## Задачи

**Задача 1.** Дано дерево на $n$ вершинах. Как проверять, является ли вершина $u$ предком вершины $v$ с ответом на запрос за $O(1)$ и преподсчетом за $O(n)$?

**Задача 2.** Пусть в алгоритме поиска компонент сильной связности обход в глубину запускался не на траспонированном графе, а на исходном. Приведите пример графа, на котором данный алгоритм найдет компоненты сильной связности некорректно. А если при этом идти в обратном порядке топологической сортировки?

**Задача 3.** Есть массив $a_1, a_2, \ldots, a_n$. Про него известно $q$ фактов. $i$-й факт говорит о том, что сумма элементов на отрезке $[l_i, r_i]$ равняется $s_i$. Гарантируется, что данные не противоречивы (то есть существует хотя бы один массив, подходящий под эти условия). Придумайте алгоритм, находящий сумму элементов массива на отрезке $[1, n]$, если она определена однозначно, и сообщающий, что единственного ответа нет, в противном случае. Время работы алгоритма должно составлять $O(n + q)$.

**Задача 4.** Назовем топологическую сортировку $v_1, v_2, \ldots, v_n$ корректной, если для каждой вершины верно, что в нее не идет ребер из вершин, записанных в последовательности после нее. Будем сравнивать две топологические сортировки как последовательности чисел. Найдите лексикографически минимальную топологическую сортировку за $O(n \log n)$.

**Задача 5.** Вам дано корневое дерево на $n$ вершинах. Посчитайте размеры поддеревьев, не считая явно через динамику, за $\mathcal{O}(n)$.

**Задача 6.** Дан ориентированный ациклический граф на $n$ вершинах и $m$ ребрах. Также дана последовательность $v_1, v_2, \ldots, v_k$. Найдите количество путей от $v_1$ до $v_k$, проходящих через $v_2, v_3, \ldots, v_{k-1}$ по модулю $C$:

a) за $O(k(n + m))$;

b) за $O(n + m)$.

**Задача 7.** Дано взвешенное дерево на $n$ вершинах. Найдите суммарную длину всех путей в дереве за $O(n)$.

**Задача 8.** Дан ориентированный граф $G$ на $n$ вершинах. Назовем его *транзитивным замыканием* граф $T(G)$, в котором ребро между $u$ и $v$ есть тогда и только тогда, когда в графе $G$ есть путь от $u$ к $v$. Множество вершин $U$ назовем *кликой*, если для любых двух различных вершин $v$ и $u$ из $U$ есть ребро либо из $v$ в $u$, либо из $u$ в $v$, либо в обе стороны. Найдите размер наибольшей клики в графе $T(G)$ за $O(n + m)$.