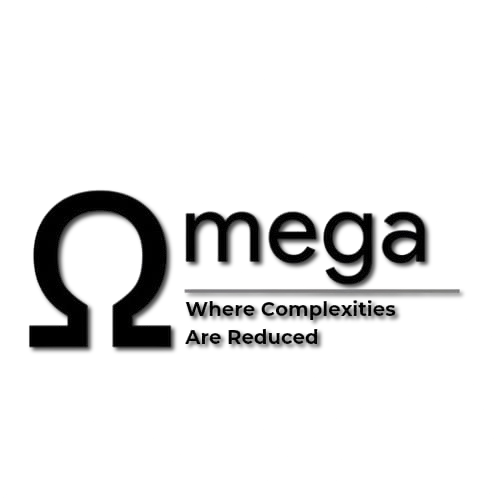
****

A competitive programmer’s Handbook

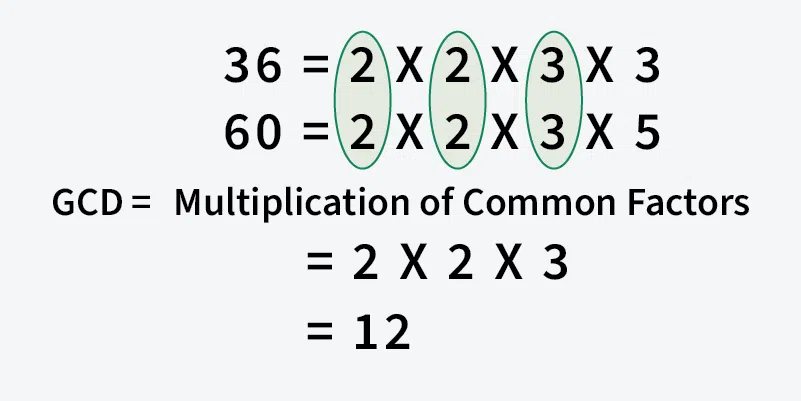
****

Welcome to the ultimate journey into the world of Competitive Programming and Data Structures. Whether you're a beginner or looking to sharpen your skills, this course is designed to equip you with the tools, techniques, and mindset to solve complex problems efficiently.

Let's dive into logic, algorithms, and beyond!

**Finding the GCD (Greatest Common Divisor)**

The GCD (or HCF - Highest Common Factor) of two numbers is the largest number that divides both numbers exactly.



**Example:**

* GCD of **20 and 28** is **4**.
* GCD of **98 and 56** is **14**.

**1. Euclidean Algorithm (Subtraction Method)**

The Euclidean algorithm finds the GCD by repeatedly subtracting the smaller number from the larger one until they become equal.

**Example:**

For **98 and 56**:

1. 98 - 56 = 42, so (42, 56)
2. 56 - 42 = 14, so (42, 14)
3. 42 % 14 == 0, so **GCD is 14**.

**PSUEDO CODE**:

def GCD (a, b):

if a == b:  
    return a

if a > b:

GCD (a – b, b)

else:

GCD (a, b – a)

GCD of 98 and 56 is 14

**2. Optimized Euclidean Algorithm (Modulo Method)**

Instead of subtraction, we use modulo operation to improve efficiency.

**PSUEDO CODE:**

def GCD (a, b):

if (b == 0):

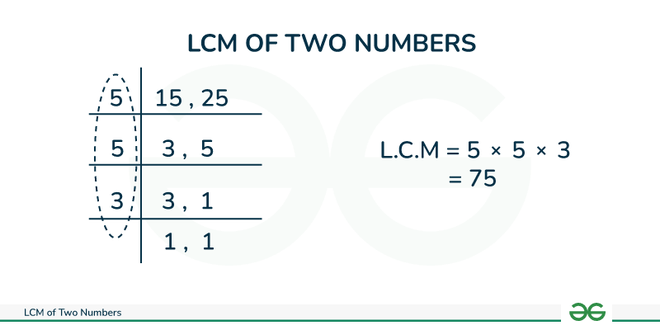
return a

return GCD (b, a % b)

GCD of 98 and 56 is 14

**Finding the LCM (Least Common Multiple)**

LCM of two numbers is the smallest number that is a multiple of both.



**Example:**

* LCM of **6 and 9** is **18**.

**1. Brute Force Method**

The basic approach is to find the maximum of both numbers, then iterate from that number, till we find a number that is completely divisible by both numbers.

**PSUEDO CODE:**

def LCM (a, b):

res = max (a, b)

while (True):

if (res % a == 0 and res % b ==0)

return res

res ++

return res;

**2. Efficient Formula-Based Approach**

An efficient solution is based on the below formula for LCM of two numbers ‘a’ and ‘b’.

a x b = LCM (a, b) \* GCD (a, b)

LCM (a, b) = (a x b) / GCD (a, b)

**PSUEDO CODE:**

def LCM (a, b):

return (a \* b) / GCD (a, b);

LCM of 6 and 9 is 18

**Practice Problems:**

<https://leetcode.com/problems/find-greatest-common-divisor-of-array/description/>

<https://www.geeksforgeeks.org/problems/lcm-and-gcd4516/1>

<https://codeforces.com/problemset/problem/1325/A>

<https://codeforces.com/problemset/problem/822/A>

<https://www.hackerrank.com/challenges/restaurant/problem>

<https://www.hackerrank.com/challenges/gcd-product/problem>

<https://atcoder.jp/contests/abc125/tasks/abc125_c>

<https://atcoder.jp/contests/abc148/tasks/abc148_c>

<https://atcoder.jp/contests/abc070/tasks/abc070_c>

**Useful links:**

<https://www.geeksforgeeks.org/program-to-find-lcm-of-two-numbers/>

<https://www.geeksforgeeks.org/program-to-find-gcd-or-hcf-of-two-numbers/>

<https://cp-algorithms.com/algebra/euclid-algorithm.html>