

ChatGPT 4o

ChatGPT

GPTs erkunden

Heute

Python GCD Calculation

GCD Calculation in Python

Finding GCD in Python

Juli

Bedürfnisse in der Karriere

AI in Education Research Paper

Mai

Hectic Meeting with Youngsters

Masterarbeitsthema Diskussion Teil

Stay in Bari!

StratDiag & Tech: Structuring

New Apollo Software Packages

Dictionary Encoding in Databases

April

Committees Relevant in Ambition

Update Dictionary and Vectors

SQL CRUD with Dictionary-Encoding

Neuer Chat

Team-Arbeitsbereich hinzufügen

Mit einem Teamtarif...

Please consider this programming problem:
Given two positive integers A and B ,
find GCD of A and B .
Provide an example solution for the problem in the Python language.

Certainly! To solve the problem of finding the Greatest Common Divisor (GCD) of two positive integers A and B , we can use the **Euclidean algorithm**. This algorithm is an efficient way to compute the GCD, based on the principle that the GCD of two numbers also divides their difference.

Here's the step-by-step Python code to solve the problem:

```
python Code kopieren
def gcd(a, b):
    while b != 0:
        a, b = b, a % b
    return a

# Example usage:
A = 48
B = 18
result = gcd(A, B)
print(f"The GCD of {A} and {B} is {result}")
```

Explanation:

- 1. Euclidean Algorithm:
 - Start with two numbers A and B .
 - Replace A with B , and B with A modulo B .
 - Repeat this process until B becomes 0. The GCD is the last non-zero remainder.
- 2. Example:
 - Let's find the GCD of 48 and 18.
 - First step: $A = 48, B = 18$.
 - Calculate $48 \bmod 18 = 12$.
 - Replace A with 18 and B with 12.
 - Calculate $18 \bmod 12 = 6$.
 - Replace A with 12 and B with 6.
 - Calculate $12 \bmod 6 = 0$.
 - Since B is now 0, the GCD is the current value of A , which is 6.

Output:

```
python Code kopieren
The GCD of 48 and 18 is 6
```

This code works efficiently even for large values of A and B . The Euclidean algorithm has a time complexity of $O(\log(\min(A, B)))$, making it very efficient.

