МФТИ, ФПМИ

Алгоритмы и структуры данных, осень 2021

Семинар №9. Дерево отрезков (2)

1. Реализуйте персистентный стек, отвечая на все вопросы за $O(1)$. Нужно уметь обрабатывать обычные операции стека (`push`, `pop`, `top`), а также уметь откатываться к версии с данным номером t.

2. Поговорим о персистентной очереди.

а) Почему нельзя столь же просто, как в первой задаче, построить персистентную очередь?

б) На лекции разбиралось, как реализовать очередь с помощью двух стеков. Чем плоха такая реализация в случае откатов версий?

в) Предложите способ реализации персистентной очереди с полной с логарифмическим временем ответа на запрос.

3. К массиву длины n поступает q запросов одного из двух видов: а) изменить число в точке; б) по индексам l и r сообщить сумму $a_l + 2a_{l+1} + 3a_{l+2} + \ldots + (r-l+1)a_r$. Обработайте все запросы за $O(n + q \log n)$.

4. На плоскости заданы n прямоугольников со сторонами, параллельными осям координат. Определите площадь их объединения за $O(n \log n)$.

5. Найдите количество инверсий в массиве длины n за $O(n \log n)$, используя дерево отрезков.

6. В статическом массиве находите k-ю порядковую статистику на подотрезке за $O(\log n)$, где n — длина массива.

7. В статическом массиве находите количество различных элементов на подотрезке за $O(\log n)$, где n — длина массива. Более формально, по индексам l и r определите размер множества $\{a_l, a_{l+1}, \ldots, a_r\}$.

8*. Поезд едет от станции 1 до станции n, останавливаясь также на станциях $2, 3, \ldots, n-1$ (именно в таком порядке). Всего в поезде s сидений. На поезд продано m билетов, каждый билет характеризуется номером сиденья и парой $i < j$ станций, между которыми едет пассажир. Ваня задаёт кассиру q запросов: можно ли добраться с l-й станции до r-й ($l < r$) новому пассажиру? Если можно, то какое минимальное количество мест ему нужно будет сменить? Запросы независимы, то есть Ваня билеты не покупает. Асимптотика: $O((m + s + n + q) \log(m + s + n + q))$.

9. Задана квадратная таблица чисел $n \times n$. Поступают запросы двух видов: а) изменить число в точке; б) найти сумму в подпрямоугольнике. Отвечайте на каждый запрос за $O(\log^2 n)$.

1. Вспомните реализацию через односвязный список. Храните в отдельном векторе корни всех версий стека.

2.

а) В очереди нужно хранить указатели на начало и на конец, а если для конечной вершины создать копию, то придётся перенаправить ребро из предпоследней вершины, и так далее.

б) Время ответа на каждый запрос может быть линейным, если откатываться заставят в момент, когда происходит переливание стеков.

в) Можно воспользоваться персистентным массивом и для каждого элемента знать все моменты времени, в которые он присутствует в очереди.

3. В вершине дерева отрезков храните соответствующую сумму с коэффициентами. Чтобы склеить результаты из двух вершин, нужно сдвинуть все коэффициенты (придётся хранить ещё одно поле).

4. Сожмите координаты. Воспользуйтесь методом сканирующей прямой, то есть поддерживайте профиль пересекаемых прямоугольников при их прохождении, скажем, слева направо.

5. Выключите все элементы, а потом восстанавливайте их порядке возрастания.

6. Отсортируйте исходный массив и запомните индексы элементов в исходной нумерации: (a_i, i). Постройте fractional cascading на вторых координатах пар. Затем нужна идея двоичного спуска.

7. Для каждого i определите $prev(i) = \max\{j \mid j < i \wedge a_j = a_i\}$. Тогда количество различных чисел на отрезке $[l, r]$ — это в точности количество позиций, у которых $prev < l$.

8*. Для каждого места запомните список отрезков, в течение которых оно свободно. С помощью дерева отрезков научитесь отвечать на запрос "на какое максимальное расстояние можно отъехать от станции l без смены места?" Затем воспользуйтесь идеей двоичных подъёмов: для каждого l и каждого k определите, как далеко можно уехать от l, сменив 2^k сидений?

9. Создадим двумерное дерево отрезков. А именно, построим дерево отрезков по абсциссам (иксам). Каждая вершина этого (внешнего) дерева соответствует какому-то отрезку абсцисс, то есть полосе в таблице. В этой вершине заведите своё дерево отрезков по ординатам (игрекам), в котором храните суммы на подотрезках.