

PS Algorithmen und Datenstrukturen 2026

Aufgabenblatt 11

Aufgabe 31

Gegeben sei ein ungerichteter Graph $G = (V, E)$, von dem bekannt ist, dass jeder Knoten maximal d Nachbarn hat. Gesucht ist eine Färbung der Knoten, sodass jeder Knoten eine andere Farbe als jeder seine Nachbarn bekommt. Beschreiben Sie einen Greedy-Algorithmus, der dieses Problem unter Verwendung von maximal $d + 1$ Farben löst.

Die chromatische Zahl des Graphen $\chi(G)$ gibt die minimale Anzahl von Farben aus, die notwendig sind, um den Graphen zu färben. Beweisen oder widerlegen Sie, dass der von Ihnen beschriebene Algorithmus immer ein optimales Ergebnis liefert. Zeigen Sie also entweder, dass die von Ihnen benötigte Anzahl von Farben immer gleich $\chi(G)$ ist, oder finden Sie ein Gegenbeispiel, bei dem Ihr Algorithmus mehr als $\chi(G)$ Farben benötigt.

Aufgabe 32

Berechnen Sie für folgenden Text eine optimale Präfix-Kodierung mittels Huffman, geben Sie also den Codebaum und die Codes für die einzelnen Symbole an.

braunschweigerbierbrauerburschenbrauenbrausigesbraunschweigerbraunbier

Aufgabe 33

Dynamic Programming kann genutzt werden, um kostspielige Rekursionen zu vermeiden. Beispiele dafür sind:

- Berechnung von Fibonacci-Zahlen:

$$F_i = \begin{cases} i & \text{falls } i \leq 1 \\ F(n-1) + F(i-2) & \text{falls } i > 1 \end{cases}$$

- Berechnung der **Länge** der längsten gemeinsamen Teilfolge (LCS, siehe VO).

Wie können diese Rekursionen mittels Dynamic Programming möglichst speichereffizient implementiert werden? (**Beispiel:** Die Länge der LCS von zwei Strings der Länge n kann mit $n + O(1)$ zusätzlichem Arbeitsspeicher berechnet werden.)