

Day 2: Red-Nosed Reports

Teil 1

Auch Tag 2 ist relativ einfach. Du benötigst keine besonderen Java-Befehle außer der `split()`-Methode aus Tag 1.

Tipps zur Vorgehensweise

- Speichere jede Zeile in einem `int`-Array oder in einer `ArrayList<Integer>`
- Ich empfehle, 3 separate Methoden zu erstellen, die für ein übergebenes Java-Array jeweils prüfen, ob:
 - ... alle Zahlen nur aufsteigend sind
 - ... alle Zahlen nur absteigend sind
 - ... die Differenz aller aufeinander folgenden Zahlen in Ordnung ist
- Prüfe dann pro Zeile, ob die Zahlen entweder aufsteigend oder absteigend sind und ob die Differenz okay ist.

Lösungsvorschlag

```
public boolean istAufsteigend(int[] levels) {
    for (int i = 1; i < levels.length; i++) {
        if (levels[i] <= levels[i-1]) // Gleichheit ist ein
        Ausschlusskriterium!
            return false;
    }
    return true;
}

public boolean istAbsteigend(int[] levels) {
    for (int i = 1; i < levels.length; i++) {
        if (levels[i] >= levels[i-1]) // Gleichheit ist ein
        Ausschlusskriterium!
            return false;
    }
    return true;
}

public boolean distanzOkay(int[] levels) {
    for (int i = 1; i < levels.length; i++) {
        if (Math.abs(levels[i-1] - levels[i]) < 1 || Math.abs(levels[i-1] -
        levels[i]) > 3) {
            return false;
        }
    }
}
```

```
}
return true;
}

public void partOne() {
    int safeReports = 0;

    for (String line: inputLines) {
        // teile an den Leerzeichen auf
        String[] strLevels = line.split(" ");

        // deklariere & initialisiere ein int-Array
        int[] levels = new int[strLevels.length];
        // "übersetze" die einzelnen Strings in Integer
        for (int i = 0; i < strLevels.length; i++) {
            levels[i] = Integer.parseInt(strLevels[i]);
        }

        //prüfe Auf-/Absteigend und Distanz
        if ((istAufsteigend(levels) || istAbsteigend(levels)) &&
distanz0kay(levels)) {
            safeReports++;
        }

    }

    System.out.println(safeReports);
}
```

From:
<https://www.info-bw.de/> -

Permanent link:
<https://www.info-bw.de/faecher:informatik:oberstufe:java:aoc:aoc2024:day02:start?rev=1733126632>

Last update: **02.12.2024 08:03**

