

## Aufgabe 4f: Modulare Programmierung

### 1 Lernziele

Anwenden der modularen Programmierung sowie Vertiefen der Kenntnisse über die Gültigkeit und Sichtbarkeit von Variablen. Sie können erste Erfahrungen sammeln im gemeinsamen Erstellen von Software: Man einigt sich auf eine Schnittstelle und jeder Beteiligte löst unabhängig vom anderen einen Teil der Aufgabe.

Die Aufteilung, wer aus der Zweiergruppe welches Modul implementiert, bleibt Ihnen überlassen. Am Ende des Praktikums müssen beide Module zusammen kompilierbar sein und das Programm die gewünschte Funktionalität aufweisen.

### 2 Aufgabe

Erstellen Sie ein Testprogramm für eine Datenstruktur `priorityQueue_t`, die die folgenden Funktionen bereitstellt:

```
priorityQueue_t *createQueue(void);  
char isEmpty(priorityQueue_t *queue);  
void insert(priorityQueue_t *queue, int value);  
int extract(priorityQueue_t *queue);  
void destroyQueue(priorityQueue_t *queue);
```

Schreiben Sie ein Modul `queue.c`, das die obige Schnittstelle implementiert. Stellen Sie sicher, dass auf die Komponenten der Struktur nur innerhalb des Moduls zugegriffen werden kann. Überlegen Sie sich, welche Komponenten die Struktur `priorityQueue_t` haben muss und definieren Sie mittels `typedef` einen Datentyp.

Die Funktion `extract` liefert immer das kleinste in der Datenstruktur gespeicherte Element und entfernt es aus der Datenstruktur. Wenn Sie also bspw. die Elemente 8, 2, 19, 6 und 4 mittels `insert` einfügen, liefert `extract` den Wert 2 und die Datenstruktur enthält dann nur noch die Elemente 8, 19, 6 und 4.

Schreiben Sie ein Modul `main.c`, das die obige PriorityQueue-Implementierung testet. Dabei soll jede Funktion der Schnittstelle getestet werden. Der Test soll vollautomatisch ablaufen, d.h. es wird eine Übersicht über erfolgreiche und fehlgeschlagene Teiltests erstellt.

### 3 Testat

Voraussetzung ist jeweils ein fehlerfreies, korrekt formatiertes Programm. Der korrekte Programmlauf muss anhand einer Beispieleingabe nachgewiesen werden. Sie müssen in der Lage sein, Ihr Programm zu erklären.