

Übungsblatt 4

Aufgabe 1 (Liste rückwärts). Sei $L = \{16, 15, \dots, 1\}$ eine rückwärts sortierte Liste. Wie viele Vergleiche brauchen

1. Mergesort,
2. Quicksort (Pivot-Element ist erstes Element der Liste),
3. Quicksort (Pivot gleich $\text{median}(L[1], L[\lceil n/2 \rceil], L[n])$, “Median of Three”),
4. Heapsort

um die Liste zu invertieren?

Aufgabe 2 (Priority Queue). Eine *Priority Queue* (dt.: *Vorrangwarteschlange*) ist eine Datenstruktur, die mindestens die folgenden Operationen unterstützen muss:

1. *extract-max* Entferne das maximale Element
2. *insert-element* Füge ein Element der Priority Queue hinzu.

Beschreiben Sie eine Methode, wie Sie mit Hilfe eines Heaps eine *Priority Queue* effizient implementieren können. Geben Sie die Worst-Case Laufzeit für beide Operationen an.

Aufgabe 3. Paul kann beim Treppensteigen entweder eine Stufe oder zwei Stufen gleichzeitig erklimmen. Auf wie viele mögliche Arten kann Paul eine n -stufige Treppe hinauflaufen (wenn er immer mit demselben Fuß beginnt)? Lösen Sie die Frage mit Hilfe einer Rekursion.

Hinweis: Die Fragestellung ist eng verwandt dem Hasenfortpflanzungsmodell eines mittelalterlichen italienischen Mathematikers.