Алгоритмы. Теор.листок 2

Артемий Клячин

17 ноября 2021 г.

Задача 1.

Итоговая формула: $2\sum\limits_{k=1}^{\lfloor\sqrt{n}\rfloor}\left\lfloor\frac{n}{k}\right\rfloor - \lfloor\sqrt{n}\rfloor^2$

Доказательство формулы:
Два числа m, q, будем называть парными делителями, если $mq \le n$.

У любого делителя i произвольного числа $k \le n$, всегда есть парный делитель $\frac{k}{i}$.
$\forall m \le n; i; j : m = i \cdot j$, выполняется, что либо только $i \le \sqrt{n}$ (случай 1), либо только $j \le \sqrt{n}$ (случай 2), либо $i \le \sqrt{n}$ и $j \le \sqrt{n}$ (случай 3).

$\sum\limits_{k=1}^{\lfloor\sqrt{n}\rfloor}\left\lfloor\frac{n}{k}\right\rfloor$ - это количество делителей, не больших $\sqrt{n}$, у всех чисел от 1 до n (k - делитель, $\left\lfloor\frac{n}{k}\right\rfloor$ - количество чисел, которые он делит). Это количество умножается на два, чтобы, при подсчёте количества делителей у всех чисел от 1 до n, учесть парные делители к уже посчитам делителям.

В выражении $2\sum\limits_{k=1}^{\lfloor\sqrt{n}\rfloor}\left\lfloor\frac{n}{k}\right\rfloor$ любые парные делители i, j учитывается ровно один раз в случаях 1 и 2, но дважды в случае 3 (так как оба делителя не больше $\sqrt{n}$ и уже были учтены в формуле $\sum\limits_{k=1}^{\lfloor\sqrt{n}\rfloor}\left\lfloor\frac{n}{k}\right\rfloor$). Поэтому нам необходимо вычесть из формулы количество пар парных делителей i, j, в которых парный делитель не нужно было дополнительно учитывать (это все пары отвечающие случаю 3).

Так как для любых $i, j : i \le \sqrt{n}, j \le \sqrt{n}$, число $ij \le n$, то любые такие i, j являются парными делителями. Таких пар i, j ровно $\lfloor\sqrt{n}\rfloor \cdot \lfloor\sqrt{n}\rfloor$. (пары (i, j) и (j, i) считаем разными, так как в одном случае мы лишний раз посчитали j, в другом i) Следовательно, количество всех делителей у всех чисел от 1 до n:

$$2\sum_{k=1}^{\lfloor\sqrt{n}\rfloor}\left\lfloor\frac{n}{k}\right\rfloor - \lfloor\sqrt{n}\rfloor^2$$

Доказано.
Формулу можно посчитать за $O(n)$ - это итоговая асимптотика.

Задача 2.

Решим задачу с помощью решета Эратосфена. Нахождение всех простых чисел от 1 до n (с помощью решета Эратосфена) займёт: $O(n \log\log n)$

Просеивание значений от n^2 до $n^2 + n$ (это n подряд идущих значений) этими набором простых чисел займёт $O(n \log\log n)$.

Все числа от n^2 до $n^2 + n$, которые по итогу не делятся ни на одно простое число от 1 до n, и будут простыми.

Итоговая асимптотика: $O(n \log\log n)$.

Задача 3.

Решим задачу с помощью решета Эратосфена. Нахождение всех простых чисел от 1 до $\sqrt{n}$ (с помощью решета Эратосфена) займёт: $O(\sqrt{n} \log\log \sqrt{n})$ по времени и $O(\sqrt{n})$ по памяти.

Далее разобьём числа на блоки вида $[\lfloor\sqrt{n}\rfloor \cdot (k-1), \lfloor\sqrt{n}\rfloor \cdot k - 1]$, где $k \in [1, \lfloor\sqrt{n}\rfloor + 3]$. Данные блоки совместно содержат все значения от 1 до n. Длина каждого блока $\lfloor\sqrt{n}\rfloor$.

Выделим массив длины $\lfloor\sqrt{n}\rfloor$. Для нахождения всех простых чисел (с помощью просеивания из алгоритма Эратосфена) в каждом блоке, понадобится лишь массив длины $\lfloor\sqrt{n}\rfloor$. Следовательно, если решать задачу сначала в первом блоке, потом во втором, потом в третьем и т.д. по порядку, то нам будет достаточно этого массива, чтобы найти простые числа во всех блоках.

В задаче мы использовали $O(\sqrt{n})$ памяти.

Нахождение простых чисел от 1 до $\sqrt{n}$ (с помощью решета Эратосфена) займёт: $O(\sqrt{n} \log\log \sqrt{n})$. Нахождение простых чисел в каждом блоке займёт $O(\sqrt{n} \log\log \sqrt{n})$ (следует из предыдущей задачи). Очистка массива после работы с блоком займёт: $O(\sqrt{n})$. Всего блоков $O(\sqrt{n})$. Следовательно, итоговая асимптотика:

$O(n \log\log n)$

Задача 4.

Формула: $\sum\limits_{d \geq 1} C_n^{kd} \equiv \frac{1}{k}(\sum\limits_{j=0}^{k-1}(1+\omega^j)^n)$, где ω корень из 1 кратности k.

Доказательство формулы:

$\frac{1}{k}(\sum\limits_{j=0}^{k-1}(1+\omega^j)^n) \equiv \frac{1}{k}(\sum\limits_{j=0}^{k-1}\sum\limits_{i=0}^{n} C_n^i \omega^{i \cdot j}) \equiv \frac{1}{k}(\sum\limits_{i=0}^{n} C_n^i \sum\limits_{j=0}^{k-1}\omega^{i \cdot j})$

Далее разберём случаи:

1) i не кратно k, тогда $w^i \not\equiv 1$. Тогда верна формула $\sum\limits_{j=0}^{k-1}\omega^{i \cdot j} \equiv \frac{(\omega^i)^k - 1}{\omega^i - 1} \equiv 0$, т.к. $\omega^k \equiv 1$

2) i кратно k, тогда $w^i \equiv 1$. Тогда верна формула $\sum\limits_{j=0}^{k-1}\omega^{i \cdot j} \equiv \sum\limits_{j=0}^{k-1} 1 \equiv k$.

Следовательно,

$\frac{1}{k}(\sum\limits_{i=0}^{n} C_n^i \sum\limits_{j=0}^{k-1} \omega^{i \cdot j}) \equiv \sum\limits_{d \geq 1} C_n^{kd}$

Формула доказана.

Значение $\frac{1}{k}(\sum\limits_{j=0}^{k-1}(1+\omega^j)^n)$ можно найти за $O(k \log n \log k)$. (k - количество итераций цикла, $\log n$ - быстрое возведение в степень n, $\log k$ - быстрое возведение в степень j)

Задача 5.

Алгоритм:

Для того чтобы решить задачу, нужно проделать двумерное дискретное преобразование фурье.

Сделаем дискретное преобразование Фурье для каждой строки матрицы коэффициентов. В полученной матрице коэффициентов сделаем ДПФ для каждого столбца. Перемножим матрицы для разных многочленов. Сделаем обратное ДПФ для каждого столбца. Затем сделает обратное ДПФ для каждой строки.

Полученная матрица будет давать коэффициенты для произведения многочленов. Чтобы не потерять коэффициенты нужно матрицы исходных многочленов увеличить до размеров, чтобы степень по x и по y результирующего полинома не превысила размеры матрицы. Если исходные многочлены описывались матрицей размера $(n+1) \times (m+1)$, то вычисления следует проводить с матрицей не меньше чем $(2n+1) \times (2m+1)$. Для работы быстрого преобразования Фурье можно взять ближайшую степень двойки.

Число операций на каждом шаге: прямое преобразование для строк $O(nm \cdot \log(m))$, для столбцов $O(mn \cdot \log(n))$, поэлементное перемножение матриц $O(nm)$, обратные преобразование по числу операций равны прямым. Всего $O(mn(\log m + \log n)) = O(mn \cdot \log(mn))$.

Задача 6аб.

Научимся считать для n вектор D_n из $9n+1$ элементов, где $D_n[i]$ равно числу билетов с числом цифр n и суммой цифр равной i. Тогда число счастливых билетов с длиной номера $2n$ будет равняться $\sum_{i=0}^{9n}(D_n[i]^2)$.

$D_0 = [1, 0, ..., 0]$. Научимся делать две операции: по D_m находить D_{m+1} и D_{2m}. Тогда за $\log(n)$ шагов мы из D_0 найдём D_n.

Первая операция: $D_{m+1} = \sum_k (D_m >> d_k)$:

Для каждой цифры d_k мы сдвигаем значения D_m на d_k значений в сторону конца массива. Суммируем полученные массивы. Понятно, что реального сдвига делать не нужно и число операций на шаг $O(nk)$.

Вторая операция:

$D_{2m} = D_m * D_m$ (свёртка).

$D_{2m}[s] = \sum_{j=0}^{9n}(D_m[j] * D_m[s-j])$.

Для уменьшения количества операций будем вычислять свёртку через преобразование Фурье. Для вычисления свёртки достаточно будет взять массивы длиной $9m+1$, так как все более старшие элементы $D_m[i]=0$ для $i>9m$. Вычислим для него преобразование Фурье, возведём каждый элемент вектора в квадрат и вычислим обратное преобразование Фурье.

Порядок применения операций: пусть двоичное представление числа $n = \overline{b_{k-1}b_{k-2}...b_0}$. Пусть $D_cur = D_0$. Будем двигаться от $k-1$ бита к нулевому: Начало цикла. Если текущий бит ненулевой, то выполняем операцию 1 для D_cur: m увеличивает на 1. Если у нас не последний бит, то выполняем операцию 2 для D_cur: m увеличиваем в 2 раза. Число операций 1 один равно числу единичных битов в представлении числа n. Число операций 2 равно числу значащих битов, считая с первого ненулевого.

Если аккуратно подсчитать число операций на все операции типа 2, то получим $O(n/2\log(n/2))+O(n/4\log(n/4))+... = O((n/2+n/4+...)\log(n)) = O(n\log(n)$ (потому что). На все операции 1 мы потратим $O(n/2)+O(n/4)+ ... = O(n)$. На вычисление ответа $O(n)$. Следовательно, всего $O(n\log(n))$ Итого нам понадобилось $O(n\cdot\log(n))$ операций для второй операции. Для всего алгоритма $O(n\cdot\log(n)^2)$.

Задача 8.

Будем считать, что $n=2^k$.

Рекурсивно посчитаем коэффициенты следующих многочленов:

$(x-a_1),(x-a_2),...,(x-a_n)$ - слой 1

$(x-a_1)(x-a_2),(x-a_3)(x-a_4)...,(x-a_{2^k-1})(x-a_{2^k})$ - слой 2

$(x-a_1)(x-a_2)(x-a_3)(x-a_4),...,(x-a_{2^k-3})...(x-a_{2^k})$ - слой 3

...

$(x-a_1)(x-a_2)...(x-a_{2^k})$ - слой $\log(n)$

Тут $2n$ многочленов. $O(\log n)$ слоёв.

Пусть $S_{i,j} = (x-a_i)(x-a_{i+1})...(x-a_j)$

Рассмотрим a_1. Найдём значение $P_n(a_1)$:

$P_n(x) = S_{1,\frac{n}{2}}(x)Q_{1,\frac{n}{2}}(x) + R_{1,\frac{n}{2}}(x)$. Где $Q_{1,\frac{n}{2}}, R_{1,\frac{n}{2}}$, - частное и остаток при делении $S_{1,n}(x)$ на $S_{1,\frac{n}{2}}(x)$ соответственно. Заметим, что $S_{1,n}(a_i) = 0\cdot Q_{1,\frac{n}{2}} + R_{1,\frac{n}{2}}(a_i) = R_{1,\frac{n}{2}}(a_i)$, для всех i от 1 до $\frac{n}{2}$. Степень многочлена $R_{1,\frac{n}{2}}(a_i)$ не более $\frac{n}{2}$.

Продолжим:

$R_{1,\frac{n}{2}}(x) = S_{1,\frac{n}{4}}(x)Q_{1,\frac{n}{4}}(x) + R_{1,\frac{n}{4}}(x)$

$R_{1,\frac{n}{4}}(x) = S_{1,\frac{n}{8}}(x)Q_{1,\frac{n}{8}}(x) + R_{1,\frac{n}{8}}(x)$

...

$R_{1,2}(x) = S_{1,1}(x)Q_{1,1}(x) + R_{1,1}(x)$

Степень $R_{i,j}(x)$ не более $i-j$, потому что степень $S_{i,j}(x)$ равна $i-j$.

По аналогии с предыдущим утверждением $P_n(a_1) = R_{1,\frac{n}{2}}(a_1) = R_{1,\frac{n}{4}}(a_1) = ... = R_{1,1}(a_1)$, а $R_{1,1}(a_1)$ считается элементарна, потому что степень $R_{1,1}(x)$ не более 1.

Следовательно, задача сводится к поиску $R_{i,i}$, для всех $i\in[1,n]$. Решение описано.

Посчитаем асимптотику. Изначально рекурсивно находим коэффициенты многочленов (находим многочлены в слое из многочленов в предыдущем слое). Произведение многочленов находим с помощью быстрого преобразования Фурье за $O(k \log k)$, где k - степень итогового многочлена. Суммарная степень многочленов в каждом слое: n. Всего $\log n$ слоёв. Следовательно, мы найдём многочлены за $O(n \log^2(n))$.

Алгоритм деления многочленов A на B (нахождение частного и остатка) работает за $O((len(A) + len(B)) \cdot \log(len(A) + len(B)))$.

Нам необходимо получить $R_{i,i}$, для всех $i \in [1, n]$. Для этого надо произвести $2n$ делений (P_n на $R_{1,\frac{n}{2}}$ и $R_{\frac{n}{2},n}$; $R_{1,\frac{n}{2}}$ на $R_{1,\frac{n}{4}}$ и $R_{\frac{n}{4},\frac{n}{2}}$ и т.д.). По аналогии с рассуждениями выше, можно сказать, что мы получим результат за $O(n \log^2(n))$.

Итоговая асимптотика: $O(n \log^2(n))$