

Aufgabe 6 (Softwarequalitätssicherung) [20 PUNKTE]

Gegeben sei folgendes Java-Programm, das mit der Absicht geschrieben wurde, den größten gemeinsamen Teiler zweier Zahlen zu berechnen:

```
/** Return the Greatest Common Divisor of two integer values. */
```

```
@Override public int gcd(int a, int b) {
```

```
    if (a < 0 || b < 0) return gcd(-a, b);
```

```
    while (a != b) {
```

```
        if (a > b) a = a - b; else b = b - a;
```

```
    } return a;
```

```
}
```

a) Bestimmen Sie eine möglichst kleine Menge an Testfällen für das gegebene Programm, um (dennoch) vollständige Zweigüberdeckung zu haben.

b) Welche Testfälle sind notwendig, um die Testfälle der Zweigüberdeckung so zu erweitern, dass eine 100

c) Beschreiben Sie zwei allgemeine Nachteile der Anweisungsüberdeckung.

d) Das gegebene Programm enthält (mindestens) zwei Fehler. Bestimmen Sie jeweils eine Eingabe, die fehlerhaftes Verhalten des Programms verursacht. Nennen Sie den Fault und den Failure, der bei der gewählten Eingabe vorliegt. Sie müssen das Programm (noch) nicht verbessern.

e) Geben Sie für die gefundenen zwei Fehler jeweils eine mögliche Verbesserung an. Es reicht eine textuelle Beschreibung, Code ist nicht notwendig.

f) Erläutern Sie, warum es im Allgemeinen hilfreich sein kann, bei der Fehlerbehebung in einem größeren Programm die Versionsgeschichte miteinzubeziehen.