# LU01.A07 - Rekursiver GGT und Trace Table



Implementieren Sie den rekursiven Algorithmus zur Berechnung des größten gemeinsamen Teilers (GGT) und erstellen Sie einen Trace Table, um den rekursiven Ablauf zu analysieren.

# Erklärung des rekursiven GGT-Algorithmus:

Der größte gemeinsame Teiler (GGT) zweier Zahlen ist der größte positive Divisor, der beide Zahlen ohne Rest teilt. Der rekursive GGT-Algorithmus basiert auf dem Euklidischen Algorithmus, der besagt, dass der GGT von zwei Zahlen a und b der gleiche ist wie der GGT von b und dem Rest der Division von a durch b. Die Rekursion endet, wenn einer der Werte 0 erreicht.

### **Anforderungen:**

- Verwenden Sie den folgenden rekursiven Algorithmus zur Berechnung des GGT.
- Erstellen Sie einen Trace Table, um die Änderung der Variablen und den Ablauf der Rekursion nachzuvollziehen.

#### **Algorithmus**

```
def ggt(a, b):
    if b == 0:
        return a
    else:
        return ggt(b, a % b)

# Beispielwerte
a = 48
b = 18
ergebnis = ggt(a, b)
print(f'GGT von {a} und {b} ist: {ergebnis}')
```

# **Aufgabe:**

- Analysieren Sie den Code Schritt für Schritt.
- Erstellen Sie einen Trace Table, der die Variablen a, b und den rekursiven Aufruf ggt (b, a % b) darstellt.
- Verwenden Sie die folgende Struktur für den Trace Table:

#### Schritt a b a % b Rekursiver Aufruf (ggt(b, a % b)) Rückgabewert

Füllen Sie den Trace Table basierend auf dem angegebenen Beispiel aus.

### **Beispielinput:**

a = 48

b = 18

## **Beispieloutput:**

GGT von 48 und 18 ist: 6

### **Beispiel für den Trace Table:**

Schritt	a	b	a % b	Rekursiver Aufruf	Rückgabewert
	_	_			
1	48	18	12	ggt(18, 12)	-
2	18	12	6	ggt(12, 6)	-
3	12	6	0	ggt(6, 0)	-
4	6	0	-	-	6

## **Erklärung des Trace Tables:**

- **Schritt 1**: Die Funktion wird mit a = 48 und b = 18 aufgerufen. Da b nicht 0 ist, erfolgt ein rekursiver Aufruf mit den neuen Werten ggt (18, 12).
- **Schritt 2**: Der Algorithmus wird nun mit a = 18 und b = 12 ausgeführt, was zu einem weiteren rekursiven Aufruf ggt (12, 6) führt.
- **Schritt 3**: Im nächsten Schritt sind die Werte a = 12 und b = 6. Es wird erneut rekursiv ggt(6, 0) aufgerufen.
- **Schritt 4**: Da b = 0 ist, wird der Wert von a zurückgegeben, was in diesem Fall 6 ist. Dies beendet die Rekursion und gibt den Wert zurück.



**Hinweis:** Verwenden Sie den Trace Table, um den rekursiven Ablauf des GGT-Algorithmus nachzuvollziehen und das Verständnis für die Funktionsweise der Rekursion zu vertiefen.

https://wiki.bzz.ch/ Printed on 2025/11/23 16:28

From:

https://wiki.bzz.ch/ - BZZ - Modulwiki

Permanent link:

https://wiki.bzz.ch/modul/m323/learningunits/lu01/aufgaben/tracetable2?rev=1722956400

Last update: 2024/08/06 17:00

