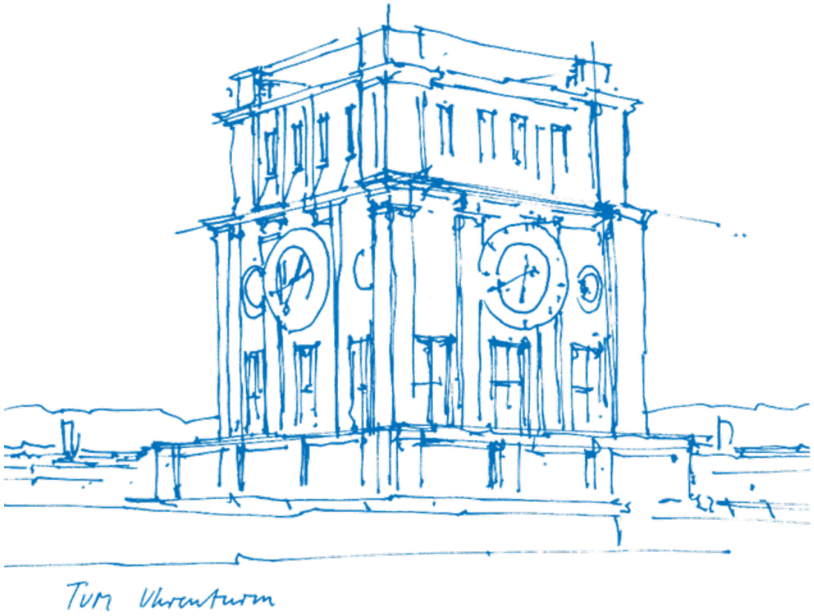
Gruppe 188 – Abgabe zu Aufgabe A325

Lukas Michael Englhauser Thua Duc Nguyen Michal Klakus
Technische Universität München
München, 25. September 2022



Berechnung der Konstante $\sqrt{2}$

Berechnung der Konstante $\sqrt{2}$

Die Quadratwurzel aus 2 ist die erste reelle Zahl, die als irrational entdeckt wurde.
In der Mathematik kann sie als $\sqrt{2}$ geschrieben werden. Die Approximation davon ist:

$$1.41421356237\ldots$$

Da $\sqrt{2}$ unendlich viele Nachkommastellen hat, kann sie nicht genau berechnet werden, sondern nur mit der folgenden mathematischen Gleichung angenähert werden:

$$\sqrt{2} = 1 + \sum_{i=1}^{\infty} \prod_{k=1}^{i} \frac{2k-1}{4k}$$

Berechnung der Konstante $\sqrt{2}$

Für die effiziente Berechnung des obigen Terms sind folgenden Verfahren verwendet:

- Karazuba-Multiplikation
- Binary Splitting

Wir werden im Rahmen unserer Präsentation:

- unseren Ansatz aufzeigen
- Ergebnisse präsentieren

Lösungsansatz

Darstellung großer Zahlen

Problem: builded-in Datentypen nicht geeignet

- begrenzte Anzahl der Nachkommastellen
- geringe Genauigkeit

Lösung: eine neue Datenstruktur namens **bignum**

- Zahlen in einem Feld von 32 Bit unsigned Integern allozieren
- Anzahl der belegten Blöcke und Nachkommablöcke speichern
- Little-Endian Format

Darstellung von Nachkommastellen

Warum Little-Endian Format?

- arithmetischer Operationen vereinfachen
- unnötige Schiebung vermeiden

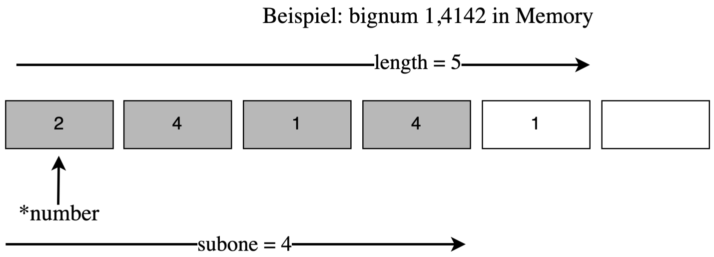


Abbildung: Einfache Darstellung der Datenstruktur bignum

Arithmetik

Ganzzahlarithmetik und die Rechnung mit Nachkommastellen für neue Datenstruktur benötigt:

- Addition Subtraktion
- Multiplikation mithilfe **Karazuba** Verfahren
- Division mithilfe **Newton-Raphson** Verfahren

Addition / Subtraktion

Pseudocode:

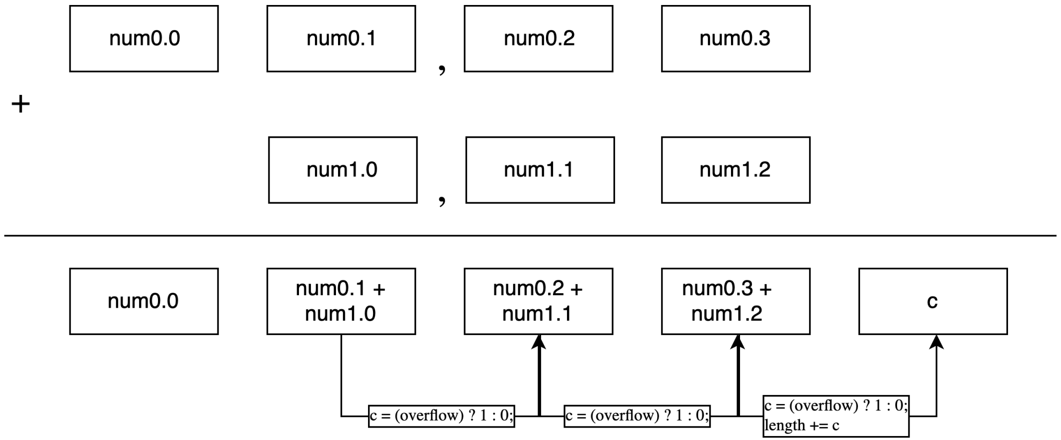


Abbildung: Darstellung der arithmetischen Operation

Karazuba-Multiplikation

Problem: Multiplikation von größeren Angaben ist relativ teuer
Beispiel: $2640 \cdot 1242$ braucht insgesamt 16 einstelligen Multiplikationen.

Die Multiplikation von zwei Zahlen der Länge n wird in θn^2 berechnet.

Karazuba-Multiplikation

Problem: Multiplikation von größeren Angaben ist relativ teuer
Idee:

- Wir teilen beide Zahlen auf zwei Teile der gleichen Länge auf:

$$2640 = (2600 + 40) = (26 * 10^2 + 40)$$
$$1242 = (1200 + 42) = (12 * 10^2 + 42)$$

Karazuba-Multiplikation

Problem: Multiplikation von größeren Angaben ist relativ teuer
Idee:

- Wir multiplizieren die beide Terme zusammen:

$$2640 \cdot 1242 =$$
$$(26 \cdot 10^2 + 40) \cdot (12 \cdot 10^2 + 42) =$$
$$10^4 \cdot 26 \cdot 12 + 10^2 \cdot (26 \cdot 42 + 40 \cdot 12) + 40 \cdot 42$$

Karazuba-Multiplikation

Problem: Multiplikation von größeren Angaben ist relativ teuer
Idee:

- Wir bemerken, dass die Anzahl von Multiplikationen immernoch 4 ist:

$$2640 \cdot 1242 = 10^4 \cdot 26 \cdot 12 + 10^2(26 \cdot 42 + 40 \cdot 12) + 40 \cdot 42$$

Karazuba-Multiplikation

Problem: Multiplikation von größeren Angaben ist relativ teuer
Idee:

- Wir ersetzen den Teil von unserer Gleichung:
  $2640 \cdot 1242 = 10^4 \cdot 26 \cdot 12 + 10^2 \cdot (26 \cdot 42 + 40 \cdot 12) + 40 \cdot 42$
- durch:
  $(26+40)(12+42) - 26 \cdot 12 - 40 \cdot 42$

Karazuba-Multiplikation

Problem: Multiplikation von größeren Angaben ist relativ teuer

Idee:

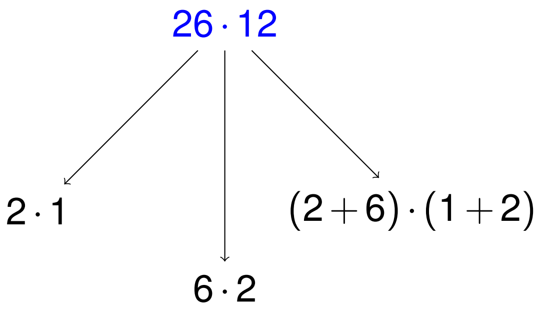
- Wir bemerken, dass die Anzahl an Multiplikationen um eins reduziert wurde.

$$2640 \cdot 1242 = 10^4 \cdot 26 \cdot 12 + 10^2 \cdot ((26+40)(12+42) - 26 \cdot 12 - 40 \cdot 42) + 40 \cdot 42$$

Karazuba-Multiplikation

Problem: Multiplikation von größeren Angaben ist relativ teuer

Idee: Wir wiederholen die Operationen rekursiv für jede nicht einstellige Multiplikation.



Karazuba-Multiplikation

Somit lässt sich die Anzahl von benötigten Operationen deutlich verringern.

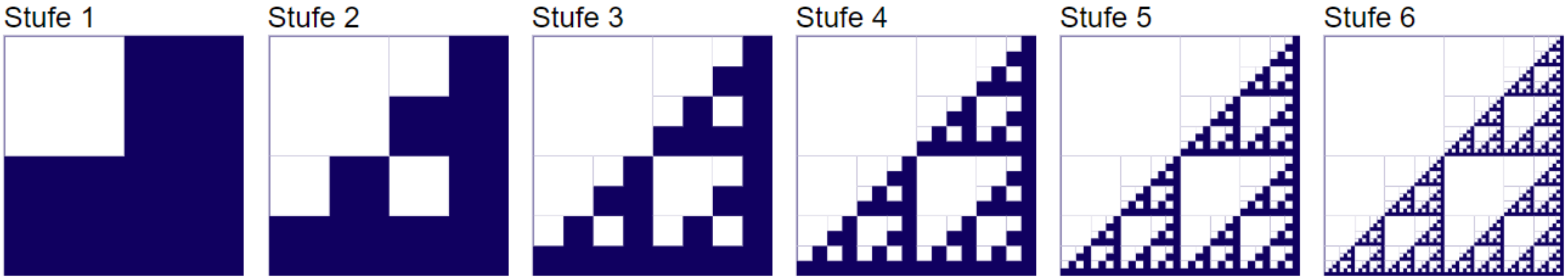


Abbildung: Visualisation

Source: https://mathgui.de/html/mult4.html

Newton-Raphson Verfahren

Quotient $\frac{N}{D}$ durch Multiplikation mit Annäherung des Kehrwerts des Nenners

- $D' = nomalize(D)$: Zähler D auf die Form $0,5 \leq D' \leq 1$ bringen.
- $N' = nomalize(N)$: gleich viele Shifts auf Zähler anwenden
- $X = \frac{48}{17} - \frac{32}{17} \times D'$: Schätzung X für den Kehrwert $\frac{1}{D}$

Newton-Raphson Verfahren

- $i = \lceil \log_2 \frac{P+1}{\log_2 17} \rceil$: Anzahl der Iteration für beliebigen Nachkommastellen P berechnen
- $X \times (2 - D' \times X)$: sukzessive genauere Schätzungen i Mal durchführen

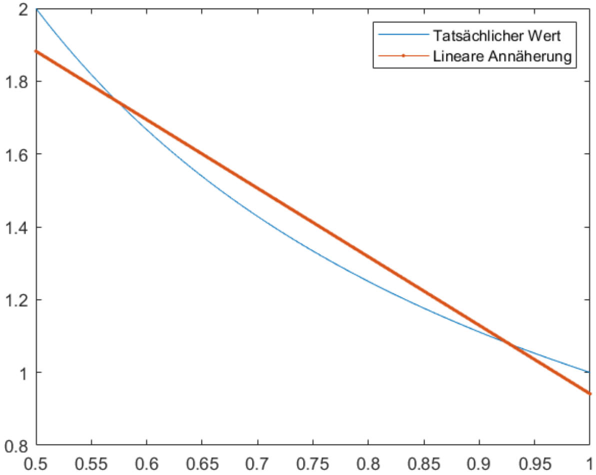
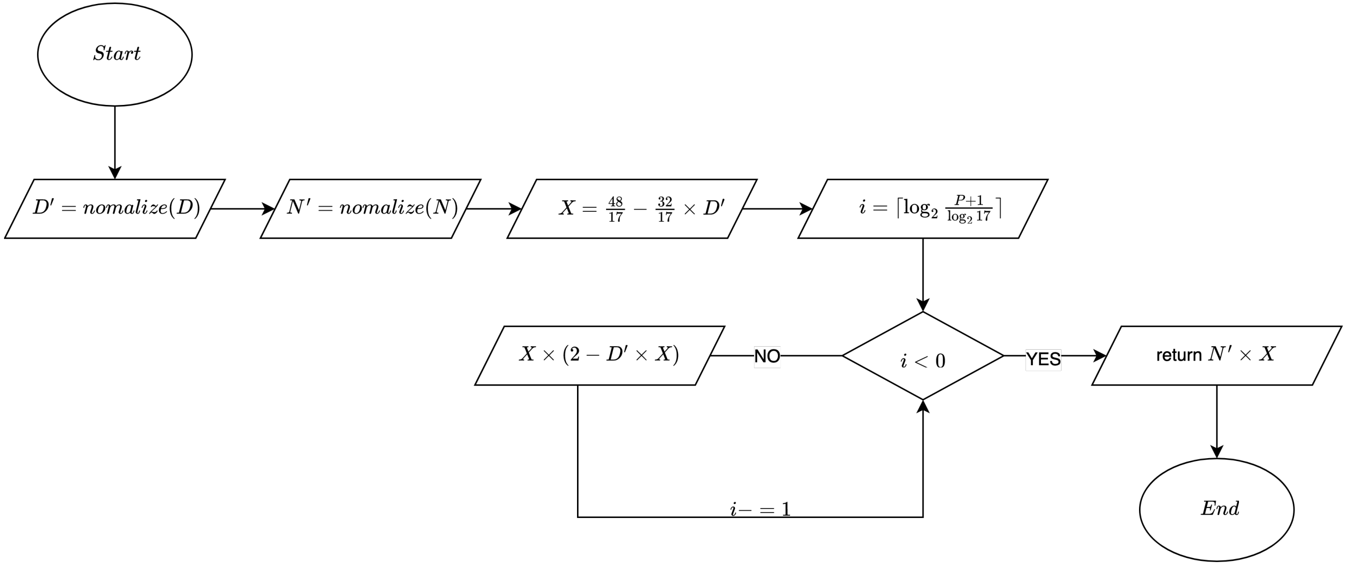


Abbildung: Darstellung der linearen Näherung mit minimalen maximalen Fehler

Newton-Raphson Verfahren

Pseudocode:



Newton-Raphson Verfahren

Vorteile:

- nur Subtraktion und Multiplikation notwendig
- Kehrwert des Nenners kann auf eine beliebige Anzahl der Nachkommastellen angenähert werden
- kostet relativ wenig Operationen. Für eine Division mit Genauigkeit 1000 Nachkommastellen kostet es 17 Multiplikation, 8 Subtraktion, und 2 Verschiebungen

Binary Splitting

Die gegebene Gleichung zur Annäherung der $\sqrt{2}$:

$$\sqrt{2} = 1 + \sum_{i=1}^{\infty} \prod_{k=1}^{i} \frac{2k-1}{4k}$$

lässt sich effizienter und schneller mithilfe des Binary Splitting Verfahrens berechnen.

Sei S_{n_1,n_2} für eine Folge für $n_1, n_2 \in N$ mit $n_1 < n_2$, und seien $a(n), b(n), p(n), q(n)$ Polynome mit ganzzahlige Koeffizienten.

$$S_{n_1,n_2} = \sum_{n_1 \leq n < n_2} \frac{a(n)}{b(n)} \cdot \frac{p(n_1) \cdots p(n)}{q(n_1) \cdots q(n)} = \sum_{n=n_1}^{n_2-1} \prod_{k=n_1}^{n} \frac{p(k)}{q(k)}$$

Binary Splitting

Zur Berechnung der Formel definieren wir rekursiv folgende Hilfsterme, wobei $n_1, n_2 \in N$ mit $n_1 < n_2$ und $n_m = \lfloor n_1 + n_2 \rfloor$ ist. Zusätzlich sind $a(n) = 1, b(n) = 1, p(n) = 2n - 1$ und $q(n) = 4n$ sind folgende Polynome gegeben.

$$P_{n_1,n_2} = p(n_1)...p(n_2 - 1)$$

$$Q_{n_1,n_2} = q(n_1)...q(n_2 - 1)$$

$$B_{n_1,n_2} = b(n_1)...b(n_2 - 1)$$

$$T_{n_1,n_2} = B_{n_1,n_2} Q_{n_1,n_2} S_{n_1,n_2} = \begin{cases} a(n_1)p(n_1) & \text{if} \quad n_1 = n_2 - 1 \\ B_{n_m,n_2} Q_{n_m,n_2} T_{n_1,n_m} + B_{n_1,n_m} Q_{n_1,n_m} T_{n_m,n_2} & \quad \textit{otherwise} \end{cases}$$

Binary Splitting

Daraus folgt, dass:

$$S(n_1,n_2) = \frac{T(n_1,n_2)}{Q(n_1,n_2)}$$

Mithilfe von dieser Formel, können wir $\sqrt{2}$ folgendermaßen darstellen:

$$\sqrt{2} = 1 + S(1,\infty) = 1 + \frac{T(1,\infty)}{Q(1,\infty)}$$

Ergebnisse

Ergebnisse

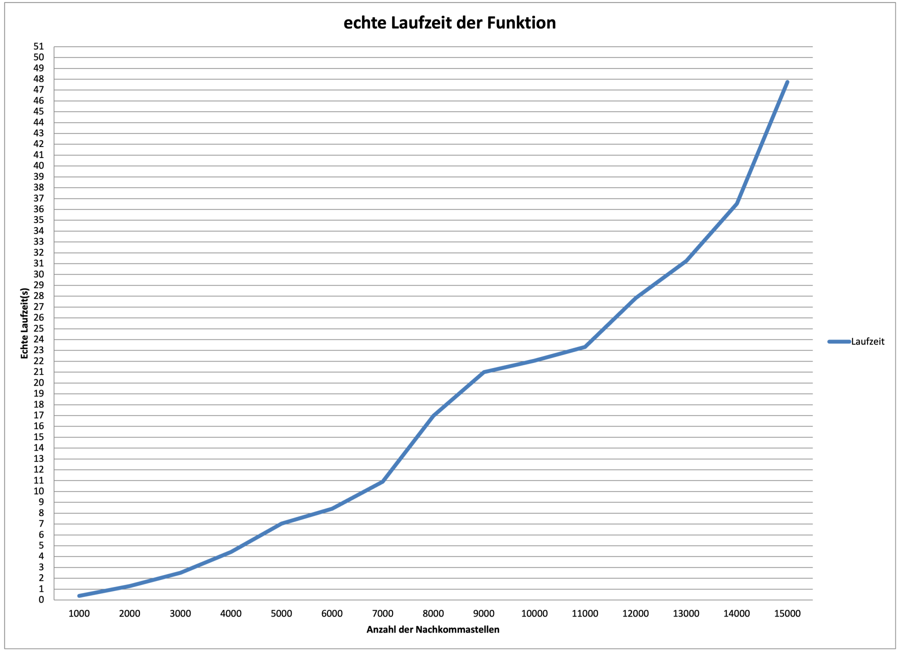


Abbildung: Performanzmessung auf Intel i7-6700

Zusammenfassung

Zusammenfassung

- Datenstruktur: Darstellung großer Zahlen, Darstellung von Nachkommastellen
- Arithmetik: Addition/Subtraktion, Karazuba, Newton Raphson
- Binary Splitting auf Approximationsformel angewandt
- Ergebnisse