

## 6266. Smallest Value After Replacing With Sum of Prime Factors

 $My\ Submissions\ (/contest/weekly-contest-324/problems/smallest-value-after-replacing-with-sum-of-prime-factors/submissions/)$ 

Back to Contest (/contest/weekly-contest-324/)

You are given a positive integer  $\, \, n \,$  .

Continuously replace n with the sum of its **prime factors**.

• Note that if a prime factor divides n multiple times, it should be included in the sum as many times as it divides n.

Return the smallest value n will take on.

Difficulty:	Medium
Total Submissions:	0
Total Accepted:	0
User Tried:	0
User Accepted:	0

## Example 1:

```
Input: n = 15
Output: 5
Explanation: Initially, n = 15.
15 = 3 * 5, so replace n with 3 + 5 = 8.
8 = 2 * 2 * 2, so replace n with 2 + 2 + 2 = 6.
6 = 2 * 3, so replace n with 2 + 3 = 5.
5 is the smallest value n will take on.
```

## Example 2:

```
Input: n = 3
Output: 3
Explanation: Initially, n = 3.
3 is the smallest value n will take on.
```

## Constraints:

•  $2 <= n <= 10^5$ 

```
JavaScript
1 ▼ function smallestValue(n) {
        if (n == 2 \mid \mid n == 4) return n;
3
        // Step 1: Find the prime factorization of n
        let factors = [];
4
5 ▼
        while (n \% 2 === 0) {
6
             factors.push(2);
7
             n /= 2;
8
        for (let i = 3; i <= Math.sqrt(n); i += 2) {
9 ▼
10 •
             while (n \% i === 0) {
                 factors.push(i);
11
12
                 n \neq i;
13
             }
14
        if (n > 2) {
15 v
16
             factors.push(n);
17
18
19
        // Step 2: Replace n with the sum of its prime factors
20
        n = factors.reduce((a, b) \Rightarrow a + b, 0);
21
22
        // Step 3: Repeat the process until n is a prime number
23 ▼
        while (true) {
             let isPrime = true;
24
25 ▼
             for (let i = 2; i \leftarrow Math.sqrt(n); i++) {
26 ▼
                 if (n \% i === 0) {
27
                      isPrime = false;
28
                     break;
                 }
```

```
30
31 ▼
             if (isPrime) {
32
                  return n;
             } else {
33 ▼
34
                  factors = [];
35 ▼
                  while (n \% 2 === 0) {
                      factors.push(2);
36
37
38
39 ▼
                  for (let i = 3; i \leftarrow Math.sqrt(n); i \leftarrow 2) {
40 ▼
                      while (n % i === 0) {
                           factors.push(i);
41
42
                           n /= i;
43
44
                  if (n > 2) {
45 ▼
46
                      factors.push(n);
47
48
                  n = factors.reduce((a, b) \Rightarrow a + b, 0);
49
             }
50
         }
51
    }
52
```

□ Custom Testcase

Use Example Testcases

Submission Result: Accepted (/submissions/detail/861441515/) @

More Details > (/submissions/detail/861441515/)

Run

♠ Submit

Share your acceptance!

Copyright © 2022 LeetCode

Help Center (/support) | Jobs (/jobs) | Bug Bounty (/bugbounty) | Online Interview (/interview/) | Students (/student) | Terms (/terms) | Privacy Policy (/privacy)

United States (/region)