

## Амортизированный анализ

1. Стек с `multirop`
2. Двоичный счетчик
3. Стек с `multirop` и `multiupush`.  
Остается ли справедливой амортизированная оценка стоимости стековых операций, равная  $O(1)$ , если включить в множество стековых операций операцию *multiupush*, помещающая в стек  $k$  элементов?
4. Счетчик с `decrement`  
Покажите, что если бы в примере с  $k$ -битовым счетчиком была включена операция *decrement*, то стоимость операций была бы равна  $\Theta(kn)$ .
5. Предположим, что над структурой данных выполняется  $n$  операций. Стоимость  $i$ -ой по порядку операции равна  $i$ , если  $i$  — это точная степень двойки, и 1 — иначе. Определить с помощью группового анализа, метода предоплаты и метода потенциалов амортизированную стоимость операции.
6. Рекурсивный обход дерева
7. Стек с копированием.  
Предположим, что над стеком выполняется последовательность операций. Размер стека при этом никогда не превышает  $k$ . После каждых  $k$  операций производится резервное копирование стека. Присвоив различным стековым операциям соответствующие стоимости, покажите, что стоимость  $n$  стековых операций, включая копирование стека, равна  $O(n)$ .
8. Счетчик с `reset`.  
Предположим, что нам нужно иметь возможность не только увеличивать показания счетчика, но и сбрасывать его. Считая, что время проверки или модификации одного бита составляют  $\Theta(1)$ , покажите, как осуществить реализацию счетчика в виде массива битов, чтобы выполнение произвольной последовательности из  $n$  операций *Increment* и *Reset* над изначально обнуленным счетчиком потребовало бы время  $O(n)$ .
9. Чему равна полная стоимость выполнения  $n$  стековых операций *push*, *pop*, *multirop*, если предположить, что в начале стек содержит  $k$  объектов, а в конце  $m$ .
10. Очередь с минимумом. Найти амортизированную стоимость операций.
11. Дек с минимумом на стеках. Найти амортизированную стоимость операций.

## Деревья отрезков

1. Найти минимум на отрезке от  $l$  до  $r$ , а так же число элементов, равных этому значению.
2. Найти значение суммы  $a_l - a_{l+1} + a_{l+2} - a_{l+3} + \dots \pm a_{r-1}$ .
3. Найти значение суммы  $a_l + 2a_{l+1} + 3a_{l+2} + \dots + (r-l)a_{r-1}$ .
4. Подсчёт количества нулей, поиск  $k$ -го нуля.
5. Посчитать количество инверсий в перестановке.
6. Есть массив  $a$  из  $n$  целых чисел и две операции:
  - (a) присвоить значение:  $a_i = x$
  - (b) найти минимальное  $i$ :
    - i. для которого  $a_i \geq k$
    - ii. для которого  $a_i \geq k$  на отрезке от  $l$  до  $r$
7. Покраска забора — запросы в порядке следования  $[l_i \dots r_i]$  покрасить в цвет  $c_i$  — найти сколько какого цвета в итоге.
8. Покраска забора 2 — красит несколько ребят, и некоторые из них только через одну, а запросы такие же — просто понять, что нам нужно два дерева отрезков
9. Дано  $n$  целых чисел  $a_1, a_2, \dots, a_n$  и  $q$  запросов. Запрос — числа  $l$  и  $k$ . Ответ на запрос — минимальное число  $r$ , такое что  $a[l \dots r]$  содержит  $k$  различных чисел. оффлайн обработка.



- (с) проверить, является ли правильной скобочной последовательностью подстрока с  $l$  до  $r$
- (d) найти наибольший префикс подстроки с  $l$  до  $r$ , который является правильной последовательностью
22. В решении этой задачи необходимо использовать Дерево отрезков без интервальной модификации. Опишите алгоритм решения следующей задачи. Перед вами полоска длины  $N$ , разбитая на  $N$  одинаковых квадратов, пронумерованных слева направо числами от 0 до  $N - 1$ . За одну операцию изменения (метод, предоставляемый интерфейсом вашей структуры данных) вы должны перекрасить все клетки с номерами от  $L$  до  $R$  включительно: белые должны стать черными, а черные - белыми. В запросах поиска необходимо определить цвет клетки в позиции  $X$ . Изначально задается раскраска полоски в черный и белый цвет. Интерфейс:
- (a) `Init(vector colors)`; Время работы метода должно быть линейным
  - (b) `ChangeColor(L, R)`; Время работы метода должно быть  $O(\log N)$
  - (c) `GetColor(X)`. Время работы метода должно быть  $O(\log N)$
23. Дано множество из  $n$  точек в трехмерном пространстве с вещественными координатами  $(x_i, y_i, z_i)$ . Точка  $(x_i, y_i, z_i)$  называется оптимальной по Парето, если **не существует** никакой другой точки  $(x_j, y_j, z_j)$  в множестве, такой, что  $x_i \leq x_j, y_i \leq y_j, z_i \leq z_j$ , причем хотя бы одно из неравенств строгое. Найти количество точек, оптимальных по Парето в данном множестве.
24. Дан массив из  $n$  нулей. Предложите структуру, которая умеет реализовывать следующие операции:
- (a) Установить все значения от  $l$  до  $r$  в единицу
  - (b) Установить все значения от  $l$  до  $r$  в ноль
  - (c) Посчитать количество максимальных по включению отрезков, состоящих из единиц, внутри  $(l, r)$ -подмассива рассматриваемого массива
25. На экране расположены прямоугольные окна, каким-то образом перекрывающиеся (со сторонами, параллельными осям координат). Вам необходимо найти точку, которая покрыта наибольшим числом из них. Окна заданы координатами левой нижней и правой верхней точки.
26. В граф добавляются и удаляются ребра. Требуется после каждого изменения печатать количество компонент связности.
27. Дано подвешенное дерево с корнем в первой вершине. Все ребра имеют веса (стоимости). Вам нужно ответить на  $M$  запросов вида “найти у двух вершин минимум среди стоимостей ребер пути между ними”.

## Meet in the middle

1. Предложите алгоритм, который по заданному числу  $N$  находит количество четверок целых положительных чисел  $(x, y, z, t)$  - решений уравнения:

$$x^2 + y^2 + z^2 + t^2 = N$$

2. У вас есть несколько камней известного веса  $w_1, w_2, \dots, w_n$ . Придумайте, как распределить камни в две кучи так, что разность весов этих двух куч будет минимальной.
3. Задано прямоугольное поле размера  $n \times m$ . В каждой клетке записано целое число; число, записанное в клетке  $(i, j)$  равно  $a_{ij}$ .  
Ваша задача — посчитать количество путей из клетки  $(1, 1)$  в клетку  $(n, m)$ , удовлетворяющих следующим условиям:
  - (a) Из клетки можно перемещаться только вниз или только вправо. Более формально, из клетки  $(i, j)$  можно переместиться в клетку  $(i, j + 1)$  или в клетку  $(i + 1, j)$ .
  - (b) *Xor* всех чисел на пути из клетки  $(1, 1)$  в клетку  $(n, m)$  должен быть равен  $k$ .
4. Даны  $a, b, m$ . Хотим найти решение уравнения:

$$a^x = b \pmod m$$

где  $a$  и  $m$  взаимно просты.

5. Дан неориентированный, невзвешенный граф  $G$ . Нужно найти количество подклик данного графа. Граф задан матрицей смежности.

## Разное

1. У нас есть число 1, за ход можно заменить его на любое из  $x + 1$ ,  $x + 5$ ,  $2x$ ,  $5x$ . За какое минимальное число ходов мы можем получить  $n$ ?
2. Даны две последовательности  $a$  и  $b$ , найти  $c$ , являющуюся подпоследовательностью и  $a$ , и  $b$  такую, что  $\text{len}(c) \rightarrow \max$ . Например, для последовательностей  $\langle 1, 3, 10, 2, 7 \rangle$  и  $\langle 3, 5, 1, 2, 7, 11, 12 \rangle$  возможными ответами являются  $\langle 3, 2, 7 \rangle$  и  $\langle 1, 2, 7 \rangle$ .
3. Дан склад грузов  $a_1, a_2, \dots, a_n$ . Есть бесконечное множество машин размера  $K$ . При погрузке должно выполняться требование, что  $\sum a_i \leq K$ . Загружают машины по одной, погруженная машина сразу уезжает. Погрузчик может брать только самый левый/правый груз из массива.  
Задача: минимизировать число машин, использованное для перевозки грузов.
4. Нам дано  $n$  предметов с натуральными весами  $a_0, a_1, \dots, a_{n-1}$ . Требуется выбрать подмножество предметов суммарного веса не больше  $S$ .
5. Дана последовательность  $a$  длины  $n$ . Найти ее самую длинную возрастающую подпоследовательность.
6. Найти количество способов закрасить таблицу  $n \times m$  с помощью 0 и 1 так, чтобы не было квадратов  $2 \times 2$  только из 0 или 1.

## Декомпозиция

1. У вас есть поезд, в котором  $k$  мест. Поезд идет через  $N + 1$  город (занумерованы от 0 до  $N$ ). Когда человек хочет купить билет, он называет два числа  $x$  и  $y$  — номера станций, откуда и куда он хочет ехать. При наличии хотя бы одного сидячего места на этом участке на момент покупки ему продается билет, иначе выдается сообщение «билетов нет» и билет не продается. Ваша задача — написать программу, обслуживающую такого рода запросы в порядке их прихода.
2. Надо придумать способ, как можно достаточно эффективно отвечать на следующие запросы:
  - (a) найти сумму на отрезке
  - (b) вставить новый элемент на позицию  $i$
  - (c) удалить  $i$ -ый элемент
3. Научиться отвечать на следующие запросы:
  - (a) Увеличить все элементы на отрезке на  $x$ .
  - (b) Найти количество элементов, которые  $\leq x$  на отрезке от  $l$  до  $r$ .
4. Дано  $k$  бочонков с медом  $kegs$ , стоящие в ряд.  
У каждого бочонка есть значение  $s_i$  — текущее количество меда в нем,  $c_i$  — объем бочонка.  
Есть операция прибавить на отрезке с  $l$  по  $r$  арифметическую прогрессию:

$$kegs_l + a, kegs_{l+1} + a + b, kegs_{l+2} + a + 2b, \dots, kegs_{r-1} + a + (r - 1 - l) \cdot b$$

Необходимо для каждого бочонка вывести первый запрос, когда он переполнится.

5. Дан граф. Два вида запросов:
  - (a) перекрасить вершину в какой-то цвет
  - (b) сказать сколько для данной вершины различных цветов среди ее соседей
6. Маленький Петя очень любит играть. Больше всего он любит играть в игру «Лунки». Это игра для одного игрока со следующими правилами:  
Есть  $N$  лунок, расположенных в ряд, пронумерованных слева направо числами от 1 до  $N$ . У каждой лунки изначально установлена своя сила выброса (у лунки с номером  $i$  она равна  $a_i$ ). Если вбросить шарик в лунку  $i$ , то он тут же вылетит из нее и попадет в лунку  $i + a_i$ , после чего он опять вылетит и т.д.. Если же лунки с таким номером нету, то он просто вылетит за край ряда.  
Вам необходимо научиться обрабатывать  $M$  ходов, каждый из которых может иметь следующий вид:

- Установить силу выброса лунки  $a$  равной  $b$ .
- Вбросить шарик в лунку  $a$  и посчитать количество прыжков шарика, прежде чем он вылетит за край ряда, а так же записать номер лунки, после выпрыгивания из которой шарик вылетел за край.

## Персистентные структуры

1. Дано  $n$  целых чисел  $a_1, a_2, \dots, a_n$  и  $q$  запросов. Запрос — числа  $l$  и  $k$ . Ответ на запрос — минимальное число  $r$ , такое что  $a[l \dots r]$  содержит  $k$  различных чисел.
2. Как можно реализовать персистентный двусвязный список? Определить тип персистентности. Посчитать амортизированную оценку для операции *insert*.
3. Как можно реализовать персистентный статический массив? Определить тип персистентности.
4. Как можно применить персистентное до для решения задачи про мед.
5. Как можно применить персистентное до для решения задачи про две перестановки.