

Homework 8

Dimitrov Blagoi

2019-05-05

Задание 2

Используем алгоритм нахождения ближайшего общего предка. Мы разделим каждый запрос (a_i, b_i) на два вертикальных пути, чтобы было проще с ним работать.

Для каждой вершины будем хранить число, изначально оно нуль. Для каждой вершины увеличиваем значение на количество путей в которых она является концом, и уменьшаем на количество в которых - началом. Таким образом получим, что для любого ребра сумма в его поддереве, т.е. в поддереве нижней вершины это количество запросов, которые проходят через это ребро.

Будем спускаться по дереву, каждый раз идя по "тяжелому" ребру, т.е. по тому, что содержит в наибольшем количестве путей (по аналогии с Long Path Decomposition).

Для всех остальных вершин, в которые можем попасть по другим ребрам - будет начат новый путь.

Теперь докажем, что наше покрытие является статически оптимальным. Рассмотрим покрытие, отличное от нашего. Рассмотрим вершину, которая принадлежит не тому пути, что в нашем покрытии (хотя бы одна такая всегда есть). Так как в нашем покрытии путь продолжился по тяжелому ребру, то в новом ребре он продолжился по легкому. Нам интересны запросы, которые проходят через эти два ребра. По нашему ребру проходит больше, или, если быть точным, не меньше запросов, чем по новому, т.к. наше - тяжелое. Следовательно число переходов не уменьшится.

Получаем, что покрытие оптимально.

■

Задание 3

Как и в предыдущей задаче, разобьем все наши запросы на вертикальные пути. Разобьем множество всех вершин на несколько так, чтобы в одном множестве находились вершины, у которых были одинаковые x :

$$x = \text{depth}(\text{vertex}) \bmod (\log_2 n)$$

Очевидно что $\exists x$: мощность его множества не больше $\frac{n}{\log_2 n}$

На этом множестве сделаем двоичный подъем с шагом в логарифм. Построить его можно за $O(\log_2 n \cdot \frac{n}{\log_2 n})$

Пусть надо подняться из вершины v в вершину u . Наш путь разделится на три этапа:

- 1) От v до нашего множества, на котором мы построили двоичный подъем.
- 2) В этом множестве.
- 3) От этого множества до u .

Ассимптотика:

- 1) $O(\log_2 n)$. Просто поднимаемся по ребрам.
- 2) $O(\log_2 n)$. Скачем двоичным подъемом.
- 3) $O(\log_2 n)$. Опять просто поднимаемся по ребрам.

Таблицу ассоциативной функции считаем аналогично двоичному подъему.

