Upper Bound for Number Of Divisors

forthright48 on July 14, 2015

Given a number N, we can find its number of divisors using prime factorization. But when the number gets too big, it gets difficult to factorize, thus harder to find the number of divisors. So given a number N, can we estimate at most how many divisors N can have? That is, we want to find the upper bound for NOD.

NOD Upper Bounds

We proceed from loose to tighter bounds.

$$NOD(N) \leq N$$

A number which is greater than N cannot divide N. For $N > 2$, we can further say $NOD(N) < N$, but the improvement is negligible.

Can we make the limit tighter?

$$NOD(N) \leq \frac{N}{2} + 1$$

A divisor D, such that $\frac{N}{2} < D < N$ cannot divide N, since the result would be less than 2 and greater than 1, a fraction. So possible divisors are $D \leq \frac{N}{2}$ and N.

Better than before by half, but its possible to make it tighter.

$$NOD(N) \leq 2 \times \sqrt{N}$$

If we write $N = A \times B$, where $A \leq B$, then $A \leq \sqrt{N}$. Each of A, B forms a divisor pair for N. A can take any value from $1 \rightarrow \sqrt{N}$, so it is possible to form only $\sqrt{N}$ pairs of divisors. Thus, there can be at most $2 \times \sqrt{N}$ divisors for N.

$$NOD(N) \approx 2 \times \sqrt[3]{N}$$

I didn't know about this approximation until I read a blog post on Codeforces. From there I found out that in practice we can safely use $\sqrt[3]{N}$ as an upper bound, though I failed to understand the proof. Apparently, this approximation has been tested for $N \leq 10^{18}$, which is large enough to be used in programming contests.

Using Highly Composite Numbers

Sometimes we need upper bound for small values of N. For example, in a problem you might need to find an upper bound of NOD for $N \leq 10^9$. For such small values of N we could use NOD of largest Highly Composite Number (HCN), which is less than or equal to N, as an upper bound.

Read more about HCN here.

For programming contest, we could memorize values of HCN that comes frequently. Mainly 1344 for $N \leq 10^9$ and $103,680$ for $N \leq 10^{18}$.

Application

The upper bound for NOD is needed for complexity analysis.

Reference

1. Codeforces – Upper bound for number of divisors: http://codeforces.com/blog/entry/14463
2. forthright48 – Highly Composite Numbers: https://forthright48.com/2015/07/highly-composite-numbers.html

Category: CPPS, Number Theory

Previous:
Highly Composite Numbers

Next:
Prime Number Theorem