

## Übungsaufgabe 2 (Sommer 2022)

aa) 3 Punkte

Bei einem Unit-Test werden einzelne Teile oder Einheiten einer Anwendung, die sogenannten Units, auf ihre Funktionalität hin überprüft.

ab) 4 Punkte

**F.I.R.S.T**-Prinzipien für Unit-Tests

- **Fast:** Die Testausführung soll möglichst schnell sein, damit sehr oft getestet werden kann.
- **Independent:** Unit-Tests sollen unabhängig voneinander sein, damit sie in beliebiger Reihenfolge, parallel oder einzeln ausgeführt werden können.
- **Repeatable:** Führt man einen Unit-Test mehrfach aus, muss er immer das gleiche Ergebnis liefern.
- **Self-Validating:** Ein Unit-Test soll entweder fehlschlagen oder gut gehen. Diese Entscheidung muss der Test treffen. Es dürfen keine weiteren manuellen Prüfungen nötig sein.
- **Timely:** Man soll Unit-Tests vor der Entwicklung des Produktivcodes schreiben.
- u. a.

ba) 5 Punkte

Jeweils 1 Punkt pro richtigem Testergebnis.

Test	Ergebnis des Tests
1.	OK
2.	Fehler
3.	Fehler
4.	OK
5.	OK

bb) 4 Punkte

```
int berechneDifferenz (int zaehlerstandAlt, int zaehlerstandNeu) {
    int differenz = -1;
    if (zaehlerstandAlt >= 0 && zaehlerstandNeu >= 0) {
        if (zaehlerstandAlt <= zaehlerstandNeu) {
            differenz = zaehlerstandNeu - zaehlerstandAlt;
        }
    }
    return differenz;
}
```

Auch andere Lösungen sind möglich.

bc) 4 Punkte

Pro vollständig richtigem Beispiel 2 Punkte

Beispiele:

Nr.	zaehlerstandAlt	zaehlerstandNeu	differenz
1	-200	230	-1
2	-60	0	-1
3	-96	-35	-1