Thanks **ZZQ** for his improvement on my original observation.

Given a multiplicative function f, we'd like to calculate its prefix sum $F(n) = \sum_{i=1}^{n} f(i)$.

For any multiplicative function g, there exists another multiplicative function h satisfying that f=g*h where * denotes the Dirichlet convolution. Since they are multiplicative, it is enough to expand and check their values at prime powers: $f(p^e) = \sum_{i=0}^e g(p^i)h(p^{e-i})$ where p is prime. Specifically, f(p) = g(1)h(p) + g(p)h(1) = g(p) + h(p). Denote the prefix sum of g as G, i.e., $G(n) = \sum_{i=1}^n g(i)$. We have $F(n) = \sum_{i=1}^n f(i) = \sum_{i=1}^n \sum_{j|i} h(j)g(\frac{i}{j}) = \sum_{j=1}^n \sum_{k=1}^{\lfloor \frac{n}{j} \rfloor} h(j)g(k) = \sum_{j=1}^n h(j)G(\lfloor \frac{n}{j} \rfloor)$. If there is a multiplicative function g such that the corresponding h satisfies that h(p) = 0 at primes (or, equivalently, f(p) = g(p)), to calculate F we only need to consider those f which is a powerful number. (Explanation: if f is not powerful, f(f) is zero, thus they do not contribute to the sum and we can safely ignore them). Since there are f(f) powerful numbers, if f(f) can be calculated in time complexity f(f) can be calculated in time complexity f(f) can be calculated in time complexity f(f) can be f(f) can be calculated in time complexity f(f) or f(f) can be calculated in time complexity f(f) can be calculated in time complex

A few examples:

- 1. $f(p^e) = p$: we can let g(x) = x. The final time complexity is $O(\sqrt{n})$.
- 2. $f(p^e) = e + 1$: we can let $g(x) = \sigma_0(x)$, the divisor counting function. The final time complexity is $O(\sqrt{n})$.
- 3. $f(p^e) = p^e 1$: we can let $g(x) = \varphi(x)$, the Euler's totient function. The final time complexity is $O(n^{\frac{2}{3}})$.