University of Freiburg
Dept. of Computer Science
Prof. Dr. F. Kuhn

Algorithms and Data Structures
Winter Term 2020/2021
Exercise Sheet 11

Exercise 1: Bitstrings without consecutive ones

Given a positive integer n, we want to compute the number of n-digit bitstrings without consecutive ones (e.g., for $n = 3$ this number is 5, as 000, 001, 010, 100, 101 are the 3-digit bitstrings without consecutive ones).

(a) Give an algorithm which solves this problem in time $\mathcal{O}(n)$. Explain the runtime.

(b) Implement your solution. You may use the template `DP.py`.

Exercise 2: Partitioning

Given a set $X = \{x_0, \ldots, x_{n-1}\}$ with $x_i \in \mathbb{N}$, we want to determine whether *there is* a subset $S \subseteq X$ such that $\sum_{x \in S} x = \sum_{x \in X \setminus S} x$ (it is not necessary to compute S).

(a) Let $W := \sum_{x \in X} x$. Give a recursive formula $s : \{0, \ldots, n-1\} \times \{0, \ldots, W\} \to \{\texttt{True}, \texttt{False}\}$ such that $s(i, j) = \texttt{True}$ if and only if there is a $S \subseteq \{x_0, \ldots, x_i\}$ such that $\sum_{x \in S} x = j$. Explain how s can be used to solve the above problem in time $\mathcal{O}(W \cdot n)$.

(b) Implement your solution. You may use the template `DP.py`. Run your algorithm on the sets given in `set1.txt`, `set2.txt` and `set3.txt`.