GYMNASIUM OTTOBRUNN

Kollegstufe 1999/2001

Leistungskurs Mathematik

Facharbeit

Das Horner-Schema

Verfasser: **Markus Bauer**

Kursleiter: **Erich Lintner**

Bewertung: **13** Punkte

Unterschrift des Kursleiters: ____________________________

Inhaltsangabe

I. Einleitung ... 3

II. Geschichtlicher Hintergrund ... 4

III. Das Horner-Schema ... 5

1. Herleitung ... 5
2. Das allgemeine Horner-Schema: ... 6
3. Abspaltung eines Linearfaktors ... 7
4. Das erweiterte Horner-Schema ... 8

IV. Vorteile des Horner-Schemas ... 9

1. Bedeutung von Additionen und Multiplikationen ... 9
2. Berechnung nach der einfachsten Methode ... 9
3. Eine „intelligentere“ Methode ... 10
4. Berechnung nach dem Horner-Schema ... 11
5. Ergebnis der Untersuchung ... 12

V. Anwendung in der Informatik ... 13

1. Grundlagen ... 13
2. Struktogramm ... 13
3. Verwirklichung in Turbo-Pascal ... 14

VI. Anwendungsbeispiel ... 15

1. Das Newton-Verfahren ... 15
2. Das Beispiel ... 15

VII. Schluss ... 17

VIII. Quellen und Hilfsmittel ... 18

1. Literaturverzeichnis ... 18
2. Internetquellen ... 18
3. Verwendete Software ... 19

IX. Anhang ... 20

X. Erklärung ... 21

I. Einleitung

In der Analysis müssen häufig die Werte einer Funktion und die ihrer Ableitung berechnet werden. Sei es, um eine Nullstelle zu bestimmen, einen Überblick über eine Funktion zu erhalten, oder um einen Graphen zu skizzieren.
Bei *Polynomen* erfordert dabei das Einsetzen der x-Werte in die Funktionsgleichung und die Berechnung der Potenzen bei nicht einfachen Zahlen und vor allem bei großen Potenzen erhebliche Mühe. Es wird viel Zeit benötigt und die Fehlerwahrscheinlichkeit ist hoch. Dabei gibt es für Polynome schon lange ein sehr einfaches Verfahren: Das *Horner-Schema*.
Ziel dieser Arbeit ist es, das Prinzip des Horner-Schemas darzustellen und einige weiterführende Aspekte, vor allem im Bereich der Informatik, auszuführen. Nach einer allgemeinen Herleitung und Erklärung des Schemas gehe ich auch auf weitere Nebeneffekte, wie das *erweiterte* Horner-Schema, ein. Außerdem werden ausführlich die Vorteile des Horner-Schemas dargestellt.
In Kombination mit dem *Newton-Verfahren* lässt sich das Horner-Schema sehr effektiv einsetzten und man kann hierbei rasch, auch bei komplizierteren Polynomen, eine Nullstelle berechnen. Hierauf wird in Form eines Beispiels eingegangen.
Ein wichtiger Aspekt dieser Arbeit ist die Anwendung des Horner-Schemas in der *Informatik*. Es lässt sich hierbei eine sehr einfache und effektive Anwendung programmieren, die ich in *Turbo-Pascal* geschrieben habe. Diese Programme habe ich in der Anlage auf einer Diskette beigefügt, die mit einer Benutzeroberfläche die Wirkungsweise des Schemas darstellen. Ebenfalls beigefügt sind die *Quellcodes* der *Hauptprogramme* und die verwendeten *Units*.
Die für deren Verständnis notwendigen Grundlagen werden in der Arbeit dargelegt.
Am Anfang der Arbeit gebe ich zunächst eine kurze Darstellung des geschichtlichen Hintergrundes.

II. Geschichtlicher Hintergrund

William George Horner wurde 1786 in Bristol, England, geboren. Er besuchte die *Kingswood School Bristol*. Bereits im Alter von vierzehn wurde er im Jahre 1800 stellvertretender Direktor und vier Jahre später sogar Direktor seiner Schule. Er verließ Bristol 1809 und gründete seine eigene Schule in Bath.

Horners einziger bedeutsamer Beitrag zur Mathematik war das *Horner-Schema* zur Lösung algebraischer Gleichungen. Es wurde am 1. Juli 1819 der *Royal Society* vorgelegt und noch im selben Jahr in den *Philosophical Transactions of the Royal Society* veröffentlicht.[1]

Jedoch war Horner nicht der Erste, der diese Methode entdeckt hatte, da sie bereits 500 Jahre früher *Chu Shih-chieh* (≈1270–1330) bekannt war. Er war einer der bedeutendsten chinesischen Mathematiker, der einen bemerkenswerten Beitrag zur Entwicklung der chinesischen Algebra und zur Theorie von Gleichungen leistete. Er schrieb zwei wichtige Aufsätze. In seiner zweiten Arbeit, *Ssu-yuan yu-chien* (1303; *Precious Mirror of the Four Elements*), beschrieb er neben anderen mathematischen Problemen auch eine Lösung für eine allgemeine Gleichung 14. Grades. Er verwendete dazu eine Umwandlungsmethode, die er *fan fa* nannte und die eben Horner als das Horner-Schema wiederentdeckte.[2]

Im neunzehnten und frühen zwanzigsten Jahrhundert hatte das Horner-Schema einen herausragenden Platz in einigen Algebrabüchern. Dies lag vor allem an *De Morgan*, der Horners Name und sein Schema in vielen Artikeln, die er veröffentlicht hatte, ausführlich behandelte.

W. G. Horner starb 1837 in Bath.[3]

---

[1] aus dem Englischen übertragen nach *O'Connor; Robertson. 1996, [a] horner.html*

[2] aus dem Englischen übertragen nach *Encyclopædia Britannica. 2000* und *O'Connor; Robertson, 1996, [b] Chu.html*

[3] aus dem Englischen übertragen nach *O'Connor; Robertson. 1996, [a] horner.html*

III. Das Horner-Schema

1. Herleitung

Hier beginne ich nun mit der Entwicklung des Horner-Schemas. Als Beispiel dient dazu ein allgemeines Polynom 3. Grades:

$$f_3(x) = a_3x^3 + a_2x^2 + a_1x + a_0; \quad a_{0,1,2,3} \in \mathrm{R}$$

Aus dieser Funktion wird zunächst die Variable x aus den ersten drei Summanden ausgeklammert. Man erhält dann:

$$f_3(x) = \left(a_3x^2 + a_2x + a_1\right)x + a_0;$$

Klammert man x noch einmal aus den ersten beiden Summanden in der Klammer aus, so ergibt sich:

$$f_3(x) = \left(\left(a_3x + a_2\right)x + a_1\right)x + a_0;$$

Nun kann man die Funktion schrittweise aufteilen und folgendermaßen berechnen:

$G_3 :=$		a_3
$H_3 :=$	$G_3 \cdot x$	$= a_3x$
$G_2 :=$	$H_3 + a_2$	$= a_3x + a_2$
$H_2 :=$	$G_2 \cdot x$	$= (a_3x + a_2)x$
$G_1 :=$	$H_2 + a_1$	$= (a_3x + a_2)x + a_1$
$H_1 :=$	$G_1 \cdot x$	$= ((a_3x + a_2)x + a_1)x$
$G_0 :=$	$H_1 + a_0$	$= ((a_3x + a_2)x + a_1)x + a_0 = f_3(x)$

Schema 1

(Das Symbol $A := B$ hat dabei folgende Bedeutung: Der Wert des Ausdrucks B ergibt den neuen Wert A. H und G sind Hilfsvariablen)[4]

Die durchgeführten Rechnungen lassen sich übersichtlicher als in *Schema 1* anordnen – im sogenannten *Horner-Schema*:

$$\begin{array}{c|c|c|c|c}
 & a_3 & a_2 & a_1 & a_0 \\
\oplus \downarrow & & G_3x = H_3 & G_2x = H_2 & G_1x = H_1 \\
\hline
 & G_3 & G_2 & G_1 & G_0 = f_3(x)
\end{array}$$

Gerechnet wird in Richtung der Pfeile. Nach unten wird immer addiert und nach rechts oben multipliziert. (Die Pfeile sind hier nur zur Verdeutlichung angebracht und werden normalerweise weggelassen.)

[4] vgl. Nitsche, 1968, 126

2. Das allgemeine Horner-Schema:

Zunächst benötigt man eine ganzrationale Funktion vom Grad *n*, ein Polynom *n*-ten Grades:

$$P_n(x) = a_n x^n + a_{n-1} x^{n-1} + \ldots + a_k x^k + \ldots + a_1 x + a_0 = \sum_{k=0}^{n} a_k x^k;$$

$n \in \mathbf{N}_0$; $k \in \{0; n\}$; $a_k \in \mathbf{R}$; [5]

Zum Berechnen eines Funktionswertes setzt man $x = x_0$ $(x_0 \in \mathbf{R})$[6] und nun kann man $P_n(x_0)$ (wie ich im vorherigen Kapitel bereits gezeigt habe) nach dem folgenden *allgemeinen Horner-Schema* ermitteln:

$P_n(x)$ x_0	a_n	a_{n-1} $G_n x_0$	a_{n-2} $G_{n-1} x_0$	... ...	a_k $G_{k+1} x_0$	... ...	a_1 $G_2 x_0$	a_0 $G_1 x_0$
	G_n	G_{n-1}	G_{n-2}	...	G_k	...	G_1	$P_n(x_0)$

Schema 2

In der ersten Zeile werden die Koeffizienten der Funktion $P_n(x)$ notiert, wobei man nicht vorkommende Koeffizienten durch eine Null ersetzten muss. Ansonsten würde man die Werte eines Polynoms niedrigeren Grades berechnen.

Nun geht man nach folgendem Algorithmus vor:

Die erste Stelle der zweiten Zeile bleibt frei, [wobei G_n mit a_n gleichgesetzt wird;] *die weiteren entstehen durch Multiplikation der unmittelbar voranstehenden Glieder der dritten Zeile mit* x_0*; die Glieder der dritten Zeile sind dabei die Summen der darüberstehenden Glieder der ersten und zweiten Zeile; das letzte Glied der dritten Zeile gibt dann den Wert* $P_n(x_0)$ *an.*[7]

[5] Diese Definitionen gelten für die gesamte Arbeit
[6] siehe 5
[7] *Lexikon Technik und exakte Naturwissenschaften. Band 5. Gesteine Kalispeter.* 1972. Seite 1484

3. Abspaltung eines Linearfaktors

Das Horner-Schema besitzt jedoch noch einige weitere Eigenschaften.

Zur Veranschaulichung beginnen wir wieder mit einem Polynom 3. Grades:

$f_3(x) = a_3x^3 + a_2x^2 + a_1x + a_0;$

Für den Fall, das sich bei x_0 eine Nullstelle befindet, kann man einen Linearfaktor $(x - x_0)$ abspalten.

Allgemein kann man zunächst eine Polynomdivision durchführen:

$$\left(a_3x^3 + a_2x^2 + a_1x + a_0\right) : \left(x - x_0\right) =$$

$$= a_3x^2 + \left(a_3x_0 + a_2\right)x + \left(\left(a_3x_0 + a_2\right)x_0 + a_1\right) + \frac{\left(\left(a_3x_0 + a_2\right)x_0 + a_1\right)x_0 + a_0}{x - x_0} =$$

$$= a_3x^2 + \left(a_3x_0 + a_2\right)x + \left(\left(a_3x_0 + a_2\right)x_0 + a_1\right) + \frac{f_3\left(x_0\right)}{x - x_0};$$

Vereinfachen wir nun mit

$b_3 = a_3$ und $b_2 = a_3x_0 + a_2$ und $b_1 = \left(a_3x_0 + a_2\right)x_0 + a_1 = b_2x_0 + a_1$

erhalten wir $f_3(x) = \left(b_3x^2 + b_2x + b_1\right)\left(x - x_0\right) + f_3\left(x_0\right);$

Wenn nun bei x_0 eine Nullstelle ist, so ist $f_3\left(x_0\right) = 0$, und daraus folgt:

$f_3(x) = \left(b_3x^2 + b_2x + b_1\right)\left(x - x_0\right);$

Vergleichen wir nun die Zahlen in der dritten Zeile des Horner-Schemas, so finden wir diese gerade als Koeffizienten des Restpolynoms $[b_3, b_2, b_1]$ *wieder.*[8]

Nehmen wir nun wieder die allgemeine Funktion $P_n(x)$ und betrachten unter diesem Aspekt das allgemeine Horner-Schema (*Schema 2*):

$P_n(x)$ x_0	a_n	a_{n-1} b_nx_0	a_{n-2} $b_{n-1}x_0$	... ...	a_1 b_2x_0	a_0 b_1x_0	**_Schema 3_**
$\frac{P_n(x)}{x-x_0}$	b_n	b_{n-1}	b_{n-2}	...	b_1	$P_n(x_0)$	

Daraus können wir nun folgendes ablesen:

$P_n(x) = \left(b_nx^{n-1} + b_{n-1}x^{n-2} + \ldots + b_2x + b_1\right)\left(x - x_0\right) + P_n\left(x_0\right)$

Falls sich bei x_0 eine Nullstelle befindet, so fällt der letzte Summand weg:

$P_n(x) = \left(b_nx^{n-1} + b_{n-1}x^{n-2} + \ldots + b_2x + b_1\right)\left(x - x_0\right)$

---

[8] Otto, 1985, Seite 20

4. Das erweiterte Horner-Schema

Mit Hilfe der Zahlen in der dritten Zeile des allgemeinen Horner-Schemas (vgl. *Schema* 3) lässt sich auf einfache Weise die erste Ableitung an der Stelle x_0 berechnen.

Es gilt nämlich mit

$$f_3(x) = \left(b_3 x^2 + b_2 x + b_1\right)(x - x_0) + f_3(x_0)$$

für die Ableitung

$$f_3'(x) = (2b_3 x + b_2)(x - x_0) + \left(b_3 x^2 + b_2 x + b_1\right)$$

und für $x = x_0$ wird $(x - x_0) = 0$ und damit folgt

$$f_3'(x_0) = b_3 x_0^2 + b_2 x_0 + b_1;$$

Wieder bilden die Zahlen der dritten Zeile des allgemeinen Horner-Schemas die Koeffizienten eines Polynoms, hier das der ersten Ableitung.

Wir können also das Horner-Schema wie folgt erweitern und erhalten in einem sowohl den Wert von f_3 als auch den von f_3' an der Stelle x_0.[9]

$f_3(x)$ x_0	a_3	a_2 $b_3 x_0$	a_1 $b_2 x_0$	a_0 $b_1 x_0$
 x_0	b_3	b_2 $c_2 x_0$	b_1 $c_1 x_0$	$f_3(x_0)$
	c_2	c_1	$f_3'(x_0)$	

Dieses Schema nennt man schließlich *erweitertes Horner-Schema*.

Allgemein sieht es folgendermaßen aus:

$P_n(x)$ x_0	a_n	a_{n-1} $b_n x_0$	a_{n-2} $b_{n-1} x_0$	... ...	a_2 $b_3 x_0$	a_1 $b_2 x_0$	a_0 $b_1 x_0$
 x_0	b_n	b_{n-1} $c_{n-1} x_0$	b_{n-2} $c_{n-2} x_0$	... ...	b_2 $c_2 x_0$	b_1 $c_1 x_0$	$P_n(x_0)$
	c_{n-1}	c_{n-2}	c_{n-3}	...	c_1	$P_n'(x_0)$	

[9] Otto, 1985, Seite 20

IV. Vorteile des Horner-Schemas

In diesem Kapitel entwickle ich genauer die Vorteile des Horner-Schema und zeige, warum man es, vor allem auch in der Informatik, gerne benutzt.

1. Bedeutung von Additionen und Multiplikationen

Auf dem Computer benötigen Additionen und vor allem Multiplikationen Rechenzeit. Zwar dauert eine Multiplikation nur wenige Millisekunden, doch häufen sich diese und wird zusätzlich noch mit Gleitkommazahlen gerechnet, wird kostbare Rechenzeit benötigt. Es ist also ein Anliegen in der Informatik, Multiplikationen möglichst zu verringern, damit ein Programm nicht unnötig verlangsamt wird.
Aber auch, wenn man den Taschenrechner benutzt, ist es von Vorteil, weniger Rechenschritte zu haben; die Wahrscheinlichkeit sich zu vertippen, und damit auch die Fehlerwahrscheinlichkeit, verringert sich.
Selbst wenn man auf dem Papier rechnet, ist man froh, wenn man möglichst wenig Multiplikationen auszuführen hat, und somit ebenfalls die Gefahr sich zu verrechnen gesenkt werden kann.
Die Notwendigkeit zur Verminderung von Additionen und Multiplikationen ist hiermit gezeigt, und nun beschreibe ich, welche Vorteile das Horner-Schema dazu bietet.

Im folgenden verwende ich eine Beispielfunktion 4. Grades $g(x) = 5x^4 + 3x^3 + 4x^2 - 7x + 1{,}5$ mit einem Wert $x_0 = 2$, an dem die Funktion berechnet werden soll.

2. Berechnung nach der einfachsten Methode

Die einfachste Methode ein Polynom an einem bestimmten Wert zu berechnen besteht darin, die Funktion von links nach rechts durchzurechnen und dabei x dem gesuchten Wert x_0 gleichzusetzen. In dem Beispiel sähe das dann so aus:

$$g(x_0) = 1{,}5 + 5 \cdot \boldsymbol{2} \cdot \boldsymbol{2} \cdot \boldsymbol{2} \cdot \boldsymbol{2} + 3 \cdot \boldsymbol{2} \cdot \boldsymbol{2} \cdot \boldsymbol{2} + 4 \cdot \boldsymbol{2} \cdot \boldsymbol{2} + \left(-7\right) \cdot \boldsymbol{2} = 107{,}5\,;$$

Wir zählen nun die Additionen und Multiplikationen:

Additionen	**4**
Multiplikationen	**10**

Ganz allgemein kann man diesen Rechenvorgang folgendermaßen beschreiben:

Startgleichung	$G_n := a_0$
Rekursionsgleichung (für $k = n-1$ bis 0)	$G_k := a_k \cdot x_0^k + G_{k+1}$
Ergebnis	$P_n(x_0) := G_0$

Methode 1

Daraus ergibt sich die Anzahl der Additionen und Multiplikationen in einer allgemeinen Tabelle:

Additionen	$\boldsymbol{n}$
Multiplikationen	$\boldsymbol{\frac{n(n+1)}{2}}$

Ergebnis 1 [10]

3. Eine „intelligentere“ Methode

Geht man ein bisschen schlauer vor, so kann man die Tatsache ausnutzen, dass, wenn man bereits $H_k = x_0^k$ berechnet hat, man x_0^{k+1} mit $H_{k+1} = H_k \cdot x_0$ ermitteln kann. Angewandt auf das Beispiel bedeutet das folgendes:

$g(x_0) = 1{,}5 + (-7) \cdot \boldsymbol{1} \cdot \boldsymbol{2} + 4 \cdot \boldsymbol{2} \cdot \boldsymbol{2} + 3 \cdot \boldsymbol{4} \cdot \boldsymbol{2} + 5 \cdot \boldsymbol{8} \cdot \boldsymbol{2} = 107{,}5$

Wir zählen wieder die Additionen und Multiplikationen:

Additionen	**4**
Multiplikationen	**8**

Und das Ganze allgemein:

Startgleichung	$G_0 := a_0; \quad H_0 := 1$
Rekursionsgleichung (für $k = 1$ bis n)	$H_k := H_{k-1} \cdot x_0$ $G_k := H_k \cdot a_k + G_{k-1}$
Ergebnis	$P_n(x_0) := G_n$

Methode 2

[10] vgl. Mennicken; Wagenführer, 1977, Seite 17

Allgemein kommen wir somit auf folgendes Ergebnis:

Additionen	***n***
Multiplikationen	**2*n***

Ergebnis 2 [11]

Das ist schon eine Einsparung von Multiplikationen gegenüber dem Vorhergehenden, doch nun werfen wir einen Blick auf das Horner-Schema.

4. Berechnung nach dem Horner-Schema

Unser Beispiel sieht im Horner-Schema folgendermaßen aus.

$g(x)$	5	3	4	-7	1,5
x_0		$5\cdot\mathbf{2}$	$13\cdot\mathbf{2}$	$30\cdot\mathbf{2}$	$53\cdot\mathbf{2}$
	5	13	30	53	107,5

Additionen	**4**
Multiplikationen	**4**

Allgemein gelten für das Horner-Schema folgende Rekursionsgleichungen:

Startgleichung	$G_n := a_n$
Rekursionsgleichung (für $k = n-1$ bis 0)	$G_k := G_{k+1} \cdot x_0 + a_k$
Ergebnis	$P_n(x_0) := G_0$

Methode 3

Da man pro Schritt nur eine Multiplikation und eine Addition benötigt, folgt daraus dieses Ergebnis:

Additionen	***n***
Multiplikationen	***n***

Ergebnis 3 [12]

[11] vgl. Demailly, 1994, Seite 4
[12] vgl. Demailly, 1994, Seite 5

5. Ergebnis der Untersuchung

Vergleicht man nun *Ergebnis 1*, *2* und *3*, so ist deutlich zu erkennen, dass sich zwar die Anzahl der Additionen durch das Horner-Schema nicht verringern lassen, jedoch ist die Anzahl der Multiplikationen erheblich geringer, als bei einfacheren Methoden. Und hier liegt auch der entscheidende Vorteil des Horner-Schemas.

Betrachtet man außerdem noch die verschiedenen Rekursionsgleichungen (*Methode 1* bis *3*), so erkennt man, dass *Methode 3* die einfachste ist. Auch hierdurch ist ersichtlich, dass die Berechnung nach dem Horner-Schema wesentlich leichter zu bewältigen ist. Eben durch ihre Einfachheit kann *Methode 3* gut in der Informatik angewandt werden, womit sich das nächste Kapitel beschäftigt.

V. Anwendung in der Informatik

Mit den inzwischen gewonnenen Erkenntnissen lässt sich ein Computerprogramm schreiben, das unter Verwendung des Horner-Schemas ein Polynom an einer bestimmten Stelle berechnet.

1. Grundlagen

Die Vorteile wurden bereits im letztem Kapitel betrachtet: Das Horner-Schema spart erheblich an Multiplikationen, wodurch ein effektiver Programmcode geschrieben werden kann. Dadurch, dass sich derselbe Vorgang immer wiederholt, ist das Horner-Schema gut für den Computer geeignet.
Verwendet wird die im vorherigen Kapitel bereits angeführte *Methode 3*:

Startgleichung: $G := a[n]$

Rekursionsgleichung: $G := G \cdot x_0 + a[k]$

Ergebnis: $P_n\left(x_0\right) = G$

2. Struktogramm

Das Programm benötigt nun folgende EVA-Routinen (Eingabe–Verarbeitung–Ausgabe):
Zuerst liest man den Grad der Funktion (*n*) und die entsprechenden Koeffizienten (*a*) ein. Zuletzt muss noch der x_0-Wert, an dem die Funktion berechnet werden soll, eingegeben werden.
In der Verarbeitung wird aus den erhaltenen Werten das Ergebnis nach dem Horner-Schema bestimmt. Der neue Wert ergibt sich, indem der vorher erhaltene mit x_0 multipliziert und anschließend der konstante Faktor addiert wird. Dieser immer gleiche Vorgang lässt sich leicht in einer Schleife realisieren.
Am Ende gibt man dann noch das Ergebnis aus.

Daraus ergibt sich ein allgemeines Struktogramm:

LESE n LESE $a[0]$ bis $a[n]$ LESE x_0 $H := a[n]$
Zähle k von n–1 bis *0* $G := G \cdot x_0 + a[k]$
SCHREIBE G

3. Verwirklichung in Turbo-Pascal

Praktisch habe ich dieses Programm in *Turbo-Pascal* realisiert. Hier dargestellt ist der Übersicht wegen eine gekürzte und vereinfachte Version des Programms *Horner.pas* (vgl. *IX Anhang*), das die wichtigsten Elemente übersichtlich darstellt.

`uses ZuHorn1, Base;`	Hier befinden sich die notwenigen Variablen und Prozeduren
`var a: TKoeffizienten;`	**type** TKoeffizienten = array[0..255] of real;
`    n: byte;`	
`    x0: real;`	x_0
`    G: real;`	Hilfsvariable
`    k: integer;`	Zählvariable
`    neu: boolean;`	Nötig, damit Programm mit der Unit *ZuHorn1* kompatibel ist
`begin`	
` ReadFunktion(a,n,neu);`	Lesen von *n*, *a*
` ReadFunktionswert(x0,neu);`	und x_0
` G := a[n];`	
` for k := n–1 downto 0 do`	Rekursive Berechnung nach dem Horner-Schema
`  G := G * x0 + a[k];`	
` WriteErgebnis(G,neu);`	Schreibt *G*
` readln;`	Wartet auf einen Tastendruck, um das Programm zu beenden.
`end.`	

VI. Anwendungsbeispiel

Nun ist es an der Zeit, die erworbenen Kenntnisse an einem Beispiel anzuwenden und nochmals zu verdeutlichen. Sehr gut eignet sich dazu eine Nullstellenberechnung, da in diesem Zusammenhang das Horner-Schema sehr effektiv eingesetzt werden kann.

1. Das Newton-Verfahren

Da mit dem Horner-Schema direkt keine Nullstellen ermittelt werden können, soll es dazu mit dem Newtonschen Näherungsverfahren kombiniert verwendet werden.

Das Newton-Verfahren beruht auf folgender Gleichung:

$$x_{r+1} = x_r - \frac{f(x_r)}{f'(x_r)}; \quad r \in \{0,1,2,...\}$$

x_r mit $r = 0$ stellt dabei einen Startwert da, der sich in der Nähe einer vermuteten Nullstelle befindet, und *r* gibt die Iterationsschritte an. Führt man nun diese Rekursionsgleichung genügend oft aus, mit $r \to \infty$, und *besitzt* $f(x)$ […] *eine einfache Nullstelle* $\bar{x}$*, so konvergiert das Newton'sche Verfahren sicher dann gegen* $\bar{x}$ *,wenn die Iteration nahe genug an* $\bar{x}$ *beginnt.*[13]

Falls x_r nicht konvergiert oder $f'(x_r) = 0$ wird, so muss ein anderer Startwert gewählt werden und mit diesem neu begonnen werden.

Wie man auch aus der Gleichung erkennen kann, werden für jeden Schritt die Werte für $f(x_r)$ und $f'(x_r)$ benötigt. Genau diese Werte lassen sich über das erweiterte Horner-Schema einfach bestimmen.

2. Das Beispiel

Als Beispiel soll die folgende Funktion 4. Grades genommen werden:

$$f(x) = 0{,}2x^4 - 1{,}4x^3 - 11x^2 + 11x + 30$$

Nun soll eine Nullstelle in der Umgebung von $x_0 = 1$ gesucht werden.

(Die Ergebnisse werden auf die dritte Stelle nach dem Komma gerundet.)

---

[13] Nische, 1968, Seite 78

Wir berechnen nach dem Horner-Schema $f(1)$ und $f'(1)$:

	$0{,}2$	$-1{,}4$	-11	11	30
$x_0 = 1$		$0{,}2$	$-1{,}2$	$-12{,}2$	$-1{,}2$
	$0{,}2$	$-1{,}2$	$-12{,}2$	$-1{,}2$	$\mathbf{28{,}80}$
$x_0 = 1$		$0{,}2$	-1	$-13{,}2$	
	$0{,}2$	-1	$-13{,}2$	$\mathbf{-14{,}40}$	

Eingesetzt in Gleichung von Newton: $x_1 = 1 - \frac{28{,}8}{-14{,}4} = 3$;

Es folgt der nächste Iterationsschritt mit $x_1 = 3$.

Wir berechnen wieder nach dem Horner-Schema:

	$0{,}2$	$-1{,}4$	-11	11	30
$x_1 = 3$		$0{,}2 \cdot 3$	$-0{,}8 \cdot 3$	$-13{,}4 \cdot 3$	$-29{,}2$
	$0{,}2$	$-0{,}8$	$-13{,}4$	$-29{,}2$	$\mathbf{-57{,}60}$
$x_1 = 3$		$0{,}2 \cdot 3$	$-0{,}2 \cdot 3$	$-14 \cdot 3$	
	$0{,}2$	$-0{,}2$	-14	$\mathbf{-71{,}20}$	

Wieder eingesetzt in Gleichung: $x_2 = 3 - \frac{-57{,}6}{-71{,}2} \approx 2{,}191$;

Da die Werte nun komplizierter werden, sei für die weiteren Berechnungen auf das Computerprogramm *Horner2.exe* im Anhang verwiesen, das die Werte der Funktion und der Ableitung nach dem Horner-Schema berechnet und übersichtlich darstellt. Daraus ergibt sich dann:

$$x_3 = 2{,}191 - \frac{-8{,}820}{-48{,}950} \approx 2{,}011$$

$$x_4 = 2{,}011 - \frac{-0{,}479}{-43{,}721} \approx 2{,}000$$

Prüfen wir für $x_4 = 2$ im Horner-Schema nach

	$0{,}2$	$-1{,}4$	-11	11	30
$x_4 = 2$		$0{,}2 \cdot 2$	$-1 \cdot 2$	$-13 \cdot 2$	$-15 \cdot 2$
	$0{,}2$	-1	-13	-15	$\mathbf{0}$

so können wir tatsächlich eine Nullstelle nachweisen.

Weiterhin können wir gleichzeitig den Linearfaktor mithilfe des Schemas abspalten. Daraus ergibt sich: $f(x) = (x-2)\left(0{,}2x^3 - x^2 - 13x - 15\right)$.

Bildet jetzt man mit dem zweiten Faktor eine neue Funktion

$g(x) = 0{,}2x^3 - x^2 - 13x - 15$,

so kann man hiermit fortfahren und eine Nullstelle von $g(x)$ suchen, und damit gleichzeitig eine Nullstelle von $f(x)$.

Hierbei möchte ich es belassen und auf die Programme im Anhang verweisen mit deren Hilfe man hier fortfahren kann.

(Als kleiner Tipp: In der Umgebung von $x = -4$ befindet sich noch eine Nullstelle. Die zwei restlichen kann man dann einfach über die Lösungsformel bestimmen, oder das Programm *Newton.exe* (vgl. Anhang) verwenden.)

VII. Schluss

Ich hoffe, dass durch meine Arbeit das Prinzip, die Funktionsweise und die Möglichkeiten der Anwendung des Horner-Schemas klargemacht werden konnten.

Da ein Schwerpunkt meiner Arbeit die Verwendung des Horner-Schemas in der Informatik darstellt, kann ich nur noch einmal auf die Programme im Anhang verweisen, die als Ergänzung und Fortführung der bereits beschriebenen Bereiche anzusehen sind. Diese Programme sind sicherlich auch interessant, wenn man die dahinterstehenden Mechanismen nicht kennt. Besonders praktisch finde ich das Programm *Newton.exe* das mithilfe des Newton-Verfahrens und der Horner-Schemas ziemlich schnell und sicher Nullstellen komplizierter Polynome findet.

VIII. Quellen und Hilfsmittel

1. Literaturverzeichnis

- Barth, Friedrich; Mühlbauer, Paul; Dr. Nikol, Friedrich; Wörle, Karl: *Mathematische Formeln und Definitionen.* Bayerischer Schulbuch-Verlag, München [6]1994.
- Demailly, Jean-Pierre: *Gewöhnliche Differentialgleichungen. Theoretische und numerische Aspekte.* Franz. Originalausgabe: *Analyse Numérique et Equations Différentielles.* Presses Universitaires de Grenoble, 1991. Aus dem Französischen von Mathias Heckele. Hrsg. v. Klas Diederich. Friedr. Vieweg & Sohn Verlagsgesellschaft mbH, Braunschweig/Wiesbaden 1994.
- *Der Grosse Brockhaus. Kompaktausgabe. Aktualisierte 18. Auflage in 26 Bänden. Band 15. Moos bis objektives Verfahren.* F. A. Brockhaus, Wiesbaden [18]1983.
- Henrici, Peter: *Elemente der numerischen Analysis. Band 1.* Originalausgabe: *Elements of Numerical Analysis*. John Wiley & Sons Inc. New York 1964. Deutsche Übersetzung: Bibliographisches Institut AG, Mannheim 1972.
- Kroll, Wolfgang: *Grund- und Leistungskurs Analysis Lehr- und Arbeitsbuch. Band 1: Differentialrechnung 1.* Ferd. Dümmlers Verlag, Bonn 1985.
- *Lexikon Technik und exakte Naturwissenschaften. Band 5. Gesteine Kalispeter.* Fischer Taschenbuch Verlag GmbH, Frankfurt am Main, Oktober 1972.
- Mennicken, Reinhard; Wagenführer, Ekkehard: *Numerische Mathematik 1.* Rowohlt Taschenbuch Verlag GmbH, Reinbek bei Hamburg, Mai 1977.
- Nitsche, Joachim: *Praktische Mathematik.* Bibliographisches Institut AG, Mannheim 1968.
- Otto, Alexandra: *Analysis mit dem Computer.* B. G. Teubner, Stuttgart 1985.
- Wörle, Karl; Kratz, Johannes. Dr. Keil, Karl-August: *Infinitesimalrechnung. Band 7.* Bayerischer Schulbuch-Verlag, München [8]1972

2. Internetquellen

- Encyclopædia Britannica: *Chu Shih-chieh.* Britannica.com Inc. 1999-2000 http://www.britannica.com/eb/article?eu=84673
- [a] O'Connor, J. J.; Robertson, E. F.: *William George Horner.*
  School of Mathematics and Statistics
  (http://www-maths.mcs.at-andrews.ac.uk/)
  University of St Andrews, Scotland (http://www.st-andrews.ac.uk)
  JOC/EFR December 1996
  http://www-groups.dcs.st-and.ac.uk/~history/Mathematicians/Horner.html

– [b] O'Connor, J. J.; Robertson, E. F.: *Chu Shih-Chieh.*
School of Mathematics and Statistics
(http://www-maths.mcs.at-andrews.ac.uk/)
University of St Andrews, Scotland (http://www.st-andrews.ac.uk/)
JOC/EFR December 1996
http://www-groups.dcs.st-and.ac.uk/~history/Mathematicians/Chu.html

3. Verwendete Software

– *Microsoft Word 2000. (SR-1)*. Microsoft Corporation 1983-1999

– *Microsoft Formel-Editor Version 3.01*. Design Science, Inc. 1990-1998. Microsoft Corporation 1998

– *Borland Pascal Version 7.0*. Borland International Inc. 1983, 92

– *Turbo Pascal Version 6.0.* Borland International Inc. 1983, 90

IX. Anhang

Der Arbeit beigefügt ist eine Diskette, auf der sich die bereits erwähnten Programme und Quellcodes befinden. Der Inhalt der Diskette ist in der Datei *Readme.txt* aufgelistet.

X. Erklärung

Ich erkläre hiermit, dass ich die Facharbeit ohne fremde Hilfe angefertigt und nur die im Literaturverzeichnis angeführten Quellen und Hilfsmittel benützt habe.

___________________, den_____________ ____________________________

Ort Datum Unterschrift des Verfassers