Домашняя работа №3

Шибаев Александр Б05-222

Ноябрь 2022

1 Первая задача

Давайте заведем двумерные sparse table: table[m][k][i][j] = минимум к прямоугольнике, у которого левый верхний угол в точке (i, j), а длинаы сторон 2^m и 2^k. Тогда при запросе минимума на произвольном прямоугольнике мы берем 4 попарно пересекающихся прямоугольника со стронами, равными степеням двойки(аналогично одномерному случаю) и получаем ответ.

2 Вторая задача

Был дан массив $arr[n]$.
Построим массив $prev[i] = prev[i-1]+1$, если $arr[i]$ открывающаяся скобка, и $prev[i] = prev[i-1]-1$ в противном случае. $prev[0] = 1$, если $arr[0]$ - открывающая скобка, и $prev[0] = -1$ иначе. Построим на массиве $prev\ sparsetable$. И теперь, когда у нас спрашиваю, является ли подстрока с l по r ПСП, нам нужно проверить, что $\min prev[i]_{i=l}^{r} \geq 0$ и что $prev[l] == prev[r]$.

3 Четвертая задача

Давайте хранить в вершине ДО, которая отвечает за отрезок $[l, r]$ следующие параметры: сумму чисел $a[l], ...a[r]$, сумму квадратов этих же чисел и сумму кубов этих чисел, т.е.: $a[l]^2 + ... + a[r]^2$ и $a[l]^3 + ... + a[r]^3$. Тогда как реагировать на запрос изменения очевидно, на запрос сумму на отрезке тоже, на запрос попарного произведение:
Обозначим $a + b + c + ... + x = S_1$ Обозначим $a^2 + b^2 + c^2 + ... + x^2 = S_2$ Обозначим $a^3 + b^3 + c^3 + ... + x^3 = S_3$
Тогда папарное произведение числе $a, b, ..., x$ равно $(S_1^2 - S^2)/2$.
Теперь нужно выразить произведение троек:
Заметим, что $D = (S_1 * S_2 - S_3)$ - сумма произведений трех чисел, среди которых ровно два равных, т.е. вида a^2b. Тогда $(S_1)^3 = S_3 + 3 * D + 6 * X$, где X - искомая сумма. Поэтому $X = ((S_1)^3 - S_3 - 3 * (S_1 * S_2 - S_3))/6$.

4 Пятая задача

Давайте построим ДО для каждого бита. и будем поддерживать сумму на отрезке в каждом дереве. Если пришел запрос xor на отрезке, то можно посмотреть на все биты числа x, и применить инверсию на отрезке в i-ом ДО, если i-ый бит в числе x - единица. При этом сумма на отрезке изменится следующим образом: $segment_tree = length - segment_tree$, где

$length$ длина подотрезка, за которое отвечает вершина. А если нам приходит запрос and на отрезке, то мы должны прибавить 2^i к ответу, если сумма на нужном отрезке в i-ом дереве равна $length$ - длине подотрезка, за которое отвечает эта вершина. И так пройти по всем i от 0 до 9. Ассимптотика очевидно $O(q \log n)$.