<u>Das HORNER-Schema</u>

Ac 11-84 bis 05-2022

Das Horner-Schema ist benannt nach dem englischen Mathematiker William George Horner (1786 bis 1837). Es bietet Rechenvorteile bei der Bearbeitung von Polynomen $p(x)$.
Mittels des sogenannten <u>einfachen Horner-Schemas</u> kann man u.a. Funktionswerte $p(x_0)$ berechnen.

Beispiel: $p(x) = 2x^4 - 8x^3 - 2x^2 + 32x - 24$.

Das Schema nutzt hierbei eine Faktorisierung von p nach folgendem Muster aus:

$p(x) = (((2 \cdot x - 8) \cdot x - 2) \cdot x + 32) \cdot x - 24$

Der erhaltene Term wird von „innen“ nach „außen“ abgearbeitet .
Während man beim ursprünglichen Polynom 10 Multiplikationen und 4 „Strichrechnungen“ benötigt, sind es beim Horner-Schema nur 4 Multiplikationen und 4 „Strichrechnungen“ .
Bei einem Polynom vom Grad n benötigt das Horner-Schema n Multiplikationen sowie n Additionen.

Obige Faktorisierung lässt sich schematisieren (Einfaches Horner-Schema) :
Vorgegeben sind die Koeffizienten von $p(x)$ sowie die Stelle x_0 (hier = 2 gewählt).
In der untersten Zeile stehen x_0 sowie die Summen der Zeilen 2 und 4 .

Einfaches Horner-Schema:

	x^4	x^3	x^2	x^1	x^0	Potenzen von p(x)
	2	-8	-2	32	-24	Koeffizienten von p(x)
		+	+	+	+	
$x_0 = 2$	$\downarrow$	4	-8	-20	24	
		$\cdot 2 \nearrow$	$\cdot 2 \nearrow$	$\cdot 2 \nearrow$	$\cdot 2 \nearrow$	
	2	-4	-10	12	$0 = p(2)$	

Man erkennt hier, dass 2 eine Nullstelle ist, da $p(2) = 0$!

Außerdem kann man (mit Weitblick) erkennen, dass in der untersten Zeile die Koeffizienten des Zerlegungspolynoms $h(x) := p(x) / (x-2)$ stehen (grüne Zahlen) !

Dies sei anhand einer Polynomdivision gezeigt:

$h(x) = p(x) / (x-x_0) = (2x^4 - 8x^3 - 2x^2 + 32x - 24) / (x-2) = 2x^3 - 4x^2 - 10x + 12$

```
 2x⁴ – 4x³
 ------------
     - 4x³  - 2x²
     - 4x³ + 8x²
     --------------
            - 10x² + 32x
            - 10x² + 20x
            ----------------
                       12x -24
                       12x -24
                       ----------
                             0
```

Man erhält das Zerlegungspolynom $h(x) = 2x^3 - 4x^2 - 10x + 12$.

Bei der Polynomdivision benötigt man n Divisionen, (2n) Multiplikationen, (2n) Subtraktionen sowie viel Schreibarbeit. Für das obige Beispiel hat man also 20 Grundrechenoperationen durchzuführen.

Wesentlich einfacher findet man, wie oben gezeigt, das Zerlegungspolynom mit Hilfe des Horner-Schemas, bei dem man nur n Multiplikationen sowie n Additionen benötigt .

Zusätzlich liefert das Schema noch den Funktionswert $p(x_0)$.

Vorteile des Horner-Schemas sind seine universelle Einsetzbarkeit, z.B. zum Berechnen von

- **Funktionswerten**,
- **Zerlegungspolynomen**,
- **Basisumwandlungen** allg. Basis $g \neq 10$ wird umgewandelt in das 10er - System,
- **Ableitungswerten** p', p'', p''', etc.
- **Entwicklungspolynomen** **$p(x-x_0)$** ; Entwicklung von p an einer beliebigen Stelle x_0 ;

Ein weiteres Anwendungsbeispiel:

Wie wandelt man **Dualzahlen** mittels des Horner-Schemas in das **Zehnersystem** um ?

Beispiel: $11010011_2 = ?_{10}$

Es ist $11010011_2 = 1 \cdot 2^7 + 1 \cdot 2^6 + 0 \cdot 2^5 + 1 \cdot 2^4 + 0 \cdot 2^3 + 0 \cdot 2^2 + 1 \cdot 2^1 + 1 \cdot 2^0$

x_0 ist hier gleich 2 und die 2er-Potenzen werden der Reihe nach aufgeschrieben

2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
1	1	0	1	0	0	1	1
+	+	+	+	+	+	+	+
0	2	6	12	26	52	104	210
1	3	6	13	26	52	105	**211 = dezimales Ergebnis**

Anmerkungen:

1) Auf die gleiche Art und Weise kann man auch Zahlen aus anderen Basissystemen (z.B. Hexadezimalsystem) in das **Zehnersystem** umwandeln.

2) Der umgekehrte Fall, nämlich die Umwandlung vom Zehnersystem in eine andere Basis lässt sich nicht mit dem Horner-Schema bewältigen, weil dann Divisionen durchgeführt werden müssen:

Beispiel: Die Zahl 37 soll vom Zehnersystem ins Zweiersystem umgewandelt werden.

Man muss nun fortwährend durch die Basis 2 dividieren und den Rest notieren:

37 / 2 = 18 Rest **1**.
18 / 2 = 9 Rest **0**.
9 / 2 = 4 Rest **1**.
4 / 2 = 2 Rest **0**.
2 / 2 = 1 Rest **0**.
1 / 2 = ***0*** Rest **1**.

Beim Ergebnis 0 endet der Algorithmus.
Die Reste (oberster Wert ganz rechts notiert) sind **100101**.
Dies ist die Dualzahldarstellung von **37**, wie man leicht nachprüfen kann.

Mit dem Horner-Schema kann man auch Funktionswerte p(x) von komplexen Zahlen
$x = a + b\cdot i$ (a,b reell) berechnen !

Beispiel: $p(x) = 6x^5 + 11x^4 - 33x^3 - 33x^2 + 11x + 6$ und $x = 2 + i$

Einfaches Horner-Schema:

	x^5	x^4	x^3	x^2	x^1	x^0	Potenzen von p(x)
	6	11	-33	-33	11	6	Koeffizienten von p(x)
		+	+	+	+	+	
$x_0 = 2+i$	$\downarrow$	$12+6i$	$40+35i$	$-21+77i$	-185+100i	$-448+26i$	
	$\cdot(2+i)\nearrow$	$\cdot x_0 \nearrow$	$\cdot x_0 \nearrow$	$\cdot x_0 \nearrow$	$\cdot x_0 \nearrow$		
	6	23+6i	$7+35i$	$-54+77i$	$-174+100i$	$-442+26i = p(2+i)$	

Der Funktionswert ist also -442 + 26i .

Selbstverständlich kann man mit dem einfachen Horner-Schema auch Funktionswerte von komplexen Polynomen (Polynomen mit komplexen Koeffizienten) bestimmen .

Weiter unten werden wir sehen, dass komplexe Argumente auch mit dem so genannten zweizeiligen Horner-Schema bestimmt werden können.

Das vollständige Horner-Schema

Mittels des vollständigen Horner-Schemas lassen sich alle **Ableitungen** eines Polynoms berechnen. Außerdem erhält man automatisch die Koeffizienten A_k des **Entwicklungspolynoms** $p(x\text{-}x_0)$, d.h. p wird entwickelt an der Stelle x_0 .

Vollständiges Horner-Schema für $p(x) = x^4 - 6x^3 + 11x^2 - 6x$; Entwicklungsstelle $x_0 = -3$.

Vollständiges Horner-Schema:

	x^4	x^3	x^2	x^1	x^0	Potenzen von p(x)
	1	-6	11	-6	0	Koeffizienten von p(x)
		$+$	$+$	$+$	$+$	
	$\downarrow$	-3	27	-114	360	
	$\cdot -3 \nearrow$	$\cdot -3 \nearrow$	$\cdot -3 \nearrow$	$\cdot -3 \nearrow$		
$x_0 = -3$	1	-9	38	-120	$360 = A_0 = p(-3)$	
		$+$	$+$	$+$		
	$\downarrow$	-3	36	-222		
	$\cdot -3 \nearrow$	$\cdot -3 \nearrow$	$\cdot -3 \nearrow$			
$x_0 = -3$	1	-12	74	$-342 = A_1 = p'(-3)$		
		$+$	$+$			
	$\downarrow$	-3	45			
	$\cdot -3 \nearrow$	$\cdot -3 \nearrow$				
$x_0 = -3$	1	-15	$119 = A_2 = p''(-3)/2!$			
		$+$				
	$\downarrow$	-3				
	$\cdot -3 \nearrow$					
$x_0 = -3$	1	$-18 = A_3 = p'''(-3)/3!$				
	$\downarrow$					
$x_0 = -3$	$1 = A_4 = p^{(4)}(-3)/4!$					

Zunächst sieht man, dass sich die Ableitungen an der Stelle -3 leicht ablesen lassen.
Z.B. ergibt sich $p^{(4)}(-3) = 1 \cdot 4! = 24$. Entsprechend $p''(-3) = 119 \cdot 2! = 238$

A_0, A_1, A_2, A_3, A_4 sind die Koeffizienten des Entwicklungspolynoms, so dass sich folgendes ergibt :

$$p(x-(-3)) = p(x+3) = (x+3)^4 - 18\cdot(x+3)^3 + 119\cdot(x+3)^2 - 342\cdot(x+3) + 360$$

bzw. mit $z = x+3$:

$$p(z) = z^4 - 18\cdot z^3 + 119\cdot z^2 - 342\cdot z + 360$$

Horner-Schema programmiert in JAVA:

```java
public static final double EPS = 3.55271368E-15; // 2^(-48)

   public static boolean istNull(double x) {
       return Math.abs(x) < EPS;
   }
   public static boolean istGanzzahlig(double x) {
       if (istNull(x)) return true;
       return istNull((x - Math.rint(x)) / x);
   }
   public static double rundeGanzzahlig(double x) {
      // für Zahlen, die ganz dicht an einer ganzen liegen
       if (istGanzzahlig(x))
             return Math.rint(x);
       return x; // andernfalls
   }

  public static double fPolynom(double[] a, double x0) {
      // berechnet den Funktionswert f(x) des Polynoms
      // p(x) = a0 + a1x + a2x² + ... + a(n-1)x^(n-1)+ a(n)x^n
      int n = a.length - 1; // n = Grad des Polynoms
      double erg = a[n];
      for (int i = n - 1; i >= 0; i--)
         erg = erg * x0 + a[i];

      return erg;
   }

   public static double[] deflationLin(double[] a, double x0) {
       // berechnet die Koeffizienten des Zerlegungspolynoms  h(x) = f(x) / (x-x0) + Rest
       // Methode: HORNER-Schema
       int n = a.length - 1; // n = Grad des Polynoms a
       double[] zerlegPoly = new double[n]; // n-1 = Grad des Zerlegungspolynoms
       zerlegPoly[n-1] = a[n];
       for (int i = n - 1; i > 0; i--)
             zerlegPoly[i-1] = rundeGanzzahlig(zerlegPoly[i] * x0 + a[i]);

       double rest = rundeGanzzahlig(zerlegPoly[0] * x0 + a[0]);

       if (!istNull(rest/8)) { // f(x0) ≠ 0 ?  dann Rest auch ausgeben
             double[] zerlegPolyMitRest = new double[n]; // Feld um 1 Komponente erweitern
             zerlegPolyMitRest[0] = rest;
             for (int i = n - 1; i >= 0; i--)
                   zerlegPolyMitRest[i+1] = zerlegPoly[i];
             return zerlegPolyMitRest;
       }

       return zerlegPoly;
   }

   public static double[] hornerKomplett(double[] a, double x0) {
       // berechnet die Koeffizienten ai des vollständigen HORNER-Schemas
       // außerdem gilt für die Ableitungen:  f(i')(x0) = ai * i!
       int n = a.length - 1; // n = Grad des Polynoms a
       for (int j = 0; j <= n; j++) { // Index des Entwicklungspolynoms-Koeffizienten
             for (int i = n-1; i >= j; i--)
                   a[i] = a[i+1] * x0 + a[i];
             if (Math.abs(a[j]) < 1E-15) // a[j] = 0 ?
                   a[j] = 0.0;
       }

       return a;
   }
```

Das zweizeilige Horner-Schema

Will man ein Polynom durch eine quadratische Funktion dividieren, kann man das mit Polynomdivision machen, oder aber mit dem zweizeiligen Horner-Schema.

Beispiel: $p(x) = 6x^5 + 11x^4 - 33x^3 - 33x^2 + 11x + 6$.

Dividiert werden soll durch $x^2 - x - 2$

Man erhält das Zerlegungspolynom $z(x) = 6x^3 + 17x^2 - 4x - 3$.

Allgemeine Formulierung:

Das Polynom $p_n(x) = a_n x^n + a_{n-1}x^{n-1} + \ldots + a_2x^2 + a_1x + a_0$
soll durch den quadratische Term $x^2 + p \cdot x + q$ dividiert werden.

Dabei entsteht das Zerlegungspolynom (bzw. "Deflationspolynom")

$$Z_{n-2}(x) = b_{n-2}x^{n-2} + b_{n-3}x^{n-3} + \ldots + b_1x + b_0$$

Die Koeffizienten b_k des Zerlegungspolynoms Z berechnen sich wie folgt:

$b_{n-2} = a_n$

$b_{n-3} = a_{n-1} - p \cdot b_{n-2}$

Für k von n - 4 ab bis 0

$b_k = a_{k+2} - p \cdot b_{k+1} - q \cdot b_{k+2}$

Geht die Division nicht ohne Rest auf, so kann man zusätzlich noch den

Restterm $(r \cdot x + t) / (x^2 + p \cdot x + q)$ bestimmen mit :

$r = a_1 - p \cdot b_0 - q \cdot b_1$

$t = a_0 - q \cdot b_0$

Im Horner-Schema sieht das so aus (Beispiel von oben):
Die vorgegebenen Zahlen sind blau markiert, die anderen werden berechnet !
Die Werte von -q und -p werden mit den (rot markierten) Ergebnissen multipliziert .
Das jeweilige Produkt wird (für -p) in die p-Zeile geschrieben, und zwar um **1** Spalte nach rechts versetzt, während es für -q in der q-Zeile um **2** Spalten nach rechts versetzt geschrieben wird !

Zweizeiliges Horner-Schema:

	x^5	x^4	x^3	x^2	x^1	x^0	Potenzen von p(x)
	6	11	-33	-33	11	6	Koeffizienten von p(x)
		$+$	$+$	$+$	$+$		
$-q=2$	$\downarrow$	0	12	34	-8	-6	
		$\cdot 2 \nearrow$	$\cdot 2 \nearrow$	$\cdot 2 \nearrow$	$\cdot 2 \nearrow$		
$-p=1$	$\downarrow$	6	17	-4	-3		
	$\cdot 1 \nearrow$	$\cdot 1 \nearrow$	$\cdot 1 \nearrow$	$\cdot 1 \nearrow$			
	6	17	-4	-3	0	0	

Eventuelle Restterme (braun) werden hier automatisch mitberechnet !

Wie sieht das Schema aus, wenn bei der Division ein Rest bleibt ?

Dazu betrachten wir die Division des Polynoms durch $x^2 - x - 1$, also $-p = 1$ $-q = 1$.

Zweizeiliges Horner-Schema:

	x^5	x^4	x^3	x^2	x^1	x^0	Potenzen von p(x)
	6	11	-33	-33	11	6	Koeffizienten von p(x)
		+	+	+	+	+	
$-q=1$	$\downarrow$	0	6	17	-10	-26	
		$\cdot 1 \nearrow$	$\cdot 1 \nearrow$	$\cdot 1 \nearrow$	$\cdot 1 \nearrow$		
$-p=1$	$\downarrow$	6	17	-10	-26		
	$\cdot 1 \nearrow$	$\cdot 1 \nearrow$	$\cdot 1 \nearrow$	$\cdot 1 \nearrow$			
	6	17	-10	-26	-25	-20	

Ergebnis:

$6x^5 + 11x^4 - 33x^3 - 33x^2 + 11x + 6 \;/\; (x^2 - x - 1) = 6x^3 + 17x^2 -10x - 26$ Rest $-25x - 20$.

Interessanter Weise kann man mit dem zweizeiligen Horner-Schema auch Funktionswerte p(z) von komplexen Zahlen $z = a + b \cdot i$ (a,b reell) berechnen !

Voraussetzung: Das Polynom p muss ein reelles sein, d.h. alle Koeffizienten müssen reell sein.

Bei einer komplexen Zahl z ist das Produkt aus z und seiner konjugiert komplexen Zahl $\overline{z}$ reell .
$z \cdot \overline{z} = (a + b \cdot i) \cdot (a - b \cdot i) = a^2 - (b \cdot i)^2 = a^2 + b^2$

Ebenso ist die Linearfaktorzerlegung $(z - (a + b \cdot i)) \cdot (z - (a - b \cdot i))$ reellwertig, d.h. das beim Ausmultiplizieren entstehende Polynom vom Grad 2 hat reelle Koeffizienten .

$$\begin{aligned} (z - (a + b \cdot i)) \cdot (z - (a - b \cdot i)) &= z^2 - (a + b \cdot i) \cdot z - (a - b \cdot i) \cdot z + a^2 + b^2 \\ &= z^2 - 2a \cdot z + a^2 + b^2 \end{aligned}$$

Im zweizeiligen Horner-Schema kann man ein Polynom durch $(z^2 + p \cdot z + q)$ dividieren.
Durch Koeffizientenvergleich mit obigem z-Term erhalten wir $p = -2a$ sowie $q = a^2 + b^2$.

Beispiel von oben (nur mit dem komplexen Argument z statt x) :

$p(z) = 6z^5 + 11z^4 - 33z^3 - 33z^2 + 11z + 6$

Wir berechnen p(z) mit $z = 2 + i$, folglich gilt: $p = -4$ und $q = 5$.

Zweizeiliges Horner-Schema:

	z^5	z^4	z^3	z^2	z^1	z^0	Potenzen von p(z)
	6	11	-33	-33	11	6	Koeffizienten von p(z)
		+	+	+	+	+	
$-q=-5$	$\downarrow$	0	-30	-175	-385	-500	
		$\cdot -5 \nearrow$	$\cdot -5 \nearrow$	$\cdot -5 \nearrow$	$\cdot -5 \nearrow$		
$-p=4$	$\downarrow$	24	140	308	400		
	$\cdot 4 \nearrow$	$\cdot 4 \nearrow$	$\cdot 4 \nearrow$	$\cdot 4 \nearrow$			
	6	35	77	100	26	-494	

Wir erhalten den Restterm $26z - 494$.
Setzen wir für jetzt $z = 2 + i$ ein, so ergibt sich der Wert $p(2 + i) = -442 + 26 \cdot i$.

Das dreizeilige Horner-Schema

Äquivalent zum zweizeiligen Horner-Schema betrachten wir das dreizeilige Horner-Schema.

Das gegebene Polynom wird nun durch eine kubische Funktion mit x^3+bx^2+cx+d dividiert !

Beispiel: $p(x) = 2x^5 - 30x^4 + 170x^3 - 450x^2 + 548x - 240$.

Dividiert werden soll durch $x^3 - 7x^2 + 14x - 8$, also $b = -7$ $c = 14$ $d = -8$.

Man erhält das Zerlegungspolynom $z(x) = x^2 - x - 2$.

Für das Horner-Schema bedeutet das eine Verwaltung von 3 Zeilen, welche sich auf die Parameter b, c, d der kubischen Funktion beziehen.
Die vorgegebenen Zahlen sind wieder blau markiert, die anderen werden berechnet !
Die Werte von -d, -c und -b werden mit den (rot markierten) Ergebnissen multipliziert .
Das jeweilige Produkt wird für -b in die b-Zeile geschrieben, und zwar um **1** Spalte nach rechts versetzt, während es für -c in der c-Zeile um **2** Spalten nach rechts versetzt geschrieben wird .
Entsprechend schreiben wir das Produkt für -b in der b-Zeile um **3** Spalten nach rechts versetzt .

Dreizeiliges Horner-Schema:

	x^5	x^4	x^3	x^2	x^1	x^0	Potenzen von p(x)
	2	-30	170	-450	548	-240	Koeffizienten von p(x)
		$+$	$+$	$+$	$+$	$+$	
$-d=8$	$\downarrow$	0	0	16	-128	240	
			$\cdot 8 \nearrow$	$\cdot 8 \nearrow$	$\cdot 8 \nearrow$		
$-c=-14$	$\downarrow$	0	-28	224	-420		
		$\cdot -14 \nearrow$	$\cdot -14 \nearrow$	$\cdot -14 \nearrow$			
$-b=7$	$\downarrow$	14	-112	210			
	$\cdot 7 \nearrow$	$\cdot 7 \nearrow$	$\cdot 7 \nearrow$				
	2	-16	30	0	0	0	

Das Ergebnis ist $z(x) = 2x^2 - 16x + 30$.
Der Rest (braune Nullen unten rechts) ist gleich 0 .

Allgemein kann man k-zeilige Horner-Schemata bilden für die Division eines Polynom n-ten Grades durch ein Polynom k-ten Grades: $p_k(x) = x^k + a_{k-1}x^{k-1} + ... + a_2x^2 + a_1x + a_0$

Der Grad k darf natürlich nicht größer als n sein !

Java-Programm für die Division durch ein quadratisches Polynom:

```
public static double[] deflationQuadr(double[] poly, double r, double t) {
   // berechnet die Koeffizienten des Zerlegungspolynoms hn-2(x) = fn(x)/(x²+rx+t), n>2
    // Methode:  zweizeiliges HORNER-Schema
    int n = poly.length - 1; // n = Grad des Polynoms poly
    double[] hPol = new double[n-1]; // n-2 = Grad des Zerlegungspolynoms, also n-1 Koeff
    hPol[n-2] = poly[n];
    hPol[n-3] = rundeGanzzahlig(poly[n-1] - r*poly[n]);
    for (int i = n - 4; i >= 0; i--)
          hPol[i] = rundeGanzzahlig(poly[i+2] - r*hPol[i+1] -t*hPol[i+2]);

    double rest1 = rundeGanzzahlig(poly[1] - r*hPol[0] - t*hPol[1]);
    double rest0 = rundeGanzzahlig(poly[0] - t*hPol[0]);
    // System.out.println(rest1 + "    " + rest0);
    if (!istNull(rest1/16) || !istNull(rest0/16)) {
          // P(x0) ≠ 0 ? , dann die beiden Reste ebenfalls ausgeben
          double[] restPolyMitRest = new double[n+1]; // Feld um 2 Komponenten erweitern
          restPolyMitRest[0] = rest0;
          restPolyMitRest[1] = rest1;
          for (int i = n-2 ; i >= 0; i--)
                restPolyMitRest[i+2] = hPol[i];
          return restPolyMitRest;
    }

    return hPol;
}
```