

## 2. Programmieraufgabe Computerorientierte Mathematik II

Abgabe: 15.5.2020 über den Comajudge bis 18:00 Uhr

### Aufgabenstellung

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

- **keys** (Max-Heap als Liste positiver, paarweise verschiedener ganzer Zahlen)

und implementieren Sie folgende Methoden:

- a) `__init__(self,keys)` Der Konstruktor soll eine übergebene Liste `keys` von paarweise verschiedenen, positiven ganzen Zahlen in einen Max-Heap umwandeln.
- b) `maximum(self)` Gibt das maximale Element des Max-Heaps `self.keys` zurück.
- c) `extractMax(self)` Gibt das maximale Element des Max-Heaps `self.keys` zurück, entfernt dieses aus `self.keys` und stellt die Max-Heap-Eigenschaft von `self.keys` wieder her.
- d) `increaseKey(self,i,k)` Erhöht den Eintrag von `self.keys[i]` auf `k`, falls `k` größer ist als `self.keys[i]`, und stellt anschließend die Max-Heap-Eigenschaft von `self.keys` wieder her. Andernfalls gibt es den String `'k too small'` zurück.  
**Anmerkung:** Beachten Sie, dass in Python Listen mit dem Index 0 beginnen, nicht mit 1. Die Angaben auf den Vorlesungsfolien sind entsprechend anzupassen.
- e) `insert(self,k)` Fügt ein Element mit dem Schlüssel `k` in `self.keys` ein und stellt anschließend die Max-Heap-Eigenschaft von `self.keys` wieder her.
- f) `heapSort(self)` Führt Heapsort auf dem Max-Heap `self.keys` aus. Da Heapsort `self.keys` aufsteigend sortiert, soll nach abgeschlossener Sortierung die Reihenfolge der Elemente in `self.keys` vertauscht werden, damit die Max-Heap-Eigenschaft wieder hergestellt wird.  
**Anmerkung:** Sie können zur Umkehr der Element-Reihenfolge von `self.keys` den Befehl `self.keys.reverse()` nutzen.

### Beispielaufufe und Tests

Zur Unterstützung wird Ihnen das ZIP-Archiv `Tests.zip` zur Verfügung gestellt. Dieses enthält drei Dateien: `test.py`, `students.in`, `correct.ot`. Kopieren Sie diese in ein Verzeichnis, in dem Ihre Abgabedatei liegt. Diese muss `co2_PA02solution.py` benannt sein. Anschließend führen Sie in einer Konsole im betreffenden Verzeichnis den Befehl

```
python3 test.py
```

aus. Es werden dann die Ausgaben der Probeaufrufe, die mittels der Probe-Eingaben in `students.in` ausgeführt werden, in der Konsole gedruckt. Diese können mit den Muster-Ausgaben in `correct.ot` verglichen werden.

*Sie können diese Aufgabe bis einschließlich Montag, den 25.5.2020, bei Ihrer Tutorin oder Ihrem Tutor vorstellen.*