



Algorithmen und Datenstrukturen

Sommersemester 2022

Übungsblatt 11

Abgabe: Dienstag, 19. Juli, 2022, 10:00 Uhr

Aufgabe 1: Bitstrings ohne aufeinanderfolgende Einsen (10 Punkte)

Gegeben eine natürliche Zahl $n \geq 1$, möchte man die Anzahl der n -stelligen bitstrings berechnen, welche keine zwei aufeinanderfolgende Einsen enthalten (für $n = 3$ bspw. wäre diese Anzahl 5, da 000, 001, 010, 100, 101 genau die 3-stelligen bitstrings sind, welche keine zwei aufeinanderfolgende Einsen enthalten).

- Geben Sie einen Algorithmus an, welcher dieses Problem in $\mathcal{O}(n)$ Zeit löst. Erklären Sie die Laufzeit. (5 Punkte)
- Implementieren Sie Ihre Lösung. Sie können dazu die Vorlage `DP.py` benutzen. Wenden Sie Ihren Algorithmus auf die Werte 10, 20 und 50 an und schreiben Sie die Ergebnisse in Ihre `erfahrungen.txt`. (5 Punkte)

Aufgabe 2: Summengleiche Partitionierung (10 Punkte)

Gegeben sei eine Menge $X = \{x_0, \dots, x_{n-1}\}$ mit $x_i \in \mathbb{N}$. Wir möchten bestimmen ob es eine Teilmenge $S \subseteq X$ gibt, so dass $\sum_{x \in S} x = \sum_{x \in X \setminus S} x$. Es ist nicht nötig S zu berechnen!

- Sei $W := \sum_{x \in X} x$. Stellen Sie eine rekursive Formel $s : \{0, \dots, n-1\} \times \{0, \dots, W\} \rightarrow \{\text{True}, \text{False}\}$ auf, mit $s(i, j) = \text{True}$ genau dann wenn es eine Teilmenge $S \subseteq \{x_0, \dots, x_i\}$ gibt mit $\sum_{x \in S} x = j$. Erläutern Sie wie s benutzt werden kann um obiges Problem in $\mathcal{O}(W \cdot n)$ Zeit zu lösen. (5 Punkte)
- Implementieren Sie Ihre Lösung. Sie können dazu die Vorlage `DP.py` benutzen. Lesen Sie die Mengen `set1.txt`, `set2.txt` und `set3.txt` ein und wenden Sie Ihren Algorithmus darauf an. Schreiben Sie die Ergebnisse in Ihre `erfahrungen.txt` (5 Punkte)