Домашнее задание

Преамбула. Приведём базовые определения из теории графов, которые используются в формулировках задач, и которые нужно правильно использовать при их решении.

Степенью вершины u (неориентированного графа) называют количество рёбер $\deg(u)$, смежных с этой вершиной. В ориентированном графе *исходящей степенью* $\deg_+(u)$ вершины u называют количество рёбер, исходящих из u (u — левый конец), а *входящей степенью* $\deg_-(u)$ называют количество рёбер, входящих в u (u — правый конец).

Маршрутом называется последовательность вершин $v_1, \ldots, v_n$, каждая соседняя пара вершин которой соединена ребром. Маршрут называется *(простым) путём*, если все вершины различны. *Длина маршрута* — это количество рёбер между вершинами пути, т.е. $n-1$. *Замкнутый маршрут* — это путь, у которого первая и последняя вершина совпадают. *Цикл* — это маршрут, все вершины которого различны (за исключением первой и последней).

Ориентированный граф называют *турниром*, если между каждой парой его вершин есть ровно одно ребро. Также такие графы называют *полными* ориентированными графами.

DAG — это ориентированный ациклический граф (Directed Acyclic Graph).

Неориентированный граф называется *двудольным*, если множество его вершин разбивается на два множества $V = L \cup R$, $L \cap R = \varnothing$, таких что один конец каждого ребра лежит в L, а другой в R. Эти множества называются левой и правой долями соответственно. *Паросочетание* в двудольном графе — такое подмножество рёбер E', что правые и левые концы любых двух рёбер из этого множества попарно различны. То есть, если $(u, v), (a, b) \in E'$ и $u, a \in L$ и $v, b \in R$, то $u \neq a$, а $v \neq b$.

1. На вход задачи подаётся граф G и его верины s и t. Постройте алгоритм, который за время $O(|V| + |E|)$ проверяет, что вершина t достижима из вершины s. Решите задачу как в случае, когда G неориентированный граф, так и в случае, когда G ориентированный граф.

2. Докажите, что каждый турнир на n вершинах содержит (простой) путь длины $n-1$. Постройте алгоритм, который получив на вход турнир, находит в нём такой путь, и оцените асимптотику его времени работы.

3. В графе G был проведён поиск в глубину. Время открытия и закрытия вершин сохранено в массивах d и f. Постройте алгоритм, который используя только данные из массивов d и f (и описание графа) проверяет, является ли ребро e графа G **а)** прямым ребром; **б)** перекрёстным ребром. *См. определения в Кормене (глава про поиск в глубину).*

4. В государстве между n городами есть m односторонних дорог. Было решено разделить города государства на наименьшее количество областей так, чтобы внутри каждой области все города были достижимы друг из друга.

1. Предложите эффективный алгоритм, который осуществляет такое разделение, докажите его корректность и оцените асимптотику.

2*. Государство решило добиться того, чтобы из каждого города можно было добраться до каждого. В силу бюджетных ограничений, было решено построить минимальное число односторонних дорог (не важно какой длины), необходимое для достижения этой цели. Предложите алгоритм, решающий задачу.

5. Вам нужно выбраться из лабиринта. Вы не знаете, сколько в нем комнат, и какая у него карта. По всем коридорам можно свободно перемещаться в обе стороны, все комнаты и коридоры выглядят одинаково (комнаты могут отличаться только количеством коридоров).

Пусть m - количество коридоров между комнатами. Предложите алгоритм, который находит выход из лабиринта (или доказывает, что его нет) за $O(m)$ переходов между комнатами. В вашем расположении имеется неограниченное количество монет, которые вы можете оставлять в комнатах, причем вы знаете, что кроме ваших монет, никаких других в лабиринте нет, и вы находитесь в нем одни.

6. Дан орграф на n вершинах ($V = \{1, \dots, n\}$), который получен из графа-пути (рёбра, которого ведут из вершины i в $i+1$) добавлением ещё каких-то m данных ребер. Найдите количество сильно связных компонент за $O(m \log m)$.

7. На вход задачи поступает описание двудольного графа $G(L, R, E)$, степень каждой вершины которого равна двум. Необходимо найти максимальное паросочетание в G (которое содержит максимальное количество рёбер). Предложите алгоритм, решающий задачу за $O(|V| + |E|)$.

8. Все степени вершин в неориентированном графе равны $2k$. Все его ребра покрашены в несколько цветов. Предложите $O(V + E)$ алгоритм, который находит в этом графе эйлеров цикл, в котором цвета всех соседних ребер разные (либо выводит, что такого цикла нет).