

2. Programmieraufgabe Computerorientierte Mathematik II

Abgabe: 12.5.2023 über den Comajudge bis 17:00 Uhr

Aufgabenstellung

In dieser Programmieraufgabe geht es um eine erste Implementierung von Max-Heaps und Operationen auf diesen. Schreiben Sie hierfür einen Typ `MaxHeap` mit dem Attribut

- `keys` (Max-Heap als Vektor paarweise verschiedener ganzer Zahlen)

und implementieren Sie folgende Methoden:

- `MaxHeap(keys::Vector{Int})` Der Konstruktor soll eine übergebene Liste `keys` von paarweise verschiedenen, positiven ganzen Zahlen in einen Max-Heap umwandeln.
- `maximum(h::MaxHeap)` Gibt das maximale Element der `keys` des Max-Heaps zurück.
- `extractMax(h::MaxHeap)` Gibt das maximale Element der `keys` des Max-Heaps zurück, entfernt dieses aus `keys` und stellt die Max-Heap-Eigenschaft wieder her.
- `increaseKey(h::MaxHeap, i::Int, k::Int)` Erhöht den Eintrag von `keys[i]` auf `k`, falls `k` größer ist als `keys[i]`, und stellt anschließend die Max-Heap-Eigenschaft wieder her.
- `insert(h::MaxHeap, k::Int)` Fügt ein Element mit dem Schlüssel `k` in `keys` ein und stellt anschließend die Max-Heap-Eigenschaft wieder her.
- `heapSort(h::MaxHeap)` Führt Heapsort auf dem Max-Heap durch, so dass `keys` aufsteigend sortiert wird.