

IMO Shortlist 2011 N1

March 30, 2025

Problem. For any integer $d > 0$, let $f(d)$ be the smallest possible integer that has exactly d positive divisors (so for example we have $f(1) = 1, f(5) = 16$, and $f(6) = 12$). Prove that for every integer $k \geq 0$ the number $f(2^k)$ divides $f(2^{k+1})$.

Solution. Let's begin by considering the following problem first. We have a list of n positive reals (x_1, x_2, \dots, x_n) and a non-negative integer k . We have to multiply each x_i with a number 2^{a_i} where $a_i \in \mathbb{N}_0$ and $\sum_{i=1}^n a_i = k$ such that the resulting sum of the list is minimized.

Lemma. In order to minimize the sum the simple greedy algorithm is followed k times. Choose the smallest number in the list, let its index be t , then multiply the number at index t with 2. The list at the end is the list with minimum sum.

Proof. WLOG sort the list first, then first of all notice that for each k there exists a solution with list of coefficients $(2^{a_1}, 2^{a_2}, \dots, 2^{a_n})$ with $a_1 \geq a_2 \geq a_3 \geq \dots \geq a_n$. This is because if for some solution we have a j s.t. $a_j < a_{j+1}$ then we can swap a_j and a_{j+1} and get a solution whose sum that's not greater than original sum, we can follow the given step until the list is sorted in descending order.

Now notice that the resulting list with this greedy algorithm will be a list where each number is multiplied with some 2^{a_i} where $\sum_{i=1}^n a_i = k$. Let's prove it is the minimum such sum. First of all notice that if $k < 2$, then the following greedy algorithm results to the list with minimum sum. Now let's suppose for the sake of induction that the greedy algorithm always helps us find the list with minimum sum for all $k < m$ where $m \geq 2$ then the coefficient of x_1 in a solution will be 2^{a_1} where $a_1 \geq 1$, (because there exists a solution with $a_1 \geq a_2 \geq a_3 \geq \dots \geq a_n$). This implies that if we have a list $(2x_1, x_2, \dots, x_n)$ and we need to find its minimum sum for $k = m - 1$ then the resulting list's sum will be equal to the sum of our solution's list. But note that we can apply greedy algorithm here to form a list with minimum sum. But now note that in our first step we will convert our list to $(2x_1, x_2, \dots, x_n)$ and after that if we apply the greedy algo $k - 1$ times we will get the same exact list with minimum sum. QED

□

Now for the original proof note that $f(2^k)$ can only have at most k distinct prime factors, note that if it contains t prime factors then these prime factors are exactly first t primes (otherwise we can replace a bigger prime with a smaller prime and make the number smaller). Hence for both $f(2^k)$ and $f(2^{k+1})$ we will have prime factors from first $k+1$ smallest primes (p_0, p_1, \dots, p_k) . Now note that $f(2^t)$ for $t \leq k+1$ is the minimum number of form $\prod_{i=0}^{i=k} p_i^{(2^{a_i})-1}$ such that $\sum_{i=0}^{i=k} a_i = t$. This is equivalent to minimising $\prod_{i=0}^{i=k} p_i^{2^{a_i}}$ such that $\sum_{i=0}^{i=k} a_i = t$. Now minimising it is equivalent to minimising $\sum_{i=0}^{i=k} 2^{a_i} \log(p_i)$. Now note that considering the list $(\log(p_1), \log(p_2), \dots, \log(p_n))$ in the lemma we get that greedy algorithm will minimize the following sum. Now note that $\sum_{i=0}^{i=k} 2^{a_i} \log(p_i)$ for $\sum_{i=0}^{i=k} a_i = t$ is minimized by the greedy algorithm and for $t+1$ our greedy algorithm will obviously have greater coefficients for each p_i , this implies that $f(2^t) | f(2^{t+1})$ for all t . QED

Exploration. Basically the solution

Tags. NT , Combi