простые*. Привести общий граф к финальному виду четырьмя стандартными операциями и операцией удаления сингулярной вершины при одном из следующих условий на цены:
1) Цены всех операций равны 1 (= все цены равны, т.е. без цен).
2) Цены стандартных операций равны 1, цена удаления равна 0.5.
В квадратных скобках указаны цепи, их висячие рёбра подчёркнуты.
1. $[abbaa]$.
2. $[\underline{a}bb\underline{a}]$.
3. $[bbaa\underline{b}]$.
4. $[\underline{b}aabb\underline{a}]$.
5. $[aa\underline{b}]$ и $[bb\underline{a}]$.
6. $[bbaabb]$ и сингулярная *b*-точка.
7. $[\underline{b}aa]$ и $[\underline{a}bb\underline{a}]$.
8. $[bb\underline{a}]$ и $[aa]$.
9. $[\underline{ab}]$ и $[aabb]$.
10. $[\underline{ab}]$ и $[bb\underline{a}]$.
11. две цепи $[aa\underline{b}]$.
12. $[aa\underline{b}]$, $[aabb]$ и сингулярная *a*-точка.
13. две цепи $[aa\underline{b}]$ и сингулярная *a*-точка.

Задача средней трудности. *Размером* общего графа назовём сумму в нём числа обычных рёбер с половиной числа сингулярных невисячих рёбер и минус число сингулярных *точек*. Для обычных точек и петель размер считаем равным 0, для сингулярных точек – равным "минус 1". Таким образом, размер общего графа – сумма размеров его компонент, а *размером компоненты* назовём сумму в ней числа обычных рёбер с половиной числа сингулярных невисячих рёбер. Доказать: размер общего графа – целое число. Установить, как меняется размер графа после каждой из пяти операций. Для какой из операций изменение размера графа зависит от её аргумента? В каком случае?
Задача средней трудности. Алгоритм «*автономного приведения*» общего графа к финальному виду состоит в следующем. Из компонент (кроме циклов длины 2 или 3) «вырезаются» обычные рёбра и замыкаются в циклы длины 2 (двойная или полуторная переклейка соседних рёбер/соседнего ребра). Затем цепи (имеющие не менее двух обычных вершин) замыкаются в циклы (склейкой или полуторной переклейкой), циклы разбиваются на циклы размера 2 (двойной переклейкой). В конце удаляются сингулярные вершины и петли. Описать этот алгоритм подробнее. Показать, что в случае равного состава алгоритм строит кратчайшую цепочку. Алгоритм, который находит абсолютный минимум функционала (здесь суммарной цены цепочки, преобразующая *a* в *b*), называется *точным*.

Задача повышенной трудности. Выразить число операций $A(a+b)$ в алгоритме автономного приведения через характеристики исходного графа $a+b$ (ниже приведён ответ к этому заданию). Доказать: если для любой операции o, применённой к произвольному общему графу G, выполнено «*неравенство треугольника*» $A(G)–A(o(G)) \le 1$, где $o(G)$ – результат применения o к G, то алгоритм точен. Пусть в a и b нет цепей (*циклический случай*). Проверить неравенство треугольника для различных операций, доказывая, таким образом, *точность алгоритма автономного приведения для циклического случая* и равных цен.

Определение. Пусть каждой операции приписана цена – строго положительное число. Цепочку с минимальной суммарной ценой операций, преобразующую a в b или приводящую $a+b$ к финальному виду, по-прежнему называем *кратчайшей*. *Автономной ценой* $A(G)$ общего графа G назовём суммарную цену цепочки операций, выдаваемой алгоритмом автономного приведения. *Преобразованием* графа G назовём цепочку s операций, которые последовательно применяются, начиная с G. Точнее, s – цепочка операций с подставленными в них аргументами (т.е. каждый раз указано, к каким аргументам применять операцию). *Качеством* $P(s)$ преобразования s назовём величину $P(G,s)=A(G)–A(s(G))–c(s)$, где $s(G)$ – граф, полученный после применения всей цепочки, начиная с G, а $c(s)$ – суммарная цена операций в s. Число $P(s)$ показывает, сколько экономится в результате применения s по сравнению с автономным приведением G, т.е. без использования s. *Взаимодействием* называется преобразование, для которого $P(G,s) > 0$, $\forall G$, где G пробегает какое-то непустое множество $G(s)$ (хотя бы из одного элемента); множество $G(s)$ нужно указать. Конечно, s лучше, если $G(s)$ – бесконечное и просто описанное множество.

Важная задача. Указать хотя бы одно взаимодействие, хотя бы такое, для которого $G(s)$ из одного элемента. Эту задачу поможет решить следующая задача.

Пусть цены стандартных операций равны 1, а $w_a>0$ и $w_b>0$ – цены удаления сингулярных a- и b-вершин, и $w_a \le w_b$ (другой случай симметричен). Пусть $w=w_a=w_b$ – цена удаления сингулярной вершины. Пусть d – размер графа G, f – число цепей нечётного размера, c – число циклов (но не петель); B – суммарное число сингулярных вершин в нём; S – сумма целых частей половин длин максимальных связных участков из обычных рёбер («*отрезков*») сложенная с числом крайних (на цепи) нечётных (по длине) отрезков и затем минус число циклических отрезков; и D – сумма дефектов компонент. Здесь *дефект* сначала определяется для компоненты без обычных рёбер, и он равен 1 для цепи нечётного строго положительного размера с нулём или одним висячим ребром или чётного ненулевого размера без висячих рёбер. Он равен 0 для других цепей и циклов, петель. Для компонент с обычными рёбрами нужно сначала их вырезать (двойная или полуторная переклейка соседних рёбер/соседнего ребра); D суммируется по компонентам. Пусть K_b – число компонент графа, которые содержат сингулярную b-вершину.

Важная задача. Для «*стационарного*» случая (т.е. $w_a = w_b$) автономная цена $A(G)$ общего графа G равна

$$A(G) = (1–w)(0.5d+0.5f–c)+w(B+S+D).$$

Для «*нестационарного*» случая (т.е. $w_a \ne w_b$) автономная цена $A(G)$ графа G равна

$$A(G)=(1–w_a)(0.5d+0.5f–c)+w_a(B+S+D)+(w_b–w_a)K_b.$$

Задача критическая. На собственных рисунках и примерах понять определения и формулы.

Задача средней трудности. Обобщить на случай произвольных цен операций следующее неравенство треугольника: для любой операции o выполняется $T(G)–T(o(G))\le 1$, где $T(G)$ – длина цепочки операций некоторого алгоритма, приводящей G к финальному виду. Доказать: из неравенства треугольника следует точность этого алгоритма при некотором тривиальном дополнительном условии (каком?). Верно ли обратное, т.е. что если алгоритм точен, то для любой операции *o* выполняется неравенство треугольника?

Определение. Цепь нечётного (чётного) размера назовём *нечётной* (соответственно, *чётной*); нуль считаем чётным числом, а –1 – нечётное число. Если в цепи имеются обычные рёбра, то *тип* цепи определяется после их вырезания (в любом порядке). Для цепи без обычных рёбер тип определяется следующим образом. $1a$ – нечётная цепь с одним висячим b-ребром; $2a$ – нечётная цепь с двумя висячими b-рёбрами или сингулярная b-точка; в первом случае говорим о типе $2a^*$, во втором – о типе $2a'$; $3a$ – нечётная цепь без висячих рёбер с двумя крайними a-рёбрами (тип $3a^*$) или одинарное a-ребро (тип $3a'$); 1_a – чётная цепь с одним висячим a-ребром размера строго больше 0 (тип 1^*_a) или размера 0 (тип $1'_a$); 2 – чётная цепь с двумя висячими рёбрами размера строго больше 0 (тип 2^*) или размера 0 (тип $2'$); 3 – чётная цепь без висячих рёбер размера строго больше 0. Аналогично определяются типы с заменой a на b. Тип 1 – объединение типов 1_a и 1_b, тип 0 – цепь без сингулярных вершин. Циклы делим на (a,b)-циклы, a-циклы, b-циклы и 0-циклы в зависимости от наличия в них сингулярных вершин обоих типов, только одного или их отсутствия. Аналогичные типы применяются для цепей и петель.

Задача критическая. Нарисовать примеры компонент указанных типов.
Задача простая. Показать, что тип цепи не зависит от порядка, в котором вырезаются обычные рёбра.
Задача средней трудности. Пусть цена всех стандартных операций равна 1, цена удаления вершины равна $w>0$ (напомним: такое соотношение цен называется *стационарным*). Обосновать формулу для $A(G)$, приведённую выше.
Задача повышенной трудности. Пусть цена всех стандартных операций равна 1, цена удаления a-вершины равна $w_a>0$, цена удаления b-вершины равна некоторому $w_b>0$ (напомним: такое соотношение цен называется *нестационарным*). Уточнить алгоритм автономного приведения для нестационарного соотношения цен. Обосновать формулу для $A(G)$, приведённую выше.
Задача средней трудности. В задаче, помеченной (*) на стр. 2, указать взаимодействия (если есть) и их качества.
Задача средней трудности. Указать взаимодействие, применяемое к одному циклу. Вычислить его качество в нестационарном случае.
Задачи средней трудности. Привести общий граф к финальному виду кратчайшей цепочкой, исследовать различные случаи стационарного и нестационарного соотношения цен, указать использованные взаимодействия и их качества для графов, состоящих из цепей (можно взять примеры указанных типов):

1) цепи типов $1a$, $1a$, $1b$, 2, и 3;
2) цепи типов $1a$, $1b$, $2a$, $2b$, 2, 3, 1_b;
3) цепи типов $1b$, $1b$, $2a$, $2b$, 2, 3, 1_a;
4) цепи типов $1a$, $1a$, $2a$, $3b$, 1_a, 1_b;
5) цепи типов $2a$, $2a$, $2b$, 3, 3, 3, 3;
6) цепи типов $3b$, $3b$, 1_b, 1_b;
7) цепи типов $2b$, $3a$, $3a$, $3b$, 2, 1_a, 1_b, 1_b;
8) цепи типов $1a$, $1a$, $1a$, $2b$, $3b$, $3b$, 2.

Задача повышенной трудности. При стационарном соотношении цен перебрать пары типов компонент и выяснить, между какими парами возможно взаимодействие. Выразить через w_a и w_b качества найденных взаимодействий для нестационарного соотношения цен. Придумать взаимодействие при $w_b>w_a$, которое является только преобразованием при $w_a=w_b$.
Задача повышенной трудности. Указать взаимодействия между тремя цепями, четырьмя цепями и между циклом и двумя цепями. Эти взаимодействия являются композицией двух

или трёх операций, указать их. «Композиция» понимается с одной особенностью: для каждой операции указано, к каким компонентам предыдущего графа она применяется. Какие могут быть причины рассматривать такие «составные» взаимодействия? Вычислить их качество в нестационарном случае.

Трудные задачи для научной работы.

1) Пусть структуры a и b с равным составом, и обе состоят из одной циклической компоненты; разрешена лишь операция двойной переклейки. Доказать: существует кратчайшая цепочка, преобразующая a в b, и такая, что в ней любой промежуточный граф состоит не более чем из двух циклических компонент; если при некоторой операции в цепочке графов возникает вторая компонента, то следующая операция оставляет лишь одну компоненту. Отсюда получить алгоритм, строящий описанную кратчайшую цепочку и работающий не более чем квадратичное время.

2) Пусть a и b с равным составом, и обе из одной линейной компоненты. Доказать: существует кратчайшая цепочка, преобразующая a в b, и такая, что в ней любой промежуточный граф содержит не более одной линейной компоненты; если при некоторой операции в цепочке графов возникает циклическая компонента, то следующая операция оставляет одну линейную. Отсюда получить алгоритм, строящий описанную кратчайшую цепочку и работающий не более чем квадратичное время. Показать: в условиях задачи не всегда удаётся преобразовать a в b минимальной цепочкой так, чтобы и все промежуточные графы имели лишь линейные компоненты.

3) Пусть a и b с равным составом, и обе содержат лишь линейные компоненты. Доказать: существует кратчайшая цепочка, преобразующая a в b, такая, что в ней любой промежуточный граф содержит не более одной циклической компоненты; если при некоторой операции в цепочке графов возникает циклическая компонента, то следующая операция оставляет только линейные. Отсюда получить алгоритм, строящий описанную кратчайшую цепочку и работающий не более чем квадратичное время.

Задача на учёную степень. Усилить результаты трёх последних задач, в частности, обобщить их на неравный состав.