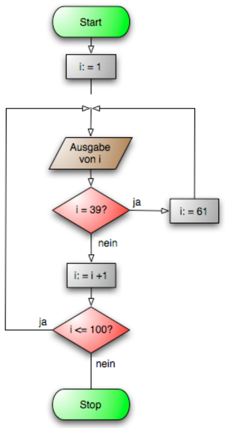
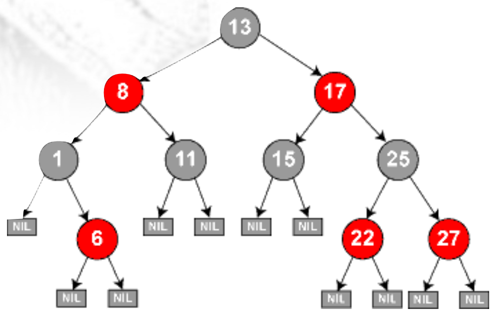
FH AACHEN
UNIVERSITY OF APPLIED SCIENCES

Algorithmen und Datenstrukturen





Version 1.26 vom 22, April 2021

Fachhochschule Aachen

Lehrgebiet für Angewandte Informatik

Prof. Dr. rer. nat. Jörg Striegnitz

Die Fibonacci-Funktion

- **Leonardo de Pisa (auch Fibonacci genannt) italienischer Mathematiker (um 1200)**
- **Berühmte Kaninchenaufgabe:**
  - Start: 1 Paar Kaninchen
  - Jedes Paar wirft nach zwei Monaten ein neues Kaninchenpaar
  - dann monatlich jeweils ein Paar
  - wie viele Kaninchenpaare gibt es in einem Jahr, wenn kein Kaninchen vorher stirbt?
  - 1,1,2,3,5,8,13, ...
  - Anzahl im n-ten Monat:

$$fib(n) = \begin{cases} 0 & \text{falls } n = 0 \\ 1 & \text{falls } n = 1 \\ fib(n-1) + fib(n-2) & \text{sonst} \end{cases}$$

Komplexität ?

Fibonacci: Rekursiver Algorithmus

```
public static int fib(int n) {
    if (n<=1) return n;
    else return fib(n-1) + fib(n-2);
}
```

- **Komplexität von `fib`:** $T_{\texttt{fib}}(n) \in O(2^n)$
- **Beweis**
  1. $fib(n)$ ist positiv: $\forall n > 0: \; fib(n) > 0$
  2. $fib(n)$ ist für $n > 2$ streng monoton steigend: $fib(n) < fib(n+1)$
  3. Abschätzung nach oben: Für alle $n > 3$ gilt

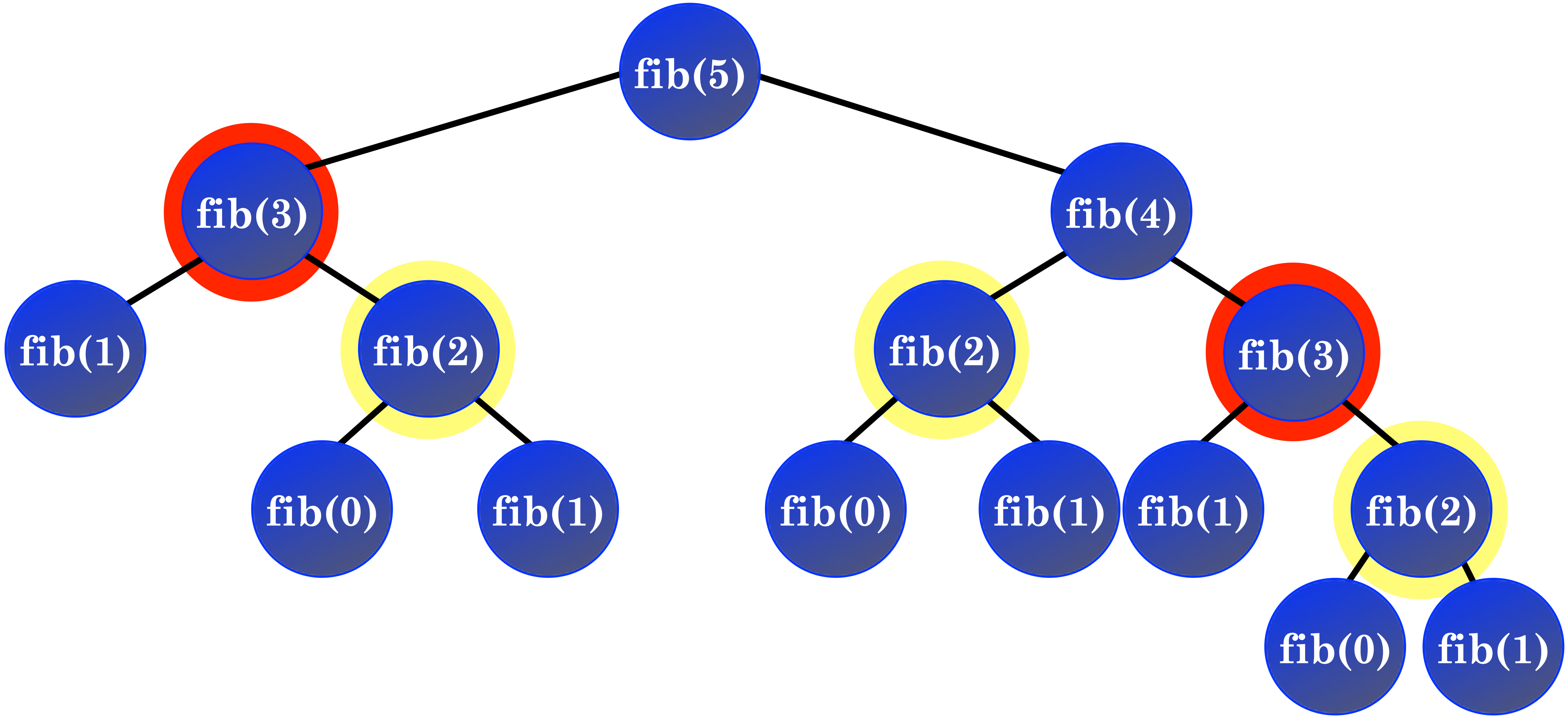
$$
\begin{aligned}
fib(n) &= fib(n-1) + fib(n-2) \\
&\quad (*\; fib(n-2) < fib(n-1) \Rightarrow 2 \cdot fib(n-1) > fib(n-2) + fib(n-1) \;*) \\
&< 2 \cdot fib(n-1) \\
&\quad (*\; fib(n-1) = fib(n-2) + fib(n-3) \;*) \\
&< 4 \cdot fib(n-2) \\
&< \cdots \\
&< 2^{n-1} \cdot fib(1) = \frac{1}{2} \cdot 2^n
\end{aligned}
$$

Also gilt: $fib(n) \in O(2^n)$

$\Theta(c^n)$ mit $c = \frac{1+\sqrt{5}}{2}$

Berechnungsbaum für $\mathrm{fib}(5)$

- **Zahlreiche Redundanzen in der Berechnung:**



- **Offenbar wird z.B. $\mathrm{fib}(2)$ dreimal berechnet - das kann man vermeiden, indem man sich Zwischenergebnisse merkt!**

Fibonacci: Iterativer Algorithmus

- **Idee**: jede Fibonacci-Zahl wird aus der Summe der beiden Vorgänger gebildet - merke diese Vorgänger in Variablen **`fib_new`** und **`fib_old`**

```
public static int fib_it(int n) {
   int p_previous=0, // fib(0)  - Vorvorgänger
       previous=1;   // fib(1)  - Vorgänger
   int current=0;    // Hier wird das Ergebnis abgelegt
   for (int i=0 ; i<n ; ++i) {
     // Die beiden Vorgänger aktualisieren
     p_previous = previous;
     previous   = current;
     // Ergebnis ermitteln
     // -> fib(i) = fib(i-1) + fib(i-2)
     current = current + p_previous;
   }
   return current;
 }
```

- **Komplexität von `fib_it`:** $T_{\texttt{fib\_it}}(n) \in \Theta(n)$

Rekursion vs. Iteration

- **Rekursive Formulierung oft**
  - eleganter / näher an mathematischer Formulierung
  - einfacher zu lesen
- **Iterative Lösungen sind oft effizienter**
  - aber komplizierter
- **Rekursion**
  - mächtiges, allgemeines Programmierprinzip
  - es gibt Sprachen, die Schleifen nur über Rekursion realisieren können (z.B. Haskell)
  - Jede rekursive Lösung kann in eine iterative überführt werden!
  - Rekursion muss nicht ineffizient sein - **tail recursion** (endständige Rekursion)

Fibonacci: Tail Recursion / Tail Calls

- **Idee:** Nach rekursivem Aufruf werden Werte auf Stapel nicht mehr verwendet
  - der aktuelle Aufrufkontext kann damit überschrieben werden! (Compiler muss entsprechende Optimierung unterstützen)
  - Ergebnis wird in Argument „akkumuliert“

```
public static int fib_tr(int n) {
  return fib_tr_help(n,0,1);
}

private static int fib_tr_help(int n,int current,int previous) {
  if (n<1) return current;
  else return fib_tr_help(n-1, current+previous, current);
}
```

- **Komplexität von `fib_tr`:** $T_{\texttt{fib\_tr}}(n) \in \Theta(n)$

Beispiel: Tail-Call Optimization

B Tail-Call Optimization (Elimination)

```
int fib_tr_help(int n,int current,int previous) {
  if (n<1) return current;
  else return fib_tr_help(n-1, current+previous, current);
}
```

Tail-Call Optimization

```
int fib_tr_help_opt(int n,int current,int previous) {
  while (true) {
    if (n<1) return current;
    else {
      // return fib_tr_help(n-1, current+previous, current);
      n = n - 1 ; int current_merk = current;
      current = current + previous;
      previous = current_merk;
    }
  }
}
```

1. Grundlagen

Einführung

- **Bei der Analyse von Algorithmen treffen wir häufig auf Rekursionsgleichungen; z.B.:**

$$F(n) = F(n-1) + F(n-2) \text{ für } n > 1 \text{ und } F(1) = 1, \;\; F(0) = 0$$

oder

$$T(n) = T(\lfloor \tfrac{n}{2} \rfloor) + T(\lceil \tfrac{n}{2} \rceil) \text{ für } n \geq 2 \text{ und } T(1) = 0$$

- **Diese Funktionen sind rekursiv**
  - Funktion taucht auch auf der rechten Seite auf
  - Auf rechter Seite wird Funktion auf „kleinere“ Argumente angewendet
  - Anfangswerte für ein oder mehrere „kleinste“ Argumente
  - Damit: Rekursion „gutmütig“ - Funktion einfach berechenbar

- **Problem: Laufzeitaussage**
  - Gesucht: **geschlossene Form**; d.h. Funktionsvorschrift
    - nicht rekursiv
    - ohne Summen- und Produktzeichen

1. Grundlagen

1. Grundlagen

1.14. Vergleich von Algorithmen
1.14.1. Einführung
1.14.2. Die Landau-Notation

1.15. Rekursionsgleichungen
1.15.1. **Einfache Strategien**
1.15.2. Das *Master-Theorem*
1.15.3. Erzeugende Funktionen

Für SoSe 2024
nicht relevant
nur für Interessierte!

Vorüberlegungen

- Manchmal kann man eine geschlossene Form recht leicht durch sukzessives Einsetzen „erraten“; z.B:

$$T(n) = \begin{cases} 1 & \text{falls } n \leq 1 \\ T(n-1) + n & \text{sonst} \end{cases}$$

$$\begin{aligned} T(n) &= T(n-1) + n \\ &= T(n-2) + (n-1) + n \\ &= T(n-3) + (n-2) + (n-1) + n \\ &\vdots \\ &= T(1) + 2 + \cdots + (n-2) + (n-1) + n \\ &= \frac{n \cdot (n+1)}{2} \end{aligned}$$

- Wir wollen dieses Vorgehen jetzt systematisieren.

Allgemeine Strategie - (1)

- Betrachten wir noch einmal unsere Rekursionsgleichung:

$$T(n) = \begin{cases} 1 & \text{falls } n \leq 1 \\ T(n-1)+n & \text{falls } n > 1 \end{cases}$$

- Wir rollen wir die Rekursionsgleichung ab (sukzessives Einsetzen) und merken uns in einem *Index* von T, wie oft wir die Gleichung für T angewendet haben:

$$\begin{aligned}
T_0(n) &= \underbrace{T(n-1)}_{=T((n-1)-1)\,+\,(n-1)} + n - 0 \\
T_1(n) &= \underbrace{T(n-2)}_{=T((n-2)-1)\,+\,(n-2)} + 2n - 1 \\
T_2(n) &= \underbrace{T(n-3)}_{=T((n-3)-1)\,+\,(n-3)} + 3n - 3 \\
T_3(n) &= T(n-4) + 4n - 6
\end{aligned}$$

- Wir wiederholen den Vorgang, bis wir die rechte Seite in Abhängigkeit vom **Einsetzungsindex** i ausdrücken können:

$$T_i(n) = T(n-(i+1)) + (i+1)\cdot n - \boxed{\sum_{j=0}^{i-1} j+1}$$

Allgemeine Strategie - (2)

$$T(n) = \begin{cases} 1 & \text{falls } n \leq 1 \\ T(n-1) + n & \text{sonst} \end{cases}$$

- Nach der i-ten Einsetzung gelangt man folglich zu:

$$\begin{aligned} T_i(n) &= T(n-(i+1)) + (i+1) \cdot n - \sum_{j=0}^{i-1} j + 1 \\ &= T(n-(i+1)) + (i+1) \cdot n - \frac{i \cdot (i+1)}{2} \end{aligned}$$

- Gemäß Rekursionsgleichung terminiert der Einsetzungsprozess bei $T(1)$

- Dies ist der Fall, wenn:

$$\begin{aligned} (n-(i+1)) &= 1 \\ i &= (n-2) \end{aligned}$$

- Also nach dem $(n$-$2)$-ten Ersetzungsschritt. Setzen wir i in $T_i(n)$ ein, erhalten wir eine **geschlossene Form**:

$$\begin{aligned} T(n) &= 1 + (n-1) \cdot n - \frac{(n-2) \cdot (n-1)}{2} \\ &= 1 + n^2 - n - \frac{n^2 - 3n + 2}{2} \\ &= \frac{n \cdot (n+1)}{2} \end{aligned}$$

Ein komplexeres Beispiel 1

B *worst case* der binären Suche I

Rekursionsgleichung: $T(n) \;=\; T\left(\left\lceil \frac{n-1}{2} \right\rceil\right) + 1 \text{ und } T(0) = 0$

Wie setzen wieder sukzessive ein, bis wir eine allgemeine Gleichung finden:

$$\begin{aligned}
T_0(n) &= T\left(\left\lceil \frac{n-1}{2} \right\rceil\right) + 1 \text{ und } T_0(0) = 0 \\
T_1(n) &= T\left(\left\lceil \frac{\left\lceil \frac{n-1}{2} \right\rceil - 1}{2} \right\rceil\right) + 1 + 1 \\
&= T\left(\left\lceil \frac{n-1}{4} - \frac{1}{2} \right\rceil\right) + 2 \\
T_2(n) &= T\left(\left\lceil \frac{\left\lceil \frac{n-1}{4} - \frac{1}{2} \right\rceil - 1}{2} \right\rceil\right) + 1 + 2 \\
&= T\left(\left\lceil \frac{n-1}{8} - \frac{1}{4} - \frac{1}{2} \right\rceil\right) + 3 \\
&\vdots \\
T_i(n) &= T\left(\boxed{\left\lceil \frac{n-1}{2^{i+1}} - \sum_{j=0}^{i-1} \frac{1}{2^{j+1}} \right\rceil}\right) + (i+1)
\end{aligned}$$

Ein komplexeres Beispiel 2

B *worst case* der binären Suche 2

Rekursionsgleichung: $T(n) \quad = \quad T\left(\left\lceil\frac{n-1}{2}\right\rceil\right) + 1 \text{ und } T(0) = 0$

Nach der i-ten Einsetzung erhalten wir also:

$$T_i(n) \quad = \quad T\left(\left\lceil \frac{n-1}{2^{i+1}} - \sum_{j=0}^{i-1} \frac{1}{2^{j+1}} \right\rceil\right) + (i+1)$$

Der Einsetzungsprozess endet, wenn die Rekursion terminiert, also wenn gilt:

$$\left\lceil \frac{n-1}{2^{i+1}} - \sum_{j=0}^{i-1} \frac{1}{2^{j+1}} \right\rceil \quad = \quad 0$$

Wir müssen herausfinden, nach dem wievielten Schritt das gilt - also lösen nach i auf

Ein komplexeres Beispiel 3

B *worst case* der binären Suche 3

Wir müssen herausfinden, nach dem wievielten Schritt das gilt - also lösen wir auf nach i

$$\left\lceil \frac{n-1}{2^{i+1}} - \sum_{j=0}^{i-1} \frac{1}{2^{j+1}} \right\rceil \quad = \quad 0$$

Zunächst vereinfachen wir die Reihe:

$$\begin{aligned}
\sum_{j=0}^{i-1} \frac{1}{2^{j+1}} &= \sum_{j=0}^{i-1} 2^{-j-1} \\
&= \sum_{j=0}^{i-1} 2^{-((i-1)-j)-1} \quad \text{(* Umordnen der Reihe *)} \\
&= \sum_{j=0}^{i-1} 2^{j-i} \\
&= \frac{1}{2^i} \cdot \sum_{j=0}^{i-1} 2^j \\
&= \frac{1}{2^i} \cdot \frac{1-2^i}{1-2} \quad \text{(* Auflösen geometr. Reihe *)} \\
&= \frac{1}{2^i} \cdot (2^i - 1) \\
&= 1 - \frac{1}{2^i}
\end{aligned}$$

B *worst case* der binären Suche 4

Beim Auflösen nach i müssen wir die obere Gaussklammer beachten:

$$\begin{aligned}
\left\lceil \frac{n-1}{2^{i+1}} - \left(1 - \frac{1}{2^i}\right) \right\rceil &= 0 \\
\left\lceil \frac{n-1}{2^{i+1}} + \frac{2}{2^{i+1}} \right\rceil - 1 &= 0 \\
\left\lceil \frac{n+1}{2^{i+1}} \right\rceil &= 1 \\
\left\lceil \frac{1}{2^{i+1}} \right\rceil &= \left\lceil \frac{1}{n+1} \right\rceil \\
\left\lceil \mathrm{ld}\left(\frac{1}{2^{i+1}}\right) \right\rceil &= \left\lceil \mathrm{ld}\left(\frac{1}{n+1}\right) \right\rceil \\
\lceil \mathrm{ld}(1) - (i+1) \rceil &= \lceil \mathrm{ld}(1) - \mathrm{ld}(n+1) \rceil \\
\mathrm{ld}(1) - (i+1) &= \mathrm{ld}(1) - \lceil \mathrm{ld}(n+1) \rceil \\
i &= \lceil \mathrm{ld}(n+1) \rceil - 1
\end{aligned}$$

Ein komplexeres Beispiel 5

B *worst case* der binären Suche 4

$$T_i(n) \quad = \quad T\left(\left\lceil \frac{n-1}{2^{i+1}} - \sum_{j=0}^{i-1} \frac{1}{2^{j+1}} \right\rceil\right) + (i+1)$$

Wir wissen jetzt, dass der Einsetzungsprozess für i=$\lceil \mathrm{ld}(n$+$1)\rceil$-1 terminiert und erhalten:

$$\boxed{\begin{aligned} T(n) &= T(0) + (\lceil \mathrm{ld}(n+1) \rceil - 1) + 1 \\ &= \lceil \mathrm{ld}(n+1) \rceil \end{aligned}}$$

1. Grundlagen

Das *Master-Theorem*

- Bei der Analyse von Algorithmen, die nach dem ***Divide-and-Conquer-*** **Verfahren** entwickelt wurden, treten oft Rekursionsgleichungen der folgenden Form auf:

$$T(n) = \begin{cases} 1 & \text{für} \quad n = 1 \\ a \cdot T(\frac{n}{b}) + d(n) & \text{für} \quad n > 1 \end{cases}$$

  mit Konstanten $a \geq 1$ und $b > 1$, sowie einer positiven Funktion $d(n)$.

- Sei $\gamma \geq 0$ und $\boldsymbol{d(n) \in O(n^\gamma)}$, dann gilt

$$T(n) \in \begin{cases} O\left(n^\gamma\right) & & a < b^\gamma \\ O\left(n^\gamma \cdot \log_b(n)\right) & \text{falls} & a = b^\gamma \\ O\left(n^{\log_b(a)}\right) & & a > b^\gamma \end{cases}$$

- Das Master-Theorem ist auch anwendbar, wenn der rekursive Aufruf eine der folgenden Gestalten hat:

$$a \cdot T\left(\left\lceil \frac{n}{b} \right\rceil\right) + d(n) \qquad \text{oder} \qquad a \cdot T\left(\left\lfloor \frac{n}{b} \right\rfloor\right) + d(n)$$

Das *Master-Theorem* - Beispiel

$$T(n) = \begin{cases} 1 & \text{für} \quad n = 1 \\ a \cdot T(\frac{n}{b}) + d(n) & \text{für} \quad n > 1 \end{cases}$$

$$T(n) \in \begin{cases} O\left(n^{\gamma}\right) & & a < b^{\gamma} \\ O\left(n^{\gamma} \cdot \log_b(n)\right) & \text{falls} & a = b^{\gamma} \\ O\left(n^{\log_b(a)}\right) & & a > b^{\gamma} \end{cases}$$

B Beispiele zum *Master-Theorem*

- Beispiel 1:

$$T(n) = 9 \cdot T\left(\frac{n}{3}\right) + n$$

Mit $a = 9$, $b = 3$, $d(n) = n$ und $d(n) \in O(n^{\gamma})$ (für $\gamma = 1$) folgt mit dem $Master\text{-}Theorem$ (3. Fall, denn es gilt $9 > b^1$):

$$T(n) \in O\left(n^{\log_3 9}\right) = O(n^2)$$

- Beispiel 2:

$$T(n) = T\left(\frac{2 \cdot n}{3}\right) + 1$$

Mit $a = 1$, $b = \frac{3}{2}$, $d(n) = 1$ und $d(n) \in O(n^{\gamma})$ (für $\gamma = 0$) folgt mit dem $Master\text{-}Theorem$ (2. Fall, denn es gilt $1 = 1$):

$$T(n) \in O\left(n^0 \cdot \log_{\frac{3}{2}} n\right) = O\left(\frac{\log_2 n}{\log_2 \frac{3}{2}}\right) = O\left(\log_2 n\right)$$

1. Grundlagen

Einsatz von Erzeugenden Funktionen

- **Das soeben dargestellte Verfahren eignet sich zur Lösung von homogenen linearen Rekursionsgleichungen:**

D Lineare Rekursionsgleichung

Eine Rekursionsgleichung der Form

$$f(n) = a_1 \cdot f(n-1) + a_2 \cdot f(n-2) + \cdots + a_k \cdot f(n-k) + b_k$$

mit den Anfangsbedingungen

$$\begin{aligned} f(1) &= b_1 \\ &\vdots \\ f(k-1) &= b_{k-1} \end{aligned}$$

heißt lineare Rekursionsgleichung k-ter Ordnung. Gilt $b_k = 0$ sprechen wir von einer homogenen Rekursionsgleichung, ansonsten von einer inhomogenen.

Erzeugende Funktionen

- **Ziel: geschlossene Formel für Fibonacci-Zahlen**
- Betrachte folgende Reihe: $F(x) = \sum_{n=0}^{\infty} \mathrm{Fib}(n) \cdot x^n$
  - $F(x)$ ist die **Erzeugende Funktion** der Fibonacci-Reihe - die Fibonacci-Zahlen sind genau die Koeffizienten dieser Reihe.
  - Differenzieren wir $F(x)$:

$$
\begin{aligned}
F^{(1)}(x) &= \sum_{n=1}^{\infty} n \cdot \mathrm{Fib}(n) \cdot x^{n-1} \\
F^{(2)}(x) &= \sum_{n=2}^{\infty} n \cdot (n-1) \cdot \mathrm{Fib}(n) \cdot x^{n-2} \\
\vdots &\quad \vdots \\
F^{(j)}(x) &= \sum_{n=j}^{\infty} n \cdot (n-1) \cdot \ \cdots \ (n-j+1) \cdot \mathrm{Fib}(n) \cdot x^{n-j}
\end{aligned}
$$

  - Offenbar gilt: $F^{(j)}(0) = j! \cdot \mathrm{Fib}(j) \quad \text{bzw.} \quad \mathrm{Fib}(j) = \frac{1}{j!} \cdot F^{(j)}(0)$

Die Fibonacci-Zahlen können über die Ableitungen der Erzeugenden Funktion rekonstruiert werden!

Einschub: 0^0

- **Die Festlegung der Bedeutung von 0^0 ist uneinheitlich**
  - undefiniert (häufig in der Analysis)
  - 0, wegen $0 \cdot 0 = 0$

- **Wir setzen $0^0=1$, was z.B. mit Hinblick auf**

$$\underbrace{\sum_{i=0}^{0} q^i}_{=0} = \underbrace{\frac{1-q^0}{1-q}}_{=0}$$

konsistenter erscheint

Eigenschaften der Erzeugende Funktion (1)

- **Warum dieser Umweg?**
- **Indexverschiebungen können einfach behandelt werden:**

Sei

$$F(x) = \sum_{n=0}^{\infty} a_n \cdot x^n$$

die erzeugende Funktion zu einer Folge a_n. Dann ist

$$\begin{aligned} x \cdot F(x) &= x \cdot \sum_{n=0}^{\infty} a_n \cdot x^n \\ &= x \cdot \left(a_0 \cdot x^0 + a_1 \cdot x_1 + a_2 \cdot x^2 + \cdots\right) \\ &= a_0 \cdot x^1 + a_1 \cdot x^2 + a_2 \cdot x_3 + \cdots \\ &= \sum_{n=0}^{\infty} a_{n-1} \cdot x^n \end{aligned}$$

Beachte: a_n=0 für n<0

Erzeugende Funktion zu $a_{n\text{-}1}$

- Indexverschiebung um eine Stelle nach links $\cong$ Multiplikation mit x
- **Jetzt:** Geschlossene Formel für Erzeugende Funktion meist einfach

Eigenschaften der Erzeugende Funktion (2)

- **Warum dieser Umweg?**
- **Geschlossene Form für folgende Reihe:**

$$\boxed{\sum_{n=0}^{\infty} \gamma^n \cdot x^n = \frac{1}{1-\gamma \cdot x}}$$

$$\begin{aligned}
\Leftrightarrow \quad & (1-\gamma \cdot x) \cdot \sum_{n=0}^{\infty} \gamma^n \cdot x^n = 1 \\
\Leftrightarrow \quad & 1 \underbrace{-\gamma x + \gamma x}_{0} \underbrace{-\gamma^2 x^2 + \gamma^2 x^2}_{0} \underbrace{-\gamma^3 x^3 + \gamma^3 x^3}_{0} -\gamma^4 x^4 \cdots = 1 \\
\Leftrightarrow \quad & 1 = 1
\end{aligned}$$

Lösen von rek. Gl. mit Erzeugenden Funktionen 1

- **Schritt 1**: Aufstellen der **Erzeugenden Funktion**

  Wir starten mit der üblichen Definition der Fibonacci-Zahlen

  undefinierte Stellen setzen wir 0

Anfangsbedingungen

$$\mathrm{Fib}(n) = \begin{cases} \boxed{\begin{array}{ll} 0 & \text{falls } n \leq 0 \\ 1 & \text{falls } n = 1 \end{array}} \\ \boxed{\begin{array}{ll} \mathrm{Fib}(n-1) + \mathrm{Fib}(n-2) & \text{sonst}(n > 1) \end{array}} \end{cases}$$

Rekursionsgleichung

Die Erzeugenden Funktion ist:

$$F(x) \quad = \quad \sum_{n=0}^{\infty} \mathrm{Fib}(n) \cdot x^n$$

Lösen von rek. Gl. mit Erzeugenden Funktionen 2

Schritt 2: Einsetzen der Rekursionsgleichung

$$\mathrm{Fib}(n) = \mathrm{Fib}(n-1) + \mathrm{Fib}(n-2) \quad \text{falls } n \geq 2$$

Die Erzeugende Funktion ist:

$$F(x) = \sum_{n=0}^{\infty} \boxed{\mathrm{Fib}(n)} \cdot x^n$$

Die Rekursionsgleichung dürfen wir hier nicht anwenden; sie gilt erst ab ***n*=2**

Wir ziehen daher die ersten beiden Glieder aus der Summe:

$$F(x) = \underbrace{\mathrm{Fib}(0) \cdot x^0}_{=0 \cdot 1} + \underbrace{\mathrm{Fib}(1) \cdot x^1}_{=1 \cdot x^1} + \sum_{n=2}^{\infty} \mathrm{Fib}(n) \cdot x^n$$

Jetzt können wir die **Rekursionsgleichung einsetzen** und erhalten:

$$\begin{aligned} F(x) &= x + \sum_{n=2}^{\infty} (\mathrm{Fib}(n-1) + \mathrm{Fib}(n-2)) \cdot x^n \\ &= x + \sum_{n=2}^{\infty} \mathrm{Fib}(n-1) \cdot x^n + \sum_{n=2}^{\infty} \mathrm{Fib}(n-2) \cdot x^n \end{aligned}$$

Lösen von rek. Gl. mit Erzeugenden Funktionen 3

Schritt 3a: Angleichen von Index und Exponent

Wir haben jetzt

$$F(x) \quad = \quad x + \boxed{\sum_{n=2}^{\infty} \mathrm{Fib}(n-1) \cdot x^n} + \boxed{\sum_{n=2}^{\infty} \mathrm{Fib}(n-2) \cdot x^n}$$

Beobachtung: Summanden der **ursprünglichen Erzeugenden** sehr ähnlich!

Unterschiede:

- Argument und Exponent sind „verschoben“
- Summe startet bei 2

Die Indizes (Argumente an Fib) gleichen wir durch Ausklammern von x an:

Erinnerung: Indexverschiebung nach links ≅ Multiplikation mit x

$$F(x) \quad = \quad x + \boxed{x \cdot \sum_{n=2}^{\infty} \mathrm{Fib}(n-1) \cdot x^{n-1}} + \boxed{x^2 \cdot \sum_{n=2}^{\infty} \mathrm{Fib}(n-2) \cdot x^{n-2}}$$

Schritt 3b: Elimination der Summen

Wir haben jetzt

$$F(x) \quad = \quad x + x \cdot \sum_{n=2}^{\infty} \mathrm{Fib}(n-1) \cdot x^{n-1} \quad + \quad x^2 \cdot \sum_{n=2}^{\infty} \mathrm{Fib}(n-2) \quad \cdot x^{n-2}$$

Wenn wir die Summen bei 0 starten lassen, müssen wir Summanden abziehen, die dann zu viel sind:

$$\begin{aligned} F(x) \quad = \quad & x + \\ & x \cdot \left[\left(\boxed{\sum_{n=0}^{\infty} \mathrm{Fib}(n) \cdot x^n} \right) - \mathrm{Fib}(0) \cdot x^0 \right] \quad + \\ & x^2 \cdot \boxed{\sum_{n=0}^{\infty} \mathrm{Fib}(n) \cdot x^n} \end{aligned}$$

$$\boxed{\sum_{n=0}^{\infty} \mathrm{Fib}(n) \cdot x^n \quad = \quad F(x)}$$

Jetzt können wir die **Summen eliminieren** und erhalten

$$F(x) \quad = \quad x + x \cdot F(x) + x^2 \cdot F(x)$$

- **Schritt 4**: **Löse die Gleichung nach $F(x)$ auf**:

$$F(x) = x + x \cdot F(x) + x^2 \cdot F(x)$$

$$\Leftrightarrow \quad 1 = \frac{x}{F(x)} + x + x^2$$

Keine Rekursion, keine Summen!

$$\Leftrightarrow \quad F(x) = \frac{x}{1 - x - x^2} = \sum_{n=0}^{\infty} \mathrm{Fib}(n) \cdot x^n$$

- **Schritt 5**: **Ziel**: drücke rechte Seite als Summe von zwei Brüchen aus:

$$F(x) = \frac{A}{1 - \alpha \cdot x} + \frac{B}{1 - \beta \cdot x}$$

A, *B*, *α*, *β* sind **Konstanten**!

$$\frac{1}{1 - \gamma \cdot x} = \sum_{n=0}^{\infty} \gamma^n \cdot x^n$$

Dann können wir übergehen zu

$$F(x) = A \cdot \sum_{n=0}^{\infty} \alpha^n \cdot x^n + B \cdot \sum_{n=0}^{\infty} \beta^n \cdot x^n$$

$$= \sum_{n=0}^{\infty} (A \cdot \alpha^n + B \cdot \beta^n) \cdot x^n$$

= Fib(*n*) !!

Lösen von rek. Gl. mit Erzeugenden Funktionen 5

Gelingt es uns $F(x)$ in die Form

$$\frac{x}{(1-\alpha\cdot x)\cdot(1-\beta\cdot x)}$$

Beachte: $\frac{1}{\alpha}$ und $\frac{1}{\beta}$ sind Nullstellen des Polynoms

$$(1-\alpha\cdot x)\cdot(1-\beta\cdot x)$$

zu überführen, gelangen wir mittels **Partialbruchzerlegung** zu

$$\frac{A}{1-\alpha\cdot x}+\frac{B}{1-\beta\cdot x}$$

Wegen

$$\frac{A}{1-\alpha\cdot x}=A\cdot\sum_{n=0}^{\infty}\alpha^n x^n \quad \text{und} \quad \frac{B}{1-\beta\cdot x}=B\cdot\sum_{n=0}^{\infty}\beta^n x^n$$

könne wir zu einer **Darstellung als Reihe** zurückkehren:

$$\begin{aligned} F(x) &= A\cdot\sum_{n=0}^{\infty}\alpha^n x^n + B\cdot\sum_{n=0}^{\infty}\beta^n x^n \\ &= \sum_{n=0}^{\infty}\underbrace{(A\cdot\alpha^n + B\cdot\beta^n)}_{=\text{Fib}(n)}\cdot x^n \end{aligned}$$

***A*, *B*, *α*, *β* sind Konstanten!**

Bestimmung der Kehrwerte der Nullstellen

- Bestimme Kehrwerte der Nullstellen des Nennerpolynoms $1\text{-}x\text{-}x^2$
  wir setzen $1\text{-}x\text{-}x^2=0$ und erhalten $x^2+\text{x}-1=0$

$$
\begin{aligned}
\frac{1}{\alpha}\Big/\frac{1}{\beta} &= -\frac{p}{2} \pm \sqrt{\left(\frac{p}{2}\right)^2 - q} \\
&= -\frac{1}{2} \pm \sqrt{\left(\frac{1}{2}\right)^2 + 1} \\
&= -\frac{1}{2} \pm \sqrt{\frac{1}{4} + 1} \\
&= -\frac{1}{2} \pm \sqrt{\frac{5}{4}} \\
&= -\frac{1}{2} \pm \frac{\sqrt{5}}{\sqrt{4}} \\
&= -\frac{1}{2} \pm \frac{\sqrt{5}}{2} \\
&= \frac{-1 \pm \sqrt{5}}{2}
\end{aligned}
$$

nun ist
$$
\begin{aligned}
\frac{1}{\frac{-1-\sqrt{5}}{2}} &= \frac{2}{-1-\sqrt{5}} \\
&= \frac{2\cdot(1-\sqrt{5})}{(-1-\sqrt{5})\cdot(1-\sqrt{5})} \\
&= \frac{2\cdot(1-\sqrt{5})}{-1+\sqrt{5}-\sqrt{5}+5} \\
&= \frac{2\cdot(1-\sqrt{5}}{4} \\
&= \frac{1-\sqrt{5}}{2}
\end{aligned}
$$

und analog:
$$
\frac{1}{\frac{-1+\sqrt{5}}{2}} = \frac{1+\sqrt{5}}{2}
$$

das **reflexierte Polynom** spart u.U. Rechenaufwand!

Einschub: Reflexiertes Polynom 1

- Reflexierte Polynome erleichtern das Berechnen der Kehrwerte der Nullstellen
- Sei

$$Q(x) \quad = \quad q_0 + q_1 \cdot x + q_2 \cdot x^2 + \cdots + q_m \cdot x^m$$

  ein Polynom m-ten Grades

- Dann ist

$$x^k \mapsto x^{(m-k)}$$

$$Q^R(x) \quad = \quad q_0 \cdot x^m + q_1 \cdot x^{m-1} + q_2 \cdot x^{m-2} + \cdots + q_m$$

  das **reflexierte Polynom** zu $Q(x)$

- es gilt:

$$\begin{aligned} & Q^R(x) &=& \quad q_0 \cdot (x - \rho_1)(x - \rho_m) \cdots (x - \rho_m) \\ \Leftrightarrow \quad & Q(x) &=& \quad q_0 \cdot (1 - \rho_1 \cdot x)(1 - \rho_2 \cdot x) \cdots (1 - \rho_m x) \end{aligned}$$

- d.h. die Nullstellen von $Q^R(x)$ entsprechen den Kehrwerten der Nullstellen von $Q(x)$!

Einschub: Reflexiertes Polynom 2

Sei $q(x) = 1-x-x^2$, so heisst $q^R(x) = x^2-x-1$ das **reflexierte Polynom**. Wir behaupten:

$$q^R(x) = (x-\alpha)(x-\beta) \Rightarrow q(x) = (1-\alpha\cdot x)(1-\beta\cdot x)$$

Mit anderen Worten, α und β sind genau die Nullstellen des reflexierten Polynoms.
Beweis: Offenbar gilt

$$\begin{aligned} x^2 q^R\left(\frac{1}{x}\right) &= x^2 \cdot \left(\frac{1}{x^2} - \frac{1}{x} - 1\right) \\ &= 1 - x - x^2 \\ &= q(x) \end{aligned}$$

Seien nun α und β die Nullstellen von $q^R(x)$; also $q^R(x) = (x-\alpha)(x-\beta)$ dann folgt mit $x^2 q^R(x) = q\left(\frac{1}{x}\right)$:

$$\begin{aligned} q\left(\frac{1}{x}\right) = x^2 q^R\left(\frac{1}{x}\right) &= x^2\left(\frac{1}{x}-\alpha\right)\left(\frac{1}{x}-\beta\right) \\ &= x\left(1-\alpha\cdot x\right)\left(\frac{1}{x}-\beta\right) \\ &= \left(1-\alpha\cdot x\right)\left(1-\beta\cdot x\right) \end{aligned}$$

Lösen von rek. Gl. mit Erzeugenden Funktionen 5

- Reflexiertes Polynom zu $1\text{-}x\text{-}x^2$ ist $x^2\text{-}x\text{-}1$
- Wir berechnen die Nullstellen mit der *p*/*q*-Formel:

$$
\begin{aligned}
\alpha/\beta &= -\frac{p}{2} \pm \sqrt{\left(\frac{p}{2}\right)^2 - q} \\
&= \frac{1}{2} \pm \sqrt{\left(\frac{1}{2}\right)^2 + 1} \\
&= \frac{1}{2} \pm \sqrt{\frac{1}{4} + 1} \\
&= \frac{1}{2} \pm \sqrt{\frac{5}{4}} \\
&= \frac{1}{2} \pm \frac{\sqrt{5}}{\sqrt{4}} \\
&= \frac{1}{2} \pm \frac{\sqrt{5}}{2} \\
&= \frac{1 \pm \sqrt{5}}{2}
\end{aligned}
$$

somit: $$F(x) = \frac{x}{\left(1 - \frac{1+\sqrt{5}}{2}x\right) \cdot \left(1 - \frac{1-\sqrt{5}}{2}x\right)}$$

Die Idee - *Ergebnis nach Bestimmung der Nullstellen*

Gelingt es uns $F(x)$ in die Form

Ja, reflexives Polynom + Satz von Vieta oder p/q-Formel

$$\frac{x}{(1-\alpha\cdot x)\cdot(1-\beta\cdot x)}$$

$$\boxed{\alpha = \frac{1+\sqrt{5}}{2} \quad \text{und} \quad \beta = \frac{1-\sqrt{5}}{2}}$$

zu überführen, gelangen wir mittels **Partialbruchzerlegung** zu

$$\frac{A}{1-\alpha\cdot x} + \frac{B}{1-\beta\cdot x}$$

Wegen

$$\frac{A}{1-\alpha\cdot x} = A\cdot\sum_{n=0}^{\infty}\alpha^n x^n \quad \text{und} \quad \frac{B}{1-\beta\cdot x} = B\cdot\sum_{n=0}^{\infty}\beta^n x^n$$

könne wir zu einer **Darstellung als Reihe** zurückkehren:

$$\begin{aligned} F(x) &= A\cdot\sum_{n=0}^{\infty}\alpha^n x^n + B\cdot\sum_{n=0}^{\infty}\beta^n x^n \\ &= \sum_{n=0}^{\infty}\underbrace{(A\cdot\alpha^n + B\cdot\beta^n)}_{=\text{Fib}(n)}\cdot x^n \end{aligned}$$

A, B, α, β sind Konstanten!

- **Zur Erinnerung:**

S Partialbruchzerlegung

Seien f und g Polynome über den komplexen Zahlen mit $\text{grad}(g) < \text{grad}(f)$ und

$$f(x) = (1 - \alpha_1 x)^{m_1} \cdot \cdots (1 - \alpha_r x)^{m_r}$$

mit $\alpha_i \in \mathbb{C}$ und $m_i \in \mathbb{N}$ $(1 \leq i \leq m)$. Dann gibt es Polynome $g_i(x)$ über den komplexen Zahlen mit $\text{grad}(g_i) < m_i$, so dass

$$\frac{g(x)}{f(x)} = \frac{g_1(x)}{(1 - \alpha_1 x)^{m_1}} + \cdots + \frac{g_r(x)}{(1 - \alpha_r x)^{m_r}}$$

Ferner gilt: Sofern $\alpha_i \in \mathbb{R}$ für alle $1 \leq i \leq r$ und f und g nur reellwerige Koeffizienten haben, so sind auch die Koeffizienten der Polynome $g_i(x)$ alle reelle Zahlen.

B Beispiele zur Partialbruchzerlegung

$$\frac{f(x)}{g(x)} = \frac{1+x^2}{(1-x)^2 \cdot (1-3x)}$$

wir starten mit folgendem Ansatz:

$$\frac{1+x^2}{(1-x)^2 \cdot (1-3x)} = \frac{ax+b}{(1-x)^2} + \frac{c}{(1-3x)}$$

Der Grad der Zählerpolynome $g_i(x)$ muss ja kleiner sein als der der Nennerpolynome. Wir bringen beide Summanden auf einen Hauptnenner:

$$\begin{aligned}\frac{1+x^2}{(1-x)^2 \cdot (1-3x)} &= \frac{(ax+b)\cdot(1-3x)+c\cdot(1-x)^2}{(1-x)^2 \cdot (1-3x)} \\ &= \frac{(a-3b-2c)\cdot x + (-3a+c)\cdot x^2 + (b+c)}{(1-x)^2 \cdot (1-3x)}\end{aligned}$$

Koeffizientenvergleich liefert drei Gleichungen :

$$-3a+c=1 \quad a-3b-2c=0 \quad b+c=1$$

Lösung dieses Gleichungssystems liefert $a=\frac{1}{2}$, $b=-\frac{3}{2}$ und $c=\frac{5}{2}$; wir erhalten:

$$\frac{1+x^2}{(1-x)^2 \cdot (1-3x)} = \frac{\frac{1}{2}x-\frac{3}{2}}{(1-x)^2} + \frac{\frac{5}{2}}{(1-3x)}$$

- **Es folgt die Partialbruchzerlegung für**

$$F(x) = \frac{x}{(1-\alpha x)\cdot(1-\beta x)} \quad \text{mit } \alpha = \frac{1+\sqrt{5}}{2} \text{ und } \beta = \frac{1-\sqrt{5}}{2}$$

Grad von Zählerpolynom ist **kleiner** als der des Nennerpolynoms
daher können wir wie folgt ansetzen:

$$\begin{aligned} F(x) = \frac{\boxed{1\cdot x}+\boxed{0}}{(1-\alpha x)\cdot(1-\beta x)} &= \frac{A}{1-\alpha x} + \frac{B}{1-\beta x} \\ &= \frac{A\,(1-\beta x) + B\,(1-\alpha x)}{(1-\alpha x)\,(1-\beta x)} \\ &= \frac{A - A\beta x + B - B\alpha x}{(1-\alpha x)\,(1-\beta x)} \\ &= \frac{\boxed{(A+B)}\boxed{-x\cdot(A\beta + B\alpha)}}{(1-\alpha x)\,(1-\beta x)} \end{aligned}$$

Koeffizientenvergleich liefert folgendes Gleichungssystem:

$$\left| \begin{array}{rcr} A+B & = & 0 \\ A\beta + B\alpha & = & -1 \end{array} \right|$$

Lösung des Gleichungssystems

$$\left|\begin{array}{rcl} A+B &=& 0 \\ A\beta+B\alpha &=& -1 \end{array}\right|$$

$$\Leftrightarrow \left|\begin{array}{rcl} B\beta-B\alpha &=& 1 \\ A\beta+B\alpha &=& -1 \end{array}\right|$$

$$\Leftrightarrow \left|\begin{array}{rcl} B\cdot(\beta-\alpha) &=& 1 \\ A\beta+B\alpha &=& -1 \end{array}\right|$$

$$\Leftrightarrow \left|\begin{array}{rcl} B &=& \frac{1}{\beta-\alpha} \\ A\beta+B\alpha &=& -1 \end{array}\right|$$

$$\Leftrightarrow \left|\begin{array}{rcl} B &=& -\frac{1}{\sqrt{5}} \\ A\beta+B\alpha &=& 1 \end{array}\right|$$

$$\Leftrightarrow \left|\begin{array}{rcl} B &=& -\frac{1}{\sqrt{5}} \\ A\beta-\frac{1}{\sqrt{5}}\alpha &=& 1 \end{array}\right|$$

$$\Leftrightarrow \left|\begin{array}{rcl} A &=& -\frac{1}{\sqrt{5}} \\ B &=& \frac{-1+\frac{1}{\sqrt{5}}\alpha}{\beta} \end{array}\right|$$

$$\Leftrightarrow \left|\begin{array}{rcl} B &=& -\frac{1}{\sqrt{5}} \\ A &=& \frac{-1+\frac{1}{\sqrt{5}}\cdot\frac{1+\sqrt{5}}{2}}{\frac{1-\sqrt{5}}{2}} \end{array}\right|$$

$$\Leftrightarrow \left|\begin{array}{rcl} B &=& -\frac{1}{\sqrt{5}} \\ A &=& \frac{\frac{-2\cdot\sqrt{5}+1+\sqrt{5}}{2\cdot\sqrt{5}}}{\frac{1-\sqrt{5}}{2}} \end{array}\right|$$

$$\Leftrightarrow \left|\begin{array}{rcl} B &=& -\frac{1}{\sqrt{5}} \\ A &=& \frac{\frac{1-\sqrt{5}}{2\cdot\sqrt{5}}}{\frac{1-\sqrt{5}}{2}} \end{array}\right|$$

$$\Leftrightarrow \left|\begin{array}{rcl} B &=& -\frac{1}{\sqrt{5}} \\ A &=& \frac{2\cdot(1-\sqrt{5})}{2\cdot\sqrt{5}\cdot(1-\sqrt{5})} \end{array}\right|$$

$$\Leftrightarrow \boxed{\left|\begin{array}{rcl} B &=& -\frac{1}{\sqrt{5}} \\ A &=& \frac{1}{\sqrt{5}} \end{array}\right|}$$

Alternativ zum Lösen des Gleichungssystems (1)

- **Einsetzen der Nullstellen:**

$$F(x) = \frac{x}{(1-\alpha x)\cdot(1-\beta x)} \quad = \quad \frac{A\,(1-\beta x) + B\,(1-\alpha x)}{(1-\alpha x)\cdot(1-\beta x)}$$

Einsetzen der Nullstelle $\frac{1}{\beta}$ führt zu

$$F\left(\frac{1}{\beta}\right) = \frac{\frac{1}{\beta}}{\left(1-\alpha\frac{1}{\beta}\right)\cdot\left(1-\beta\frac{1}{\beta}\right)} \quad = \quad \frac{A\cdot\left(1-\beta\frac{1}{\beta}\right) + B\cdot\left(1-\alpha\frac{1}{\beta}\right)}{\left(1-\alpha\frac{1}{\beta}\right)\cdot\left(1-\beta\frac{1}{\beta}\right)}$$

$$= \quad \frac{B\cdot\left(1-\alpha\frac{1}{\beta}\right)}{\left(1-\alpha\frac{1}{\beta}\right)\cdot\left(1-\beta\frac{1}{\beta}\right)}$$

Also muss gelten

$$\boxed{\frac{1}{\beta} = B\cdot\left(1-\alpha\frac{1}{\beta}\right)}$$

Alternativ zum Lösen des Gleichungssystems (2)

- **Auflösen nach B:**

$$
\begin{aligned}
B &= \frac{1}{\beta \cdot \left(1 - \alpha \cdot \frac{1}{\beta}\right)} \\
&= \frac{1}{\beta - \alpha} \\
&= \frac{1}{\frac{1-\sqrt{5}}{2} - \frac{1+\sqrt{5}}{2}} \\
&= \frac{1}{\frac{-2\sqrt{5}}{2}} \\
&= -\frac{1}{\sqrt{5}}
\end{aligned}
$$

$$\boxed{\alpha = \frac{1+\sqrt{5}}{2} \quad \text{und} \quad \beta = \frac{1-\sqrt{5}}{2}}$$

- **Analog verfährt man mit der anderen Nullstelle um A zu erhalten**

Die Idee - *Ergebnis nach Partialbruchzerlegung*

Gelingt es uns $F(x)$ in die Form

Ja, reflexives Polynom + Satz von Vieta oder p/q-Formel

$$\frac{x}{(1-\alpha\cdot x)\cdot(1-\beta\cdot x)}$$

$$\boxed{\alpha = \frac{1+\sqrt{5}}{2} \quad \text{und} \quad \beta = \frac{1-\sqrt{5}}{2}}$$

zu überführen, gelangen wir mittels **Partialbruchzerlegung** zu

$$\frac{A}{1-\alpha\cdot x} + \frac{B}{1-\beta\cdot x}$$

$$\boxed{A = \frac{1}{\sqrt{5}} \quad \text{und} \quad B = -\frac{1}{\sqrt{5}}}$$

Wegen

$$\frac{A}{1-\alpha\cdot x} = A\cdot\sum_{n=0}^{\infty}\alpha^n x^n \quad \text{und} \quad \frac{B}{1-\beta\cdot x} = B\cdot\sum_{n=0}^{\infty}\beta^n x^n$$

könne wir zu einer **Darstellung als Reihe** zurückkehren:

$$\begin{aligned} F(x) &= A\cdot\sum_{n=0}^{\infty}\alpha^n x^n + B\cdot\sum_{n=0}^{\infty}\beta^n x^n \\ &= \sum_{n=0}^{\infty}\underbrace{(A\cdot\alpha^n + B\cdot\beta^n)}_{=\text{Fib}(n)}\cdot x^n \end{aligned}$$

A, B, α, β sind Konstanten!

$$\boxed{\text{Fib}(n) = \frac{1}{\sqrt{5}}\cdot[\alpha^n - \beta^n]}$$

Die Idee - *Ergebnis nach Partialbruchzerlegung*

Gelingt es uns $F(x)$ in die Form

Ja, reflexives Polynom + Satz von Vieta oder p/q-Formel

$$\frac{x}{(1-\alpha\cdot x)\cdot(1-\beta\cdot x)}$$

$$\boxed{\alpha = \frac{1+\sqrt{5}}{2} \quad \text{und} \quad \beta = \frac{1-\sqrt{5}}{2}}$$

zu überführen, gelangen wir mittels **Partialbruchzerlegung** zu

$$\frac{A}{1-\alpha\cdot x} + \frac{B}{1-\beta\cdot x}$$

$$\boxed{A = \frac{1}{\sqrt{5}} \quad \text{und} \quad B = -\frac{1}{\sqrt{5}}}$$

Wegen

$$\frac{A}{1-\alpha \cdot x}$$

könne wir zu eine

Insgesamt erhalten wir

$$\mathrm{Fib}(n) = \frac{1}{\sqrt{5}} \cdot \left[\left(\frac{1+\sqrt{5}}{2}\right)^n - \left(\frac{1-\sqrt{5}}{2}\right)^n\right]$$

$$\begin{aligned} F(x) &= A\cdot\sum_{n=0}^{\infty}\alpha^n x^n + B\cdot\sum_{n=0}^{\infty}\beta^n x^n \\ &= \sum_{n=0}^{\infty}\underbrace{(A\cdot\alpha^n + B\cdot\beta^n)}_{=\mathrm{Fib}(n)}\cdot x^n \end{aligned}$$

$$\boxed{\mathrm{Fib}(n) = \frac{1}{\sqrt{5}}\cdot[\alpha^n - \beta^n]}$$