

Иерархия цитадели

Автор задачи: Александр Гордеев, разработчик: Даниил Орешников

Переформулируем задачу: в дереве, в котором все листья расположены на одном и том же уровне, требуется отсортировать эти самые листья по возрастанию, меняя порядок детей у каких-либо вершин.

Заметим следующий факт: если листья в поддереве какой-либо вершины не образуют множество подряд идущих чисел, то отсортировать их не получится. И, наоборот, если поддерево каждой вершины содержит листья с подряд идущими номерами, то есть способ отсортировать их. Действительно, если в каком-то поддереве найдутся i и k , что существует j между ними ($i < j < k$) не из этого поддерева, то эта тройка листьев не сможет быть упорядочена: либо j будет стоять после k , либо до i .

Обратное утверждение следует из алгоритма решения: сделаем **dfs** (на самом деле в рамках данной задачи достаточно просто пройти по внутренним вершинам дерева в порядке убывания их номеров), и для каждой вершины посчитаем, чему равны минимальный и максимальный номера листьев в ее поддереве, проверив, что между этими значениями лежит ровно столько же чисел, сколько листьев в поддереве.

Когда эта информация посчитана для всех u_i — детей вершины v , отсортируем их в порядке возрастания номеров листьев в поддеревьях, и проверим, что для любых двух соседних детей выполняется $\text{min_leaf}(u_i) = \text{max_leaf}(u_{i-1}) + 1$. Только если все такие равенства выполняются, листья поддерева вершины v и поддеревьев всех ее потомков образуют отрезки подряд идущих значений, а подходящий способ сортировки — это перестановка u_i в полученном нами отсортированном порядке.

Для каждой вершины мы сортируем всех ее детей, что дает асимптотику времени работы $\mathcal{O}(n \log n + m)$.