

Problem 5. 2520 is the smallest number that can be divided by each of the numbers from 1 to 10 without any remainder. What is the smallest positive number that is evenly divisible by all of the numbers from 1 to 20?

Programming Knowledge required: How to write loops. How to compute LCM of two numbers. How to compute GCD of two numbers. How to extend it to n numbers.

Solution Outline: This problem is asking us to compute LCM of numbers from $[1, 20]$. Before computing LCM we need to know how we compute GCD of two positive numbers. GCD can be computed using Euler's algorithm, which has following recurrence:

$$GCD(a, b) = \begin{cases} a, & \text{if } b = 0 \\ GCD(b, a \% b) & \text{otherwise} \end{cases} \quad (1)$$

Now the LCM of two positive numbers can be computed using the following equation:

$$LCM(a, b) = \frac{a \times b}{GCD(a, b)} \quad (2)$$

To extend this to n numbers we compute it similar to running sum technique. Taking the LCM of current element and running LCM until now, updating the running LCM. This will be more clear when you look at the source code.

We will initialize `running_lcm` initialized to 1. Then we will run a *for* loop from 1-20, computing LCM and updating the `running_lcm`. Finally, `running_lcm` contains the final answer.

Python Solution

```
1 def GCD(a, b):
2     if b == 0:
3         return a
4     return GCD(b, a % b)
5
6 def LCM(a, b):
7     return a * b // GCD(a, b)
8
9 running_lcm = 1
10 for i in range(1, 21):
11     running_lcm = LCM(running_lcm, i)
12
13 print(running_lcm)
```
