Solution

ExfJoe

福建省长乐第一中学

March 11, 2017

Outline

约数

约数

- 原题等价于求 $xyz \le n$ 的有序三元组个数

约数

- 原题等价于求 $xyz \le n$ 的有序三元组个数
- 假设 $x \le y \le z$，则 $x \le n^{\frac{1}{3}}$，$y \le \sqrt{\frac{n}{x}}$

约数

- 原题等价于求 $xyz \le n$ 的有序三元组个数
- 假设 $x \le y \le z$，则 $x \le n^{\frac{1}{3}}$，$y \le \sqrt{\frac{n}{x}}$
- 枚举 x, y，我们可以计算出合法的 z 的数量

约数

- 原题等价于求 $xyz \leq n$ 的有序三元组个数
- 假设 $x \leq y \leq z$，则 $x \leq n^{\frac{1}{3}}$，$y \leq \sqrt{\frac{n}{x}}$
- 枚举 x, y，我们可以计算出合法的 z 的数量
- 假设合法数为 ans，则 x, y, z 可以任意排列，因此 $ans * 6$

约数

- 原题等价于求 $xyz \le n$ 的有序三元组个数
- 假设 $x \le y \le z$，则 $x \le n^{\frac{1}{3}}$，$y \le \sqrt{\frac{n}{x}}$
- 枚举 x, y，我们可以计算出合法的 z 的数量
- 假设合法数为 ans，则 x, y, z 可以任意排列，因此 $ans * 6$
- 但是相等情况会重复计数，我们再计算有两个数相等、三个数全相等的情况数

约数

- 原题等价于求 $xyz \le n$ 的有序三元组个数
- 假设 $x \le y \le z$，则 $x \le n^{\frac{1}{3}}$，$y \le \sqrt{\frac{n}{x}}$
- 枚举 x, y，我们可以计算出合法的 z 的数量
- 假设合法数为 ans，则 x, y, z 可以任意排列，因此 $ans * 6$
- 但是相等情况会重复计数，我们再计算有两个数相等、三个数全相等的情况数
- 两个数相等：$\sum \frac{n}{i^2}$，三个数全相等：$n^{\frac{1}{3}}$

约数

- 原题等价于求 $xyz \le n$ 的有序三元组个数
- 假设 $x \le y \le z$，则 $x \le n^{\frac{1}{3}}$，$y \le \sqrt{\frac{n}{x}}$
- 枚举 x, y，我们可以计算出合法的 z 的数量
- 假设合法数为 ans，则 x, y, z 可以任意排列，因此 $ans * 6$
- 但是相等情况会重复计数，我们再计算有两个数相等、三个数全相等的情况数
- 两个数相等：$\sum \frac{n}{i^2}$，三个数全相等：$n^{\frac{1}{3}}$
- 时间复杂度：$O(n^{\frac{2}{3}})$

Outline

简单的操作

简单的操作

- 只有类型 5：莫队

简单的操作

- 只有类型 5：莫队
- 只有类型 2 类型 5：即每个数上一次出现的位置为 P_i，则一次询问可以看做一次二维数点，二维数据结构维护修改与询问即可

简单的操作

- 只有类型 5：莫队
- 只有类型 2 类型 5：即每个数上一次出现的位置为 P_i，则一次询问可以看做一次二维数点，二维数据结构维护修改与询问即可
- 类型 1：考虑容斥，记数集为 $|S|$，所有数的一次方、二次方、三次方之和分别是 S_1, S_2, S_3，则

$$sum = \frac{S_1^3 - 3S_1S_2 + 2S_3}{6}$$

其它的处理类似类型 5

简单的操作

- 只有类型 5：莫队
- 只有类型 2 类型 5：即每个数上一次出现的位置为 P_i，则一次询问可以看做一次二维数点，二维数据结构维护修改与询问即可
- 类型 1：考虑容斥，记数集为 $|S|$，所有数的一次方、二次方、三次方之和分别是 S_1, S_2, S_3，则

$$sum = \frac{S_1^3 - 3S_1S_2 + 2S_3}{6}$$

  其它的处理类似类型 5
- 操作 3、操作 4：预先用一棵平衡树预处理出所有操作的真实位置

简单的操作

- 只有类型 5：莫队
- 只有类型 2 类型 5：即每个数上一次出现的位置为 P_i，则一次询问可以看做一次二维数点，二维数据结构维护修改与询问即可
- 类型 1：考虑容斥，记数集为 $|S|$，所有数的一次方、二次方、三次方之和分别是 S_1, S_2, S_3，则

$$sum = \frac{S_1^3 - 3S_1S_2 + 2S_3}{6}$$

  其它的处理类似类型 5
- 操作 3、操作 4：预先用一棵平衡树预处理出所有操作的真实位置
- 合并以后，先用平衡树预处理出插入删除，剩下的修改询问操作使用二维数据结构或是 CDQ 分治套个数据结构即可

Outline

最小生成树

最小生成树

- 考虑画个图，然后按照 Kruskal 的过程转化一下边

最小生成树

- 考虑画个图，然后按照 Kruskal 的过程转化一下边
- $(A, A+i)$ 连一条 $(C+1+2i)$ 的边

最小生成树

- 考虑画个图，然后按照 Kruskal 的过程转化一下边
- $(A, A+i)$ 连一条 $(C+1+2i)$ 的边
- $(B, B+i)$ 连一条 $(C+2+2i)$ 的边

最小生成树

- 考虑画个图，然后按照 Kruskal 的过程转化一下边
- $(A, A+i)$ 连一条 $(C+1+2i)$ 的边
- $(B, B+i)$ 连一条 $(C+2+2i)$ 的边
- (A, B) 连一条 C 的边

最小生成树

- 考虑画个图，然后按照 Kruskal 的过程转化一下边
- $(A, A+i)$ 连一条 $(C+1+2i)$ 的边
- $(B, B+i)$ 连一条 $(C+2+2i)$ 的边
- (A, B) 连一条 C 的边
- 前两种连了整张图一圈，考虑以第一种边为例

最小生成树

- 考虑画个图，然后按照 Kruskal 的过程转化一下边
- $(A, A+i)$ 连一条 $(C+1+2i)$ 的边
- $(B, B+i)$ 连一条 $(C+2+2i)$ 的边
- (A, B) 连一条 C 的边
- 前两种连了整张图一圈，考虑以第一种边为例
- d_i 表示 $(i, i+1)$ 间的最小权值

最小生成树

- 考虑画个图，然后按照 Kruskal 的过程转化一下边
- $(A, A+i)$ 连一条 $(C+1+2i)$ 的边
- $(B, B+i)$ 连一条 $(C+2+2i)$ 的边
- (A, B) 连一条 C 的边
- 前两种连了整张图一圈，考虑以第一种边为例
- d_i 表示 $(i, i+1)$ 间的最小权值
- $d_i = \min(d_i, C)$，随后循环一遍 $d_i = \min(d_i, d_{i-1}+2)$

最小生成树

- 考虑画个图，然后按照 Kruskal 的过程转化一下边
- $(A, A+i)$ 连一条 $(C+1+2i)$ 的边
- $(B, B+i)$ 连一条 $(C+2+2i)$ 的边
- (A, B) 连一条 C 的边
- 前两种连了整张图一圈，考虑以第一种边为例
- d_i 表示 $(i, i+1)$ 间的最小权值
- $d_i = \min(d_i, C)$，随后循环一遍 $d_i = \min(d_i, d_{i-1}+2)$
- 总边数 $n+Q$，Kruskal 即可