

Артефакты

Авторы задачи и разработчики: Владимир Новиков, Максим Дмитриев

У авторов есть два совершенно разных решения, в разборе будут оба.

Сразу обозначим за $mask$ битовую маску длины k , в которой i -й бит $mask$ равен 1, если и только если мы уже когда-то посещали артефакт с номером i .

Утверждение: оптимальный обход — это сумма ребер, взятых дважды, минус сумма ребер на диаметре нашего пути. Это следует просто из рассмотрения выбранного маршрута: его можно представить как непрерывный простой путь, от которого ответвляются поддеревья, на которых каждое ребро обходится в обе стороны (чтобы обойти все дерево и вернуться на «основной» маршрут).

Первое решение — Алгоритм Дейкстры. Действительно, давайте для каждой вершины создадим 2^k её копий. Пусть $dp[i][mask]$ — минимальный ответ на задачу, если мы закончили наш путь в вершине i , предварительно посетив $mask$ артефактов. Тогда мы можем запустить алгоритм Дейкстры из всех вершин i с $mask = 0$. Теперь осталось научиться обновлять нашу $mask$.

Пусть мы стоим в вершине i с артефактом $a[i]$. Тогда обозначим $new_mask = mask \text{ or } 2^{a[i]}$, то есть маска с добавленным артефактом $a[i]$. Попытаемся обновить ответ для $dp[i][new_mask]$ через $dp[i][mask]$, после чего присвоим $mask = new_mask$ (несложно понять, что нам надо выполнять такое присваивание в любом случае). Затем пройдемся по ребрам, исходящим из вершины i , и попробуем обновить ответ для $dp[j][new_mask]$. Ответом на задачу будет минимум по всем $dp[i][2^k - 1]$. Если же мы не смогли посетить ни одну такую $mask$, то ответ на задачу равен -1 . Итоговая асимптотика будет равна $\mathcal{O}(n \cdot 2^k \cdot \log(n \cdot 2^k))$. Подметим, что в этой задаче стоит написать Дейкстру через очередь, а не через `set`.

Второе решение — динамическое программирование по подмаскам и поддеревьям. Давайте заведем $dp[i][mask][flag]$ — минимальный ответ на задачу, если дерево подвешено за вершину i и мы собрали $mask$ артефактов с некоторым условием $flag$.

- Пусть $flag = 1$. Это означает, что нам надо два раза пройти по каждому ребру (т.е. полностью обойти дерево, даже по путям вверх).
- Пусть $flag = 2$. Это означает, что нам надо два раза пройти по некоторым ребрам (т.е. полностью обойти дерево), но при этом по какому-то пути вверх мы пройдем только один раз.
- Пусть $flag = 3$. Это оптимальный ответ для пункта i , если мы уже «вычли» диаметр.

Рассмотрим как мы пересчитываем данное dp . Пусть мы стоим в вершине u и переходим в вершину v . Перебор всех различных масок и подмасок занимает 3^k , все переходы мы делаем за $\mathcal{O}(1)$. Ответ нам надо обновлять из состояний $dp[i][2^k - 1][3]$. Итого наша асимптотика времени работы $\mathcal{O}(n \cdot 3^k)$.