LCM Summation

locked

by pruthvishalcodi1

Editorial by PruthvishE

∑LCM(i, n) = ((∑(d * ETF(d)) + 1) * n) / 2 where ETF(d) is eulers totient function of d and d belongs to the set of divisors of n.

Statistics

Difficulty: Medium
Required Knowledge: eulers totient function
Publish Date: Jul 04 2019

Set by PruthvishE

Problem Setter's code :

```
// C++ implementation of the approach
#include <bits/stdc++.h>
using namespace std;

#define n 1000002
#define ll long long int

ll phi[n + 2], ans[n + 2];

// Euler totient Function
void ETF()
{
    for (int i = 1; i <= n; i++) {
        phi[i] = i;
    }

    for (int i = 2; i <= n; i++) {
        if (phi[i] == i) {
            phi[i] = i - 1;
            for (int j = 2 * i; j <= n; j += i) {
                phi[j] = (phi[j] * (i - 1)) / i;
            }
        }
    }
}

// Function to return the required LCM sum
ll LcmSum(int m)
{
    ETF();

    for (int i = 1; i <= n; i++) {

        // Summation of d * ETF(d) where
        // d belongs to set of divisors of n
        for (int j = i; j <= n; j += i) {
            ans[j] += (i * phi[i]);
        }
    }

    ll answer = ans[m];
    answer = (answer + 1) * m;
    answer = answer / 2;
    return answer;
}

// Driver code
int main()
{
```

```
    int m = 5;

    cout << LcmSum(m);

    return 0;
}
```