

Обозначения

$$\sum_{i=1}^n a_i = a_1 + a_2 + \dots + a_n$$

$f(n) = \mathcal{O}(g(n))$ — существуют такие числа a и b , что для всех n : $f(n) \leq a + bg(n)$.

$$\prod_{i=1}^n a_i = a_1 \cdot a_2 \cdot \dots \cdot a_n$$

$a \equiv b \pmod{m}$ — a и b имеют одинаковые остатки при делении на m .

Задачи

Определение. Функция Эйлера $\varphi(n)$ — количество натуральных чисел не больших n , которые взаимно просты с n . Если число n имеет разложение на простые числа в виде $n = p_1^{a_1} \cdot p_2^{a_2} \cdot \dots \cdot p_k^{a_k}$, то

$$\varphi(n) = n \cdot \left(1 - \frac{1}{p_1}\right) \cdot \left(1 - \frac{1}{p_2}\right) \cdot \dots \cdot \left(1 - \frac{1}{p_k}\right)$$

или иначе говоря

$$(p_1 - 1) \cdot p_1^{a_1 - 1} \cdot (p_2 - 1) \cdot p_2^{a_2 - 1} \cdot \dots \cdot (p_k - 1) \cdot p_k^{a_k - 1}$$

Определение. (Теорема Эйлера) Для любого числа k взаимно простого с n , $k^{\varphi(n)} \equiv 1 \pmod{n}$.

Частный случай этой теоремы при простом n называется *малой теоремой Ферма*. Этот случай примечателен тем, что $\varphi(p) = p - 1$ для простого p , поэтому теорема принимает совсем простой вид.

Задача 1. Найти количество натуральных чисел, которые меньше n и взаимно просты с числом m , за $\mathcal{O}(\sqrt{m})$.

Задача 2. Найти сумму НОД по всем подотрезкам массива натуральных чисел, не больших C , за $\mathcal{O}(n \log C)$.

Задача 3. Найти сумму НОД по всем непустым подмножествам массива из n натуральных чисел, не больших C , за $\mathcal{O}(n + C \log C)$. Ответ найдите по модулю $10^9 + 7$.

Задача 4. Найдите сумму всех простых чисел от 1 до n за $\mathcal{O}(n \log \log n)$ времени и $\mathcal{O}(\sqrt{n})$ памяти.

Задача 5. Дан массив a длины $n \leq 10^6$. Назовем k -разбиением такое разбиение массива a на подотрезки, что длины всех подотрезков (кроме, возможно, последнего) равны k . Силой подотрезка назовем куб суммы его элементов. Необходимо для каждого k от 1 до n посчитать сумму сил всех подотрезков в k -разбиении массива a .

Задача 6. Докажите, что в процессе работы расширенного алгоритма Евклида все x_i не превосходят b_i по модулю, а все y_i не превосходят a_i по модулю. Из чего следует, что если входные данные алгоритма Евклида помещаются в какой-то тип данных, то и во время вычислений не произойдет никаких переполнений.

Задача 7. Перемножьте два числа a и b по модулю $m \leq 10^{18}$ на C++ без использования `int128` за

- $\mathcal{O}(\log m)$
- $\mathcal{O}(1)$

Задача 8. За $\mathcal{O}(n)$ для каждого числа от 1 до n найдите:

- Количество его делителей.
- Сумму его делителей.
- Функцию Эйлера от него.

Задача 9. Дан массив a длины n . Необходимо для каждого k от 1 до n посчитать сумму элементов массива a на позициях, делящихся на k ($a_0 + a_k + a_{2k} + \dots$) за а) $\mathcal{O}(n \log n)$ б) $\mathcal{O}(n \log \log n)$.

Задача 10. Назовем суммирующей функцией Эйлера $\varphi_{\Sigma}(n)$ сумму всех натуральных чисел, не больших n , которые взаимно просты с n . Придумайте формулу для $\varphi_{\Sigma}(n)$.

Задача 11. Назовём натуральное число кубастым, если его можно представить в виде $a^3 \cdot b$ для каких-то натуральных $a > 1, b \geq 1$. Найти количество кубастых чисел, не больших $n \leq 10^{18}$.

Задача 12. а) Даны числа a_1, a_2, \dots, a_n , а также простой модуль p . Необходимо найти обратные по модулю p ко всем a_i за $\mathcal{O}(n + \log p)$.

- Найдите обратные ко всем числам от 1 до n по простому модулю p ($n < p$) за $\mathcal{O}(n)$.

Задача 13. Даны два числа a и b . Проверьте, влезает ли их произведение в тип `long long`.

Задача 14. Пусть $n \geq p$. Обозначим за $n!_p$ число, полученное из $n!$ делением его на p пока делится. Посчитайте $n!_p$ по модулю p за а) $\mathcal{O}(p \log_p n)$ б) $\mathcal{O}(p + \log_p n)$.

Задача 15. Дан массив a длины $n \leq 10^6$ ($1 \leq a_i \leq 10^7$). Необходимо найти пару его элементов с наименьшим значением НОКа.

Задача 16. (*) Научитесь выполнять n запросов поиска дискретного логарифма (найти такое x_i , что $a_i^{x_i} \equiv b_i \pmod{p}$) по простому модулю p за $\mathcal{O}(\sqrt{p \cdot n} + n \log p)$.

Задача 17. (**) Найдите количество простых чисел от 1 до n за $\mathcal{O}(n^{2/3})$.

Задача 18. (**) Докажите, что представленный код поиска обратного по простому модулю работает, и найдите его асимптотику:

```
int inv(int a) {
    if(a == 1) {
        return 1;
    }
    return MOD - (long long) MOD / a * inv(MOD % a) % MOD;
}
```

Задача 19. (***) Докажите, что количество делителей $d(n)$ числа n — это субполиномиальная величина. То есть для любого $\varepsilon > 0$ верно, что $\frac{d(n)}{n^\varepsilon} \rightarrow 0$ при $n \rightarrow +\infty$ (иными словами, $d(n) = o(n^\varepsilon)$).

Задача 20. (*****) Правда ли, что для любого натурального n существует натуральное $m \neq n$, такое что $\varphi(n) = \varphi(m)$?