

Solution

Wearry

Stay determined!

Refract

注意到直接按照题目要求按照 Y 坐标顺序转移复杂度难以优化. 考虑倒着转移, 那么每次 X 坐标的极值一定是单调变化的, 于是按照 X 坐标排序转移, X 的极大值就一定是单调的, 而极小值每次只需要考虑新加入的点影响到的部分, 可以一边转移一边更新.

Sequence

本题解法比较多, 这里讲一下出题人的做法:

对于第一问可以根据题意暴力递归计算答案, 复杂度 $O(Q \log N)$.

对于第二问考虑从第一问中归纳出关于 x_N 的性质: 将 N 二进制分解, 末尾的 0 对答案产生的影响可以快速处理, 末尾的若干个连续的 1 可以合并到它们左边的那个 0 上并得到一个新的 1, 合并过程中会产生 -1 的贡献当且仅当 1 的个数大于一且是奇数. 可以数位 dp.

Calc

考虑转化条件:

令 $d = \gcd(a, b)$, $a = a'd$, $b = b'd$.

$$a'd + b'd \mid a'b'd^2 \Rightarrow a' + b' \mid a'b'd \Rightarrow a' + b' \mid d$$

令 $d = t(a' + b')$, 则:

$$b = b'd = b'(a' + b')t$$

考虑枚举 $p = b'$, $q = (a' + b')$, 计算 t 的数量:

$$Ans = \sum_{p=2}^{\sqrt{N}} \sum_{q=p+1}^{2p-1} [\gcd(p, q) = 1] \left\lfloor \frac{N}{pq} \right\rfloor$$

Algorithm 1

定义 L_i 为 i 的质因子个数. 首先枚举 p , 考虑有哪些 q 能对答案产生贡献, 每次用 $O(L_p)$ 的时间判断 p, q 是否互质.

复杂度为 $O(\sqrt{N} \sum_{i=2}^{\sqrt{N}} L_i) = O(N \log \log \sqrt{N})$

Algorithm 2

枚举完 p 之后, $T = \lfloor \frac{N}{p} \rfloor$ 的值就确定了.

只需要考虑 $\lfloor \frac{T}{q} \rfloor$ 的值即可, 考虑对这样的 q 进行分块, 每一块内部利用 2^{L_p} 的容斥计算与 p 互质的 q 的个数.

复杂度近似为 $O(\sqrt{N} \sum_{i=2}^{\sqrt{N}} 2^{L_i})$

Complexity Analysis

分析实际运行的情况, 发现理论复杂度更高的算法二表现优于算法一, 原因是 $\lfloor \frac{T}{q} \rfloor$ 的取值很少. 考虑综合两种算法的长处, 在 q 比较小的时候, $\lfloor \frac{T}{q} \rfloor$ 的取值比较多, 我们使用算法一, q 比较大的时候使用算法二.

假设这个分界点为 k , 复杂度:

$$f(k) = (k - p)L + (\lfloor \frac{T}{k} \rfloor + \lfloor \frac{T}{2p} \rfloor)2^L$$

只考虑其中与 k 相关的部分:

$$f(k) = kL + \frac{T2^L}{k}$$

根据基本不等式, 使得 f 最优的 $k = \sqrt{\frac{T2^L}{L}}$, 此时 $f(k) = 2\sqrt{T2^L L}$. 算一下复杂度, 发现在 $N = 2^{31} - 1$ 时, 复杂度大约为 8×10^7 .