

# Aufgabe 5: Wortsuche-Dokument

Team-ID: 00070

Team: Gaußgamz

Bearbeiter/-innen dieser Aufgabe:

Nils Stäcker, Manuel Pielka, Benedict Weis

16. Oktober 2021

Lösungsidee.....	1
Umsetzung.....	1
Beispiele.....	2
Quellcode.....	2

## Lösungsidee

Um die Aufgabe 5 zu lösen, haben wir uns quasi zu jedem Gewicht zwei weitere Character gemerkt. Diese Characters bekommen entweder den Wert 0, 1 oder #. Anfangs bekommt das erste Gewicht eine 1 und eine 0. Die restlichen Gewichte bekommen zwei 0. Letzendlich wird dann jedes Gewicht, bei dem der erste Character eine 1 ist, wird dann auf die rechte Seite getan und jedes Gewicht bei dem der zweite Character eine 1 ist, wird dann auf die linke Seite getan. Das Gewicht, welches man nun abmessen kann, wird nun abgespeichert. Danach wird erst der zweite Character, so wie beim binär addieren (nur von links nach rechts) um eins erhöht. Hierbei wird beachtet, dass der zweite Character keine 1 sein kann, solange der erste Character des dazugehörigen Gewichts eine 1 ist. Dies wiederholt sich solange, bis beide Character ihr Maximum erreicht haben. Jetzt muss man nur noch schauen, ob das Gewicht von 10g bis 10000g in Zehnerschritten bereits abgespeichert ist und wenn es nicht abgespeichert ist, dass man sich dann das nächste Ergebnis raussucht.

## Umsetzung

Zuerst haben wir mithilfe der Klassen scanner und File die Gewichtsstücke eingelesen. Die Gewichte haben wir uns dann so in eine ArrayList abgespeichert, dass wenn wir zweimal das selbe Gewicht haben, dass das Gewicht dann auch zwei Mal in der ArrayList abgespeichert wird. Um uns die extra Character zu speichern, haben wir zwei Char Arrays erstellt, die genau so lang sind, wie wir auch einzelne Gewichtsstücke haben. Um unsere Zwischenergebnisse, so wie auch die Endergebnisse werden in einer Hashmap gespeichert. Um sicherzugehen, dass alle möglichen Kombinationen der Characters in den beiden Char Arrays zu berechnen, haben wir zwei ineinander geschachtelte while-Schleifen geschrieben.

## **Beispiele**

### **Quellcode**

**Der Code zum Einlesen der Datei:**

**Hier der Code um alle möglichen Ergebnisse berechnet:**

**Hier der Code, der alle aus allen Ergebnisse nur die gewollten holt:**

**Hier der Code, der die Ergebnisse dann ausgibt:**