题目 (/t/problems) **SPOJ (/t/problems-spoj)**

SPOJ DIVCNT3

zimpha (/u/zimpha) 11 Jan

题目链接：Counting Divisors (cube) (http://www.spoj.com/problems/DIVCNT3/)

题目大意： T 组测试数据，每组测试数据给出一个整数 n，求 $\sum\limits_{i=1}^{n}\sigma_0(i^3)$，$n\le 10^{11}$。

Margarita (/u/Margarita) likes this. Reply

zimpha (/u/zimpha) 11 Jan

自问自答。

网上流传着的做法都是洲阁筛做法，从ranking上看跑的都不是非常快，但是Min_25只跑了7.1s，想必正解是其它做法，我昨晚努力思考了一下，想出了一个$O(n^{2/3})$的别解，~~虽然不清楚是不是正解~~大抵是正解，但是应该常数上比洲阁筛快上不少。

先求$\sigma_0(n^3)$的值，令$n=\prod p_i^{e_i}$，考虑哪些因子只有n^3有，而n没有的。不妨假设$d=\prod p_i^{d_i}$，显然那些$d_i>e_i$的是n^3独有的。那么对于每个n的约数d，贡献就是$3^{\omega(d)}$（每个幂次可以加e_i，加$2e_i$或者保持不变），也就是说：

$$\sigma_0(n^3)=\sum_{d|n}3^{\omega(d)}$$

然后，发挥了oeis的作用，得知$3^{\omega(n)}=\sum\limits_{d|n}\sigma_0(d)\mu^2(d)$。于是经过一系列化简，我们就得到了下面的式子：

$$\sum_{i=1}^{n}\sigma_0(i^3)=\sum_{i=1}^{n}\sigma_0(i)\mu^2(i)\sum_{j=1}^{\lfloor\frac{n}{i}\rfloor}\sigma_0(j)$$

令$f(n)=\sum\limits_{i=1}^{n}\sigma_0(i)\mu^2(i)$，$g(n)=\sum\limits_{i=1}^{n}\sigma_0(i)$，只要我们能快速求出每个$f(\lfloor\frac{n}{i}\rfloor)$和$g(\lfloor\frac{n}{i}\rfloor)$，这题就算解决了。

$g(n)$可以参考`DIVCNT1`，我们有$O(n^{0.6})$的做法，或者暴力一些，有一个$O(n^{2/3})$的做法。

考虑求$f(n)$，回忆下$\sum_{i=1}^{n}\mu^2(i)$的$O(\sqrt{n})$做法，把这个（其实是个容斥）套用过来，可以得到

$$f(n)=\sum_{i=1}^{\lfloor\sqrt{n}\rfloor}\mu(i)\sum_{d=1}^{\lfloor\frac{n}{i^2}\rfloor}\sigma_0(i^2d)$$

考虑$\sigma_0(nm)$的值，可以简单推导（$d|nm$等价于$d=\frac{pm}{q},p|n,q|m,(p,q)=1$）得到

$$\sigma_0(nm)=\sum_{d|(n,m)}\mu(d)\sigma_0(\frac{n}{d})\sigma_0(\frac{m}{d})$$

带入上面的$f(n)$，可以得到

$$f(n)=\sum_{i=1}^{\lfloor\sqrt{n}\rfloor}\mu(i)\sum_{d|i^2}\sigma_0(\frac{i^2}{d})\mu(d)g(\lfloor\frac{n}{i^2d}\rfloor)$$

可以观察到，只有i和d是`squarefree number`的时候才需要计算一些东西，本地跑了跑，假设需要的$g(n)$都算好了，这个式子计算量差不多是$O(\sqrt{n})$。

于是线性筛预处理前$n^{2/3}$的$f(n)$和$g(n)$，比$n^{2/3}$大的那些用$O(\sqrt{n})$的做法算，就可以$O(n^{2/3})$算出所有的$f(\lfloor\frac{n}{i}\rfloor)$和$g(\lfloor\frac{n}{i}\rfloor)$了。

附代码：~~还没写。~~**写好了** (https://github.com/zimpha/competitive-programming/blob/master/spoj/DIVCNT3.cc)

update：和Min_25确认了下，他用的也是这个做法，经过一系列的优化，现在我的代码跑的比Min_25快了一点。

ftiasch (/u/ftiasch) and Nero (/u/Nero) like this. Reply

zimpha (/u/zimpha) 14 Jan

zimpha (https://post.icpc.camp/d/782/2) 习得了新的**做法** (https://gist.github.com/zimpha/25929b668aed23a8607d233d69d61064)，跑的比上面的快那么一点，而且更加通用。**洲阁筛应该可以退出积性函数求和的历史舞台了。**

解决了以下几个问题：

- 【UR #13】Sanrd (http://uoj.ac/submission/217929)
- **简单的函数** (https://loj.ac/submission/56015)
- Counting Divisors (general) (http://www.spoj.com/problems/DIVCNTK/)
- The Sum of Unitary Totient (http://acm.zju.edu.cn/onlinejudge/showProblem.do?problemId=5340)

简单的拟合了一下，在$n \le 10^{12}$以内，和$n^{0.69}$很接近，姑且复杂度就认为是这个好了。

(/\
ftiasch (/u/ftiasch), Nero (/u/Nero), and Magolor (/u/Magolor) like this.

Reply

ftiasch (/d/782-spoj-divcnt3/5) and Magolor (/d/782-spoj-divcnt3/7) replied to this.

L **LiveFish (/u/LiveFish)** 15 Jan

多谢楼上！其实两三年前我就觉得XX筛无法将预处理和实际计算的复杂度配平非常丑陋，但一直没有找到通用的O(n$^{2/3}$
)左右复杂度的算法。楼上完美解决了这个问题，非常令人激动！

Reply

ftiasch (/u/ftiasch) 15 Jan

zimpha (https://post.icpc.camp/d/782/3) 这是哪里习得的呢......

Reply

zimpha (/u/zimpha) 15 Jan

ftiasch (https://post.icpc.camp/d/782/5) 最早是在**Squarefree factorisations (http://www.javaist.com/rosecode/problem-347-Squarefree-factorisations-askyear-2016)**这题的forum里见Min_25用过，当时不以为意。前几天在SPOJ上造了**DIVCNTK (http://www.spoj.com/problems/DIVCNTK/)**这题，Min_25用这个方法轻松艹std。。然后就努力学习了下。

Reply

Magolor (/u/Magolor) 18 Jan

zimpha (https://post.icpc.camp/d/782/3) 请问楼主能不能详细解释一下这种做法（或提供一些资料），谢谢。

Reply

zimpha (/u/zimpha) 19 Jan

Magolor (https://post.icpc.camp/d/782/7) SPOJ的Blue.Mary大佬写了一个，你可以去看看。

link: **http://www.spoj.com/problems/TEES/ (http://www.spoj.com/problems/TEES/)**

Magolor (/u/Magolor) and Cydiater (/u/Cydiater) like this.

Reply

3 MONTHS LATER

crazy_cloud (/u/crazy_cloud) 18 Apr

请问如果要套上按n/x分块，每次求积性函数在这一段的区间的和的情况下，有什么比哈希更优秀的实现方法可以保证时间和空间复杂度吗？
如果是洲阁筛，因为每次转移时只是从前i-1个质数转移到前i个质数，数组可以直接滚动，然后在计算过程中统计对答案的贡献。但是这种筛法的转移可能会用到的质数并不是相邻的。

Reply

Write a Reply...