

Lehrstuhl für Software Engineering
RWTH Aachen University
Prof. Bernhard Rumpe
Mathias Pfeiffer, M. Sc.
Hendrik Kausch, M. Sc.
Dipl.-Inform. Deni Raco

Softwaretechnik
Übung
WS 2022/23

Aufgabenblatt 7

Abgabe: 05.12.2022 10:30 Uhr

Aufgabe 7.1 (6 Punkte)

Man muss dazu in der Lage sein, sich zielgerichtet und eigenständig in eine neue Programmiersprache einzuarbeiten. In dieser Aufgabe werden Sie Programme zur Berechnung des größten gemeinsamen Teilers zweier ganzer Zahlen in sechs verschiedenen Programmiersprachen implementieren.

Der größte gemeinsame Teiler $ggT(a, b)$ zweier ganzer Zahlen a und b ist definiert als die ganze Zahl n mit der Eigenschaft, dass sie Teiler von a sowie von b ist und dass jede ganze Zahl, die ebenfalls ein Teiler von a sowie von b ist, auch ein Teiler von n ist.

Betrachten wir zum Beispiel die Zahlen 12 und 18.

- Die Zahl 12 hat die Teiler 1, 2, 3, 4, 6, 12.
- Die Zahl 18 hat die Teiler 1, 2, 3, 6, 9, 18.
- Die gemeinsamen Teiler von 12 und 18 sind 1, 2, 3, 6.
- Der größte gemeinsame Teiler von 12 und 18 ist also $ggT(12,18) = 6$.

Der größte gemeinsame Teiler zweier Zahlen kann leicht mit dem euklidischen Algorithmus berechnet werden. Die folgende Abbildung zeigt diesen Algorithmus in Pseudocode:

01	In: Two integers a and b
02	Out: An integer
03	
04	if $a = 0$ then
05	return $\text{abs}(b)$
06	endif
07	if $b = 0$ then
08	return $\text{abs}(a)$
09	endif
10	while($b \neq 0$) do
11	$h := a \% b$
12	$a := b$
13	$b := h$
14	endwhile
15	return $\text{abs}(a)$

- Die Funktion `abs` ist eine Hilfsfunktion, die den Betrag einer ganzen Zahl zurückgibt. Beispielsweise gibt `abs(-3)` die Zahl 3, `abs(5)` die Zahl 5 und `abs(0)` die Zahl 0 zurück.
- Der Operator `%` ist der Modulo Operator und gibt den Rest der Ganzzahldivision zweier ganzer Zahlen zurück. Beispielsweise gibt `14 % 3` die Zahl 2 zurück.

Der Algorithmus gibt bei der Eingabe folgender Zahlen korrekte Resultate zurück:

- In: 12, 18 Out: 6
- In: 16, 20 Out: 4
- In: 120, 900 Out: 60
- In: 105, 26 Out: 1

Hinweise:

- Sie können die Lösungen der folgenden Aufgabenteile in Ihre Abgabe integrieren und müssen nicht zusätzliche Dateien einreichen. Erstellen Sie Ihre Lösungen bitte so, dass sie leicht kopierbar sind, falls möglich, um den Korrekturaufwand zu verringern.
- Falls Sie Probleme bei der Installation oder Anwendung eines Compilers oder Interpreters haben, dann versuchen Sie Ihre Probleme mithilfe einer Internetsuchmaschine zu beheben.
- Ihre Lösungen müssen jeweils den gesamten Quellcode beinhalten.
- Der Quellcode muss jeweils kompilieren.
- Sie können beliebig viele zusätzliche Methoden, Funktionen, Klassen, etc. implementieren und beliebige Standardbibliotheken nutzen.

Teilaufgabe a) (1 Punkt)

- Implementieren Sie den euklidischen Algorithmus zur Berechnung des größten gemeinsamen Teilers zweier ganzer Zahlen in der Programmiersprache **Java**. Implementieren Sie zusätzlich die `main` Methode zum simplen Testen Ihrer Implementierung. Lassen Sie sich zum Testen die berechneten Werte für die obigen vier Eingabepaare auf der Konsole ausgeben.
- Sie können einen Java Compiler auf der folgenden Webseite downloaden:
<https://www.oracle.com/de/java/technologies/javase-jdk15-downloads.html>

Benutzen Sie folgendes Gerüst für Ihren Quellcode:

```
01 public class GcdCalculator {
02     public static void main(String[] args) {
03         // Your code for testing ...
04     }
05
06     public int gcd(int a, int b) {
07         // Your code ...
08     }
09 }
```

Teilaufgabe b) (1 Punkt)

Implementieren Sie den euklidischen Algorithmus zur Berechnung des größten gemeinsamen Teilers zweier ganzer Zahlen in der Programmiersprache **C**. Implementieren Sie zusätzlich die `main` Methode zum simplen Testen Ihrer Implementierung. Lassen Sie sich zum Testen die berechneten Werte für die obigen vier Eingabepaare auf der Konsole ausgeben.

- Die folgende Webseite enthält Instruktionen zur Installation eines C Compilers:
http://hades.mech.northwestern.edu/index.php/Installing_a_C_Compiler_and_IDE

Benutzen Sie folgendes Gerüst für Ihren Quellcode:

```
01 #include <stdio.h>
02 #include <stdlib.h>
03
04 // Your code for declarations if necessary ...
05
06 int main() {
07     // Your code for testing ...
08 }
09
10 int gcd(int a, int b) {
11     // Your code ...
12 }
```

Teilaufgabe c) (1 Punkt)

Implementieren Sie den euklidischen Algorithmus zur Berechnung des größten gemeinsamen Teilers zweier ganzer Zahlen in der Programmiersprache **Python**.

- Sie können einen Python Compiler auf der folgenden Webseite downloaden:
<https://www.python.org/downloads/>

Benutzen Sie folgendes Gerüst für Ihren Quellcode:

```
01 def gcd(a, b):
02     // Your code for gcd here ...
03
04 print(gcd(12,18))    # 6
05 print(gcd(16,20))   # 4
06 print(gcd(120,900)) # 60
07 print(gcd(105,26))  # 1
```

Teilaufgabe d) (1 Punkt)

Implementieren Sie den euklidischen Algorithmus zur Berechnung des größten gemeinsamen Teilers zweier ganzer Zahlen in der Programmiersprache **JavaScript**.

- Sie können eine Anleitung zum Download eines und für die ersten Schritte mit einem JavaScript Interpreter auf der folgenden Webseite finden:
<https://gridscale.io/community/tutorials/node-js-beginner-guide/>

Benutzen Sie folgendes Gerüst für Ihren Quellcode:

```
01 function gcd(a, b) {  
02     // Your code ...  
03 }  
04  
05 console.log(gcd(12,18)) // 6  
06 console.log(gcd(16,20)) // 4  
07 console.log(gcd(120,900)) // 60  
08 console.log(gcd(105,26)) // 1
```

Teilaufgabe e) (1 Punkt)

Implementieren Sie den euklidischen Algorithmus zur Berechnung des größten gemeinsamen Teilers zweier ganzer Zahlen in der Programmiersprache **Go**. Implementieren Sie zusätzlich die `main` Methode zum simplen Testen Ihrer Implementierung. Lassen Sie sich zum Testen die berechneten Werte für die obigen vier Eingabepaare auf der Konsole ausgeben.

- Sie können einen Go Interpreter auf der folgenden Webseite downloaden:
<https://golang.org/doc/install>

Benutzen Sie folgendes Gerüst für Ihren Quellcode:

```
01 package main  
02  
03 import "fmt"  
04  
05 func main() {  
06     // Your code for testing ...  
07 }  
08  
09 func gcd(a, b int) int {  
10     // Your code ...  
11 }
```

Teilaufgabe f) (1 Punkt)

Implementieren Sie den euklidischen Algorithmus zur Berechnung des größten gemeinsamen Teilers zweier ganzer Zahlen in der Programmiersprache **Kotlin**. Implementieren Sie zusätzlich die `main` Methode zum Testen Ihrer Implementierung. Lassen Sie sich zum Testen die berechneten Werte für die obigen vier Eingabepaare auf der Konsole ausgeben.

- Instruktionen zum Download und zur Nutzung eines Kotlin Compilers können auf der folgenden Webseite gefunden werden:
<https://kotlinlang.org/docs/tutorials/command-line.html>

Benutzen Sie folgendes Gerüst für Ihren Quellcode:

```
01 fun main() {  
02     // Your code for testing ...  
03 }  
04  
05 fun gcd(a: Int, b: Int): Int {  
06     // Your code ...  
07 }
```

Aufgabe 7.2 (4 Punkte)

Teilaufgabe a) (2 Punkte)

Welche Aspekte/Eigenschaften eines Systems lassen sich durch UML Zustandsdiagramme (Statecharts) und UML Sequenzdiagramme modellieren? Wo liegt der wesentliche semantische Unterschied zwischen den beiden Diagrammtypen?

Teilaufgabe b) (2 Punkte)

Lassen sich UML Aktivitätsdiagramme und Zustandsdiagramme zur Modellierung eines Softwaresystems kombinieren? Wenn ja, welche Modellierungselemente eines Zustandsdiagramms könnten durch Aktivitätsdiagramme detailliert werden?