Technische Universität München
Fakultät für Informatik
Lehrstuhl für Effiziente Algorithmen
Dr. Riko Jacob
Gero Greiner

Sommersemester 2009
Übungsblatt 7

Algorithmen für die Speicherhierarchie

Abgabetermin: 17.6.2009 **vor** *der Vorlesung*

Suchalgorithmen

Aufgabe 1

Analysieren Sie die folgenden Sortierverfahren im Cache-Oblivious-Modell:

- Bubblesort
- Gnomesort:

```
int i = 0;
while (i < n-1) {
  if (a[i] > a[i+1]) {
    vertausche a[i] und a[i+1];
    if (i > 0) i--;
  }
  else i++;
}
```

- Bucketsort mit k Buckets
- Heapsort

Aufgabe 2

Um cache-oblivious besser operieren zu können, kann der Heap eines Heapsorts mit einem vEB-Baum realisiert werden. Wieviele I/O-Zugriffe sind dann nötig? Kann dieser Heapsort einen Mergsort asymptotisch schlagen?

Aufgabe 3

Analysieren Sie die I/O-Komplexität von Quicksort im Cache-Oblivious-Modell für den worst-case sowie den best-case.

Aufgabe 4

Im besten Fall wählt Quicksort immer den Median als Pivot-Element. Dieser kann mit dem Median-of-Medians-Algorithmus in $\mathcal{O}(n)$ Rechenoperationen gefunden werden. Welche I/O-Komplexität ergibt sich für die Berechnung des Medians im I/O-Modell und im Cache-Oblivious-Modell?