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 Te
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-Encodi
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 find the GCD (Greatest Common Divisor) of two positive integers A and B, you can use the Euclidean algorithm. This algorithm is efficient and works by repeatedly replacing the larger number by its remainder when divided by the smaller number until one of the numbers becomes zero. The other number at this point is the GCD.

Here's how you can implement this in Python:

python — Code kopieren

```python
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. The `gcd` function takes two arguments `a` and `b`.
2. It uses a `while` loop that continues until `b` becomes 0.
3. Inside the loop, it updates `a` to `b`, and `b` to the remainder of `a` divided by `b` (`a % b`).
4. When `b` becomes 0, `a` contains the GCD of the original numbers, which is returned as the result.

Example:

For $A = 48$ and $B = 18$:

- $48 \% 18 = 12$, so now $a = 18$ and $b = 12$.
- $18 \% 12 = 6$, so now $a = 12$ and $b = 6$.
- $12 \% 6 = 0$, so now $a = 6$ and $b = 0$.

At this point, the loop ends, and the GCD is 6.

Output:

csharp — Code kopieren

```
The GCD of 48 and 18 is 6
```

This solution efficiently computes the GCD using the Euclidean algorithm.