Лабораторная работа 6
«Паросочетание в двудольном графе»

Пусть задан неориентированный двудольный граф G с долями V_1 и V_2. Подмножество M множества ребер $E(G)$ графа называется паросочетанием, если в нем нет двух ребер с общей концевой вершиной. Требуется реализовать алгоритм, который находит в графе G паросочетание с максимальным числом ребер.

Алгоритм

Вход. Двудольный граф G с долями V_1 и V_2.

Выход. Наибольшее паросочетание графа G.

Шаг 1. По неориентированному графу G построим направленный граф G^* следующим образом: в графе G каждое ребро $\{u, v\} \in E(G)$, где $u \in V_1$ и $v \in V_2$, заменим на дугу (u, v).

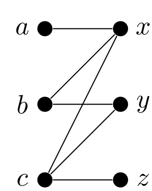
Шаг 2. Добавим к графу G^* две вершины s и t. Включим в граф G^* дугу (s, u) для каждой вершины $u \in V_1$, а также дугу (v, t) для каждой вершины $v \in V_2$.

Шаг 3. С помощью обхода графа (поиска в ширину или поиска в глубину) определим достижима ли в графе G^* вершина t из вершины s и в случае положительного ответа найдем в этом графе (s, t)-путь P.

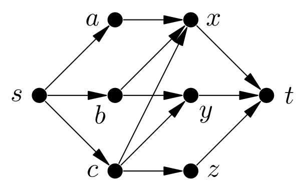
Шаг 4. Если вершина t не достижима из вершины s в графе G^*, то положим $M = \emptyset$ и для каждой дуги (v, u) графа G^* такой, что $v \in V_2$ и $u \in V_1$ добавим в множество M ребро $\{u, v\}$ графа G; получившееся в результате множество M является наибольшим паросочетанием графа G. СТОП. Алгоритм завершает свою работу.

Шаг 5. Пусть в графе G^* вершина t достижима из вершины s. Скорректируем граф G^* следующим образом: в графе G^* первую и последнюю дуги пути P удалим, а остальным дугам этого пути поменяем ориентацию на противоположную. Переходим на Шаг 3.

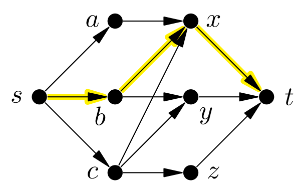
Пример. Рассмотрим двудольный граф G



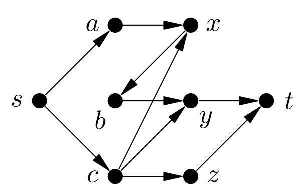
с долями $V_1 = \{a, b, c\}$ и $V_2 = \{x, y, z\}$. Построим направленный граф G^*. Граф G^* получается из графа G ориентацией всех ребер слева направо и добавлением двух вершин — вершины s с выходящими из нее дугами с концами в вершинах множества V_1 и вершины t с входящими в нее дугами с началами в вершинах множества V_2



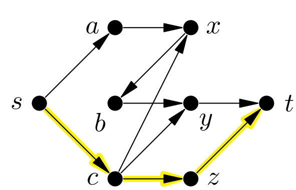
Найдем в графе G^* (s, t)-путь



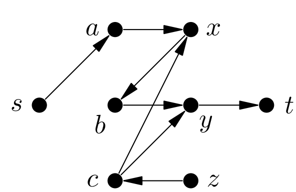
Рассмотрим в графе G^* дуги найденного пути. Первую и последнюю дуги пути удалим, а промежуточным дугам, а именно дуге (b, x), поменяем ориентацию



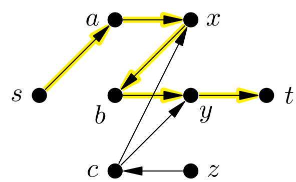
Найдем (s, t)-путь



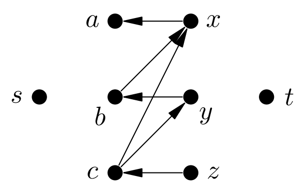
Крайние дуги пути удалим, а промежуточные дуги пути (в этом случае единственную дугу (c, z)) заменим на обратные



Найдем (s, t)-путь



Крайние дуги пути удалим, а промежуточные дуги пути обратим



В получившемся графе нет (s, t)-путей. Алгоритм завершает свою работу. В результирующем графе G^* нас интересуют дуги, которые начинаются в вершинах множества V_2 и заканчиваются в вершинах множества V_1

$$(x, a), (y, b), (z, c).$$

Этим дугам соответствуют ребра в исходном графе G

$$\{a, x\}, \{b, y\}, \{c, z\}.$$

Совокупность этих ребер представляет собой наибольшее паросочетание графа G.