



Prof. Dr. Thomas Wiemann

Algorithmen und Datenstrukturen

Übungsblatt 6

Wintersemester 2025/26

Aufgabe 6.1 (Nullen und Neunen)

Schreiben Sie eine Java-Klasse, die den Benutzer zur Eingabe eines Wertes N auffordert und dann (unter Verwendung einer Warteschlange) die kleinste positive ganze Zahl berechnet und ausgibt, die ein Vielfaches von N ist und deren Ziffern nur aus Nullen oder Neunen bestehen. Verwenden Sie dazu die `QueueArray`-Implementierung in Java. Hinweis: *Es könnte hilfreich sein, zu wissen, wie man eine Queue verwendet, um aufeinanderfolgende Binärzahlen zu generieren.*

Aufgabe 6.2 (Fibonacci-Folge)

Schreiben Sie ein Java-Programm, welches das n -te Glied der Fibonacci-Folge $f(n)$ ausgibt. Nutzen Sie dazu sinnvoll eine Queue. Erinnerung: $f(n) = f(n-1) + f(n-2)$ für $n \geq 3$.

Aufgabe 6.3 (Offenes Hashing)

Sei folgende Hashfunktion s für Strings gegeben: $f(s) = (\text{AnzahlVokale}(s) + \text{AnzahlZeichen}(s)) \bmod 8$. Beispiel: $f(\text{Martha}) = (2 + 6) \bmod 8 = 8 \bmod 8 = 0$. Berechnen Sie die Hashwerte der folgenden Elemente und fügen Sie diese in gegebener Reihenfolge nach dem Verfahren des offenen Hashings in eine Hashtabelle der Größe 8 ein: Patrizia, Sebastian, Maike, Lukas, Nele, Sarah, Matthias, Manuel.

Aufgabe 6.4 (Geschlossenes Hashing)

Gegeben seien die Schlüssel 1001, 1542, 429, 1320, 17, 900, 417, 2302, 1920 sowie die Hashfunktion $f(x) = \lfloor x/100 \rfloor \bmod N$ und eine Hashtabelle der Größe $N = 10$ (geschlossenes Hashing). Fügen Sie die Schlüssel einmal unter Verwendung von linearem Sondieren und einmal mittels quadratischen Sondieren in die Hashtabelle ein. Was passiert?