Reduce to integer factorization. If $n = \prod_i p_i^{q_i}$, then the answer is $\prod_i (q_i + 1)$, which only depends on q_i. This shows after finding all prime factors of n that $\leq n^{\frac{1}{3}}$, we only need to determine whether the remaining number n' is a prime or the multiple of two primes, which can be done using primality test.
The running time for brute force is $O(n^{\frac{1}{3}})$, for pollard-rho is $\tilde{O}(n^{\frac{1}{6}})$, and for elliptic-curve is $L_{n^{\frac{1}{3}}}\left[\frac{1}{2}, \sqrt{2}\right]$. However, the current fastest algorithm general number field sieve (GNFS) is not improved, which has running time $L_n\left[\frac{1}{3}, \sqrt[3]{\frac{64}{9}}\right]$.
On the other hand, the following links indicate that this problem is probably not easier than integer factorization:

https://crypto.stackexchange.com/questions/68158/is-it-possible-to-check-if-a-number-is-the-product-of-two-primes-without-factori

https://mathoverflow.net/questions/3820/how-hard-is-it-to-compute-the-number-of-prime-factors-of-a-given-integer/10062#10062

References