

## 4. Übungsblatt

Ausgabe: 23. April 2013 · Besprechung: 6. Juni 2013

### SIMD I

Gegeben sei ein C-Array, der mit Zufallszahlen aus dem Bereich [0, 99] gefüllt ist:

```
uint8_t a[ARRAYSIZE];

for (unsigned int i = 0; i < ARRAYSIZE; i++)
    a[i] = rand() % 100;
```

Schreiben Sie eine Routine, die die Anzahl Array-Elemente ermittelt, die größer sind als 42. Benutzen Sie dazu

- eine skalare ("naive") Routine,
- eine skalare Routine, die "software predication" anwendet, und
- eine SIMD-optimierte Routine

und vergleichen Sie die entsprechenden Laufzeiten.

### SIMD II

In der Vorlesung wurde gezeigt, wie SIMD-Instruktionen genutzt werden, um 32-zu- $n$ -Bit-Kompression zu erzeugen, d. h., statt voller 32 Bits werden nur  $n$  Bits zur Speicherung aller Zahlen eines Arrays verwendet.

Schreiben Sie entsprechende Routinen zur Kompression und Dekompression für  $n \in \{7, 8, 9\}$ . Verwenden Sie dazu einmal eine skalare Implementation und einmal eine SIMD-optimierte Version; vergleichen Sie die Laufzeiten beider Varianten.

**Hinweis:** Bei der reinen Dekompression wird Ihr System möglicherweise begrenzt durch die Bandbreite Ihres Hauptspeichers. Um den reinen Durchsatz Ihrer Dekompressionsroutine zu messen, können Sie die Dekompression z. B. direkt mit einer Aggregation (etwa der Bildung einer Summe) kombinieren und somit das tatsächliche Schreiben des Ergebnisses vermeiden.