

# Praktische Informatik 1

Bachelor-Programm, Humboldt-Universität zu Berlin

13. Übung, Abgabe: 30.1./01.02.2006

## Aufgabe 1.

Lassen sich BubbleSort, MergeSort und QuickSort entsprechend ihrer Geschwindigkeit in eine Reihenfolge bringen? Diskutieren Sie eine Reihenfolge und mögliche Probleme beim Finden dieser Reihenfolge.

## Aufgabe 2.

Wie hoch ist der Rechenaufwand für eine Suche nach einem Element in einer sortierten Liste, einer unsortierten Liste und für einen Zugriff auf ein Element in einer Hash-Tabelle? Wie hoch ist der Rechenaufwand für das Einfügen von Elementen in diesen Strukturen? Können Sie die Datenstrukturen nach Geschwindigkeit ordnen?

## Aufgabe 3.

Das Sortierproblem liegt in der Komplexitätsklasse  $O(n \log n)$ ; das heißt, jeder Algorithmus der das Problem löst, benötigt mindestens  $\alpha n \log n$  Rechenoperationen im ungünstigsten Fall (für die „schwierigste“ Eingabe). Modifizieren Sie QuickSort so, dass der Algorithmus im günstigsten Fall eine lineare Laufzeit hat, also nur  $\alpha n$  Operationen benötigt! Was ist der günstigste Fall – oder ist die Aufgabe unlösbar?