

## Copilot

Sure! Let's solve this problem step-by-step.

The **GCD (Greatest Common Divisor)** of two positive integers A and B is the largest positive integer that divides both A and B without leaving a remainder. One of the most efficient ways to find the GCD is by using the **Euclidean algorithm**.

## Euclidean Algorithm

The Euclidean algorithm is based on the principle that the GCD of two numbers also divides their difference. Here's how it works:

1. If B is 0, then the GCD is A.
2. Otherwise, replace A with B and B with the remainder of A divided by B.
3. Repeat the process until B becomes 0. The GCD is the last non-zero value of A.

## Python Implementation

Let's implement this algorithm in Python:

```
def gcd(a, b):  
    while b != 0:  
        a, b = b, a % b  
    return a  
  
# Example usage  
A = 48  
B = 18  
print("The GCD of", A, "and", B, "is", gcd(A, B))
```

## Explanation

1. **Function Definition:** We define a function gcd that takes two parameters a and b.
2. **While Loop:** The loop continues as long as b is not zero.
3. **Swap and Remainder:** Inside the loop, we update a to be b, and b to be the remainder of a divided by b ( $a \% b$ ).
4. **Return Statement:** When b becomes zero, the loop stops, and we return a as the GCD.

## Example

For A = 48 and B = 18:

- First iteration:  $a = 18, b = 48 \% 18 = 12$
- Second iteration:  $a = 12, b = 18 \% 12 = 6$
- Third iteration:  $a = 6, b = 12 \% 6 = 0$
- The loop stops, and the GCD is 6.

Feel free to ask if you have any questions or need further clarification!