SWT WS 2020/21
Gruppe 010
Lennart Mesters, 343325
Laura Koch, 406310
Marc Ludevid, 405401
Til Mohr, 405959

Aufgabenblatt 6

Andrés Montoya, 405409
Dobromir I. Panayotov, 407763
Fabian Grob, 409195
Lennart Holzenkamp, 407761
Simon Michau, 406133
Tim Luther, 410886

Aufgabe 6.1

a) Java:

```java
public class GcdCalculator{

    public static void main(String[] args) {
        System.out.println("Ggt von 12 und 18: " + gcd(12, 18));
        System.out.println("Ggt von 16 und 20: " + gcd(16, 20));
        System.out.println("Ggt von 120 und 900: " + gcd(120, 900));
        System.out.println("Ggt von 105 und 26: " + gcd(105, 26));
    }

    public static int gcd(int a, int b) {
        int h = 0;

        if( a == 0){
            return Math.abs(b);
        } else if ( b == 0) {
            return Math.abs(a);
        }

        while( b != 0 ) {
            h = a \% b;
            a = b;
            b = h;
        }
        return Math.abs(a);
    }
}
```

b) C:

```c
#include<stdio.h>
#include<stdlib.h>

int gcd(int a, int b) {
    int h;

    if( a == 0 ) {
        return abs(b);
    } else if ( b == 0 ) {
        return abs(a);
    }

    while( b != 0 ) {
        h = a % b;
        a = b;
        b = h;
    }
    return abs(a);
}

int main() {
    printf("Ggt von 12 und 18: %d \n", gcd(12,18));
    printf("Ggt von 16 und 20: %d \n", gcd(16,20));
    printf("Ggt von 120 und 900: %d \n", gcd(120,900));
    printf("Ggt von 105 und 26: %d \n", gcd(105,26));
}
```

c) Python:

```python
def gcd(a,b):
    h = 0

    if a == 0:
        return abs(b)
    elif b == 0:
        return abs(a)

    while b != 0:
        h = a % b
        a = b
        b = h

    return abs(a)

print("Ggt von 12 und 18: " + str(gcd(12,18)))
print("Ggt von 16 und 20: " + str(gcd(16,20)))
print("Ggt von 120 und 900: " + str(gcd(120,900)))
print("Ggt von 105 und 26: " + str(gcd(105,26)))
```

d) JavaScript

```javascript
function gcd(a, b) {
    h = 0;
    if(a == 0) {
        return Math.abs(b);
    } else if ( b == 0) {
        return Math.abs(a);
    }

    while( b != 0 ) {
        h = a % b;
        a = b;
        b = h;
    }
    return Math.abs(a);
}

console.log("Ggt von 12 und 18: " + gcd(12,18));
console.log("Ggt von 16 und 20: " + gcd(16,20));
console.log("Ggt von 120 und 900: " + gcd(120,900));
console.log("Ggt von 105 und 26: " + gcd(105,26));
```

e) Go

```go
package main

import "fmt"

func main() {
  fmt.Println("Ggt von 12 und 18:", gcd(12, 18))
  fmt.Println("Ggt von 16 und 20:", gcd(16, 20))
  fmt.Println("Ggt von 120 und 900:", gcd(120, 900))
  fmt.Println("Ggt von 105 und 26:", gcd(105, 26))
}

func gcd(a, b int) int {

  if a == 0 {
    return abs(b)
  } else if b == 0 {
```

```
    return abs(a)
  }

  for b != 0 {
    h := a % b
    a = b
    b = h
  }
  return abs(a)
}

func abs(c int) int {
  if c < 0 {
    return -c
  }
  return c
}
```

f) Kotlin

```
import kotlin.math.abs

fun main() {
    println("Ggt von 12 und 18: " + gcd(12,18))
    println("Ggt von 16 und 20: " + gcd(16,20))
    println("Ggt von 120 und 900: " + gcd(120,900))
    println("Ggt von 105 und 26: " + gcd(105,26))
}

fun gcd(a: Int, b: Int): Int {
    var h: Int
    var x = a
    var y = b
    if ( x == 0) {
        return abs(y)
    } else if( y == 0 ) {
        return abs(x)
    }
    while( y != 0) {
        h = x % y
        x = y
        y = h
    }
    return abs(x)
}
```

SWT WS 2020/21
Gruppe 010
Lennart Mesters, 343325
Laura Koch, 406310
Marc Ludevid, 405401
Til Mohr, 405959

Aufgabenblatt 6

Andrés Montoya, 405409
Dobromir I. Panayotov, 407763
Fabian Grob, 409195
Lennart Holzenkamp, 407761
Simon Michau, 406133
Tim Luther, 410886

Aufgabe 6.2

a) Mit einem Zustandsdiagramm lässt sich darstellen in welchen Zuständen sich ein laufendes System befinden kann und mit welchen Ereignissen unter welchen Bedingungen sich diese Zustände verändern können.
Bei einem Sequenzdiagramm hingegen wird eine beispielhaft Folge von Abläufen dargestellt. Dabei wird der Austausch zwischen konkreten Objekten beschrieben.
Der Unterschied der beiden Diagramme liegt darin, dass sich mit dem Zustandsdiagramm das Verhalten eines Systems darstellen lässt und mit einem Sequenzdiagramm die Interaktion zwischen den einelnen Objekten veranschaulicht wird.

b) Um ein Softwaresystem zu modellieren, kann kann man Zustands- und Aktivitätsdiagramme kombinieren. Dabei kann man Aktionen in einem Zustandsdiagramm, die zu eine Wechesel des Zustands führen durch die Aktionen eines Aktivitätsdiagramms modelliert werden.