

《Factorisation》解题报告

厦门双十中学 汪文潇

1 题目来源

Codechef September Challenge 2014

2 题目大意

每个测试点有 T 组数据，每组数据给定一个大于 1 的整数 N ，要求给出若干个大于 1 的整数使它们的积为 N ，如果你给出了 M 个数，就会得到 M^2 的评分。

对于每个测试点，你的评分会是每组数据评分的和，要求评分越大越好。

单个测试点时间限制在 Codechef 上实际应是 0.6s，Tsinsen 上是 1s。

3 数据规模及约定

$$T = 100$$

其中 10 组，有 $2 \leq N \leq 10^{18}$ ， N 随机均匀分布。

另 15 组，有 $2 \leq N \leq 10^{18}$ ， N 不是随机均匀分布（对于 Tsinsen 上的数据， N 为 3 个质数的积）

另 50 组，有 $2 \leq N \leq 10^{1000}$ ， N 随机均匀分布。

另 25 组，有 $2 \leq N \leq 10^{1000}$ ，所有 N 的素因子不超过 10^{18} （据我猜测，原数据里 N 是由若干不超过 10^{18} 的随机数相乘得来，因而这里也采用了这种生成方式。对于这类数据，如果生成数据采用若干 10^{18} 级别的素数相乘，似乎很难比直接输出原值得到更多的分数）

4 关键字

因数分解 高精度运算

5 解题思路

显然，最优情况是把 N 分成其所有质因子的乘积，但是我们也知道，大数的质因数分解是难解的，时间限制也很紧张，这几乎是不可能的，所以要考虑数据的特性。

对于 $N \leq 10^{18}$ 的情况，可以通过现有算法如 Pollard Rho 直接分解出所有质因子，但这会消耗大量时间。注意到评分是每组数据给出的解的个数的平方和，与 $N \leq 10^{1000}$ 的另外 75 组数据比起来，其对评分的影响很小，是很不划算的。

实际上这部分数据可以直接通过枚举最小的若干质数来得到不差的解，对时间的占用也极小。

对于 $N \leq 10^{1000}$ 的情况，似乎最好的方法仍然是尝试若干质数。实际上不难发现，第 4 类数据能够容易地找到过百的质因子，远远超过前 3 类数据。考虑到评分方式的特性，在第 4 类上花费较多时间似乎更容易得到更高的分数。

因而应该先快速找出所有的第 4 类数据，然后对其进行进一步的分解，而对于前 3 类数据甚至可以直接忽略。

下面提供一种方法。

先用前 100 个质数对每个 N 进行尝试，然后把所有 N 按照当前已分出的质因子个数排序，然后对前 25 大分别尝试大约前 90000 个质数。

为了能够测试更多的素数，需要通过在高精度运算时减少模运算的次数来优化常数。

同时，还有一个优化是对要测试的所有质数进行打包，把若干个质数的乘积用来尝试，以减少高精度除法、取模的次数。但是经过尝试，效果并不明显，最终我决定在代码中去除这个优化。

至此，已经能在此题上得到相当高的分数（不低于考场最高分）。

6 数据制作

见数据规模。

特别的是，原出题人对于非随即均匀分布的部分的生成方式并未给出，系由猜测得出。