Розбір задачі «Прикольна»

Нескладно помітити,що кількість чисел, найбільший спільний дільник степенів факторизації яких ділиться на g, дорівнює $\sqrt[g]{n}$. Тоді якщо скористатися формулою включень-виключень, то отримаємо такий результат:

$$\sum_{g=1}^{n} \mu(g) \cdot \sqrt[g]{n}$$

Оскільки степені простих у факторизації не перевищують 60, то можна було не в явному вигляді рахувати функцію Мебіуса, а рахувати її для простих чисел і добутків двох і трьох різних простих чисел(подібний добуток із 4 чисел перевищує 60).

Розбір задачі «Цікава»

Можна помітити, що якщо $g = gcd(a, b)$, то $r = gcd(a, b - g)$ буде ділитися на g. Це випливає із властивостей алгоритму Евкліда, який використовується для обрахування найбільшого спільного дільника. З цього випливає, що $r = k * g$, і якщо $k \neq 1$, то це означає, що r більше за g принаймні вдвічі. Отже, випадків, коли збільшується g, буде не більше ніж $log(min(a, b))$, і нашою задачею є грамотно відслідковувати, коли саме ці значення будуть збільшуватися.

Позначимо $x = \frac{a}{g}, y = \frac{b}{g}$. Тоді $gcd(x, y) = 1$, і момент, коли це значення перестане бути рівним 1 і буде моментом, коли найбільший спільний дільник збільшується. Оскільки на кожній ітерації значення b зменшується на g, то значення y зменшується на 1. Тобто, нам потрібно знайти найменше таке k, що $gcd(x, y - k) > 1$.

Нехай в пари $(x, y - k)$ є спільний дільник d, тоді $y \bmod d = k$. Отже, потрібно знайти таке $d > 1$, що d є дільникоом y і $y \bmod k$ є найменшим. Це можна зробити наївним перебором дільників за $O(\sqrt{n})$. Отже, сумарна складність алгоритму буде $O(\sqrt{n} \cdot log(n))$

Розбір задачі «Маленька»

Запишемо відповідь у вигляді суми: $\sum_{i=1}^{n} \sum_{j=1}^{n} [gcd(i, j) = 1]$. Тепер перегрупуємо доданки в цій сумі.

Перш за все, можна помітити, що пари (i, j) і (j, i) дають однаковий результат, то можна обраховувати тільки ті пари, для яких $i \leqslant j$, помноживши результат на 2. Але оскільки тепер випадки, де $i = j$, враховуються двічі, то їх потрібно від цієї суми відняти. Після перетворення отримаємо такий вираз:

$2 \cdot \sum_{i=1}^{n} \sum_{j=1}^{i} [gcd(i, j) = 1] - \sum_{i=1}^{n} [gcd(i, i) = 1]$.

Тепер помітимо наступне: друга сума дорівнює 1, тому що $gcd(i, i) = i$, і дорівнює 1 воно рівно в 1 випадку. Отже, наша сума має вигляд:

$2 \cdot \sum_{i=1}^{n} \sum_{j=1}^{i} [gcd(i, j) = 1] - 1$

Тепер помітимо, що в нашій сумі значення вкладеної суми відповідає означення функції Ейлера, тобто: $2(\sum_{i=1}^{n} \sum_{j=1}^{i} [gcd(i, j) = 1]) - 1 = 2(\sum_{i=1}^{n} \varphi(i)) - 1$

Обрахунок функції Ейлера можна робити, використовуючи факторизацію чисел, знайдену решетом Ератосфена. Отже, підсумкова складність буде $O(n \cdot log(log(n)))$

Розбір задачі «Неочікувана»

Легко помітити, що для будь якого натурального n справедливо те, що $gcd(n, n + 1) = 1$. Цей факт досить інтуїтивний і нескладно доводиться, тож використаємо йогоо і зрупуємо числа у пари $(l, l + 1), (l + 2, l + 3), \ldots (r - 1, r)$. Цього цілком достатньо, щоб розв'язати дану задачу. Цілком можливо, що існують інші способи, які застосовні до даної задачі, але цей спосіб є найпростішим і найочевиднішим.

Розбір задачі «Красива»

Нехай $cnt(g)$ — кількість наборів, таких що їхній НСД дорівнює g. Обраховуватимемо її за наступним принципом: порахуємо кількість наборів, для яких НСД ділиться на g, після чого віднімемо $cnt(2 \cdot g), cnt(3 \cdot g), cnt(4 \cdot g), \ldots$, словом значення $cnt(i)$ для всіх $i > g$, таких що i кратне g. Те, що залишиться, і буде кількістю таких наборів, для яких НСД дорівнює g.

Тоді якщо НСД ділиться на d, то всі числа з набору також діляться на d.Тож якщо кількості дільників числа d в послідовностях дорівнюють $c_{d,1}, c_{d,2}, \ldots, c_{d,n}$, то тоді кількість варіантів обрати не менш ніж 2 числа дорівнює $(c_{d,1}+1)(c_{d,2}+1)\ldots(c_{d,n}+1) - c_{d,1} - c_{d,2} - \ldots - c_{d,n} - 1$ (для кожної послідовності a_i є $c_{d,i}$ варіантів обрати число і варіант не використовувати її взагалі, їхній добуток буде загальною кількістю способів, від якої треба відняти кількість варіантів обрати множину розміру 1 і варіант, коли множина пуста).

Об'єднаємо ці ідеї, і отримаємо таку формулу:

$$cnt(g) = \prod_{i=1}^{n}(c_{g,i}+1) - 1 - \sum_{i=1}^{n} c_{g,i} - \sum_{k=2}^{\lfloor m/g \rfloor} cnt(k \cdot g)$$

. Значення $c_{g,i}$ можна обраховувати за допомогою гармонічного обходу по масиву кількостей елементів(банально рахуємо суму $counter_{i,g} + counter_{i,2\cdot g} + \ldots$, де $counter_i$ - це масив кількостей для послідовності з номером i).

Складність: $O(nalog(a))$.

Розбір задачі «Мила»

Нескладно помітити, що кожне натуральне $n > 1$ можна розкласти на суму двійок і трійок. Один із способів може бути таким: парні числа розкладаємо на суму двійок, а непарні — на суму двійок і однієї трійки. Оскільки кількість двійок в кожному розкладі максимально можлива, то такий спосіб буде оптимальним.

Розбір задачі «Радикальна»

Перепишемо цю суму іншим чином: для кожного числа вивести кількість чисел, які на нього діляться. Тоді це буде $\sum_{i=1}^{n} \lfloor \frac{n}{i} \rfloor$. Можна помітити, що для $i \leqslant \sqrt{n}$ ми можемо порахувати ці значення перебором, а якщо ця умова не виконується, то очевидно, що $\lfloor \frac{n}{i} \rfloor \leqslant \sqrt{n}$. Для цих чисел ми можемо перебрати значення $x = \lfloor \frac{n}{i} \rfloor$ і порахувати, скільки існує відповідних значень i. Для $x = 1$ ми точно знаємо, що нам підійде значення n і, можливо, якісь значення, менші за n. Найлегше знайти це значення бінпошуком. Для $x = 2$ ми знайдемо значення бінпошуком, відштовхуючись від останнього числа, для якого $x = 1$, так само, відштовхуючись від попереднього, зробимо для $x = 3$ і тд.

Складність: $O(\sqrt{n} \cdot log(n))$.

Розбір задачі «Приємна»

Напишемо решето Ератосфена, в якому для кожного числа i дізнаємося його мінімальний дільник d_i. Тепер ми можемо знаходити факторизацію числа,просто кожного разу поділивши його на мінімальний дільник. Таким чином ми можемо отримати факторизацію числа за час $O(length)$, де $length$ — довжина факторизації. Знаючи її, легко можна обрахувати суму, яку потрібно вивести в задачі.

Складність: $O(n \cdot log(n))$.

Bonus: як розв'язати задачу за складність $O(n \cdot log(log(n)))$?

Розбір задачі «Незвичайна»

Подамо факторизацію у вигляді $n = p_1^{a_1} \cdot p_2^{a_2} \cdot \ldots \cdot p_k^{a_k}$. Тепер розглянемо якесь $p_i^{a_i}$. Ми знаємо, що у дільник це просте число або не входить, або входить у одному з виглядів: $p_i, p_i^2, \ldots, p_i^{a_i}$. Для кожного з них існує $k = (a_1+1) \cdot \ldots \cdot (a_{i-1}+1) \cdot (a_{i+1}+1) \cdot \ldots \cdot (a_k+1)$ дільників, у який він входить. Тоді нам потрібно кожне з чисел врахувати у відповіді саме k разів. Тобто, для $p_i^{a_i}$ відповідь домножиться на $p_i^k \cdot (p_i^2)^k \cdot \ldots \cdot (p_i^{a_i})^k = p_i^{k \cdot a_i \cdot (a_i+1)/2}$. Перемножишви всі ці величини за модулем, отримаємо відповідь на нашу задачу.

Розбір задачі «Чарівна»

Попередньо порахуємо значення $S(i)$ за допомогою факторизації решетом Ератосфена і заведемо масив підрахунку cnt_i = кількість значеннь масиву a, що дорівнюють i. Тепер використаємо гармонічний обхід по парах i, j, де $j \bmod i = 0$, і якщо $S(j) \bmod S(i) = 0$, додамо до відповіді значення $cnt_i \cdot cnt_j$ (Для випадку $i = j$ потрібно додати $cnt_i \cdot (cnt_i - 1)$).

Складність: $O(n \cdot log(n))$.

Розбір задачі «Загадкова»

Доведемо, що функція $f(n)$ є функцією Ейлера. З умови знаємо, що це кількість таких x, що $gcd(x, n - x) = 1$, а з алгоритма Евкліда випливає, що $gcd(x, n - x) = gcd(x, n)$. Тобто, з цього випливає, що це означення є еквівалентним означенню функції Ейлера.

Тоді з властивостей функції Ейлера $g(n) = \sum_{d|n} \varphi(d) = n$. Отже, функція $F(n, k)$ зводиться до того, що ми змінюємо n на $\varphi(n)$, якщо $k \bmod 2 = 1$. Ввнаслідок таких маніпуляцій із n його значення швидко збігається до 1(швидкість збіжності можна оцінити як $O(log(n))$), після чого аргументи функції залишатимуться без змін. Цей факт можна використати, щоб обрахувати значення результуючої функції дуже швидко. Значення функції Ейлера можна обраховувати наївним алгоритмом за $O(\sqrt{n})$.

Складність: $O(\sqrt{n} \cdot log(n))$.