

Олимпиадное программирование

Занятие 25. ДД по неявному ключу. Merge-sort tree

Труфанов Павел Николаевич

Онлайн-школа  Фоксфорд

Foxford.ru 2019-2020

Три интересных задачи

- ▶ Вывести все элементы дерева в порядке возрастания
- ▶ Построить дерево, если у вас имеется отсортированный список ключей.
- ▶ Построить дерево, если у вас имеется отсортированный список ключей и приоритетов.

А что, если мы будем хранить в ключе индекс элемента. Тогда все будет отсортировано по индексу и мы сможем поддерживать обычный массив в дереве. Но при любой операции нам нужно будет сдвигать индексы. Поэтому давайте поддерживать индексы неявно через размер поддеревьев.

Теперь функция `split` будет отрезать от дерева первые k элементов. Код будет довольно простой.

Заметим, что наша старая функция merge никак не задействовала ключи дерева. Поэтому она не изменится.

Хочу узнать сумму на подотрезке. Для этого поддерживаю функцию суммы в поддереве. Теперь когда приходит запрос l, r , я режу дерево по r , потом первый кусок по l и смотрю значение функции в корне. Надо теперь не забыть сделать обратно merge.

Давайте сделаем запрос на подотрезке используя функцию, похожую на запрос в дереве отрезков. Таким образом не будем тратить много времени на split и merge.

Дерево merge-sort

Задача: хочу отвечать на запросы: сколько чисел на отрезке L, R лежат в числовом диапазоне A, B . Давайте построим дерево отрезков, где в каждой вершине будем хранить отсортированный массив всех ключей в ее поддереве. Таким образом, в одной вершине я могу ответить на запрос за логарифм бинарным поиском.

- ▶ Добавляем запросы изменения к предыдущей задаче.
- ▶ Количество различных чисел на отрезке.

До встречи!

FOXFORD.RU

Онлайн-школа Фоксфорд



Фоксфорд