

# ÜBUNGSBLATT 7

Stack und Queue

## AUFGABE 7.1: VERKETTETE LISTEN

(2 PUNKTE)

1. Implementieren Sie einen Stack unter Verwendung einer einfach verketteten Liste durch folgende Funktionen: `StackInit`, `IsEmpty`, `IsFull`, `Push`, `Pop`, `Top`, und `PopTop`;
2. Implementieren Sie eine Queue unter Verwendung einer einfach verketteten Liste durch folgende Funktionen: `QueueInit`, `IsEmpty`, `IsFull`, `Put`, `Get`, und `Front`.

Verwenden Sie dafür die Folien 15 und 16 von der Vorlesung.

## AUFGABE 7.2: QUEUE

(2 PUNKTE)

Implementieren Sie eine Queue mit Hilfe von 2 Stacks, im Folgenden INBOX und OUTBOX genannt, wie folgt:

1. `Put(WERT)` legt das neue Element WERT auf dem Stack INBOX ab;
2. `Pop()` entfernt das oberste Element vom Stack OUTBOX. Falls der Stack OUTBOX leer ist, werden die Elemente aus dem Stack INBOX hintereinander entfernt und in umgekehrter Reihenfolge auf dem Stack OUTBOX abgelegt;
3. Bestimmen Sie die Komplexität der zwei Funktionen.

Verwenden Sie dafür den in der Aufgabe 7.1 implementierten Stack und bestimmen Sie die Komplexität der Algorithmen.

## AUFGABE 7.3: SIEB DES ERATOSTHENES

(2 PUNKTE)

Das Sieb des Eratosthenes ist ein Algorithmus zur Bestimmung aller Primzahlen kleiner oder gleich einer vorgegebenen Zahl. Implementieren Sie den Algorithmus, wie folgt:

1. Alle natürlichen Zahlen ab 2 und bis zu einem maximalen eingelesenen Maximalwert werden in einer Queue aufgeschrieben;
2. Die erste Zahl in der Queue ist immer eine Primzahl, wird von der Queue entfernt und als Primzahl ausgegeben;
3. Danach werden alle Ihre Vielfächer von der Queue gestrichen;
4. Der Algorithmus iteriert bis die Queue leer ist.

Verwenden Sie dafür, die in der Aufgabe 7.2 implementierten Queue und bestimmen Sie die Komplexität der Algorithmus.