Datenstrukturen und Effiziente Algorithmen

Multiplikation langer Zahlen

Stefan Näher
FB IV – Informatik
Universität Trier
naeher@uni-trier.de

www.informatik.uni-trier.de

Rechnen mit beliebig langen ganzen Zahlen

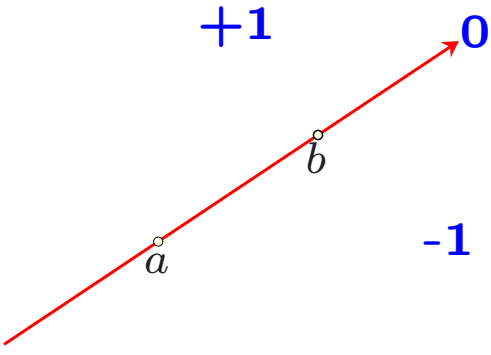
Verschlüsselung von Daten (Kryptographie)

Grundlage für rationale und reelle Zahlen

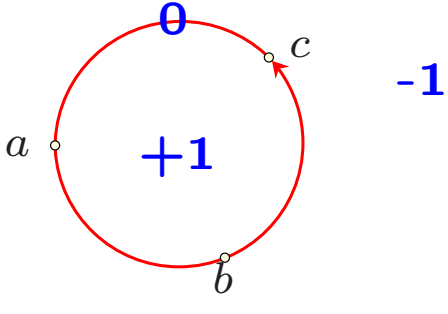
Vermeidung von Rundungsfehlern

Exakte Arithmetik für geometrische Probleme

orientation(a,b,c)



side_of_circle(a,b,c,d)



Schriftliches Multiplizieren (Schulmethode)

9 8 7 6 5 4 3 2 1 × 9 8 7 6 5 4 3 2 1 =

```
                    + 0 9 8 7 6 5 4 3 2 1
                  + 1 9 7 5 3 0 8 6 4 2
                + 2 9 6 2 9 6 2 9 6 3
              + 3 9 5 0 6 1 7 2 8 4
            + 4 9 3 8 2 7 1 6 0 5
          + 5 9 2 5 9 2 5 9 2 6
        + 6 9 1 3 5 8 0 2 4 7
      + 7 9 0 1 2 3 4 5 6 8
    + 8 8 8 8 8 8 8 8 8 9
------------------------------------
= 9 7 5 4 6 1 0 5 7 7 8 9 9 7 1 0 4 1
```

Multiplikation von zwei n-stelligen Zahlen

$A = a_{n-1}\ a_{n-2}\ \ldots\ a_0$

$B = b_{n-1}\ b_{n-2}\ \ldots\ b_0$

Wir nehmen Binärdarstellung an,

d.h. alle Ziffern sind aus $\{0, 1\}$.

Algorithmus 1: Die Schulmethode

$$a_{n-1} \ldots a_1\, a_0 \;\times\; b_{n-1} \ldots b_1\, b_0 \;=$$

$$\begin{aligned}
&(a_{n-1} \;\ldots\; a_1\; a_0) \cdot b_0 \quad \cdot 2^0 \\
+\;&(a_{n-1} \;\ldots\; a_1\; a_0) \cdot b_1 \quad \cdot 2^1 \\
&\vdots \\
+\;&(a_{n-1} \;\ldots\; a_1\; a_0) \cdot b_{n-1} \cdot 2^{n-1}
\end{aligned}$$

Beobachtung:

Die Zahl der elementaren Operationen ($+$,$\times$) auf Ziffern ist proportional zu n^2.

Folgerung:

Eine **Verdopplung** der Zahl der Stellen führt zu einer **Vervierfachung** der Laufzeit.

Algorithmus 2: Die Methode von Karatsuba

Teile die Faktoren in jeweils zwei Hälften.

$A = A_1 \cdot 2^{n/2} + A_0 \qquad B = B_1 \cdot 2^{n/2} + B_0.$

Dann gilt: $A \times B =$

$(A_1 \times B_1) \cdot 2^n + (A_1 \times B_0 + A_0 \times B_1) \cdot 2^{n/2} + A_0 \times B_0$

Idee: schreibe den mittleren Term als

$(A_1 + A_0) \times (B_1 + B_0) - A_1 \times B_1 - A_0 \times B_0$

Beobachtung:

Ein Produkt von n-stelligen Zahlen kann zurückgeführt werden auf **drei** Produkte von $n/2$-stelligen Zahlen:

$A_1 \times B_1 \qquad (A_1 + A_0) \times (B_1 + B_0) \qquad A_0 \times B_0$

Folgerung:

Eine **Verdopplung** der Zahl der Stellen führt zu einer **Verdreifachung** der ausgeführten Operationen.

Damit ist die Laufzeit proportional zu $n^{\log_2 3} = n^{1{,}58496\ldots}$.

Laufzeitanalyse

Anzahl der Rechenschritte als Funktion $T(n)$,

wobei n die Länge der Eingabe (Zahl der Ziffern).

Multiplikation:

Schulmethode:	$T(n) = n^2$
Karatsuba:	$T(n) = n^{1{,}58496}$
Schönhage/Strassen:	$T(n) = n \cdot \log n \cdot \log(\log n)$

Offene Frage: Geht es noch schneller ?