

Benford-Verteilung

Benford-Verteilung Haftendorn 2011

siehe Erklärungen "Benford-Gesetz-Beweis.pdf" auf www.mathematik-verstehen.de
Bereich Stochastik → Verteilungen → Diskrete Verteilungen.

Auf diesen Seiten hier werden die Benford-Wahrscheinlichkeiten berechnet und dargestellt.

Auf Seite 3 wird das Wachstum mit der e-Funktion simuliert und die führenden Ziffern werden untersucht.

Sie sind Benford-verteilt. Das wird noch mit einem Chiquadratstest untermauert.

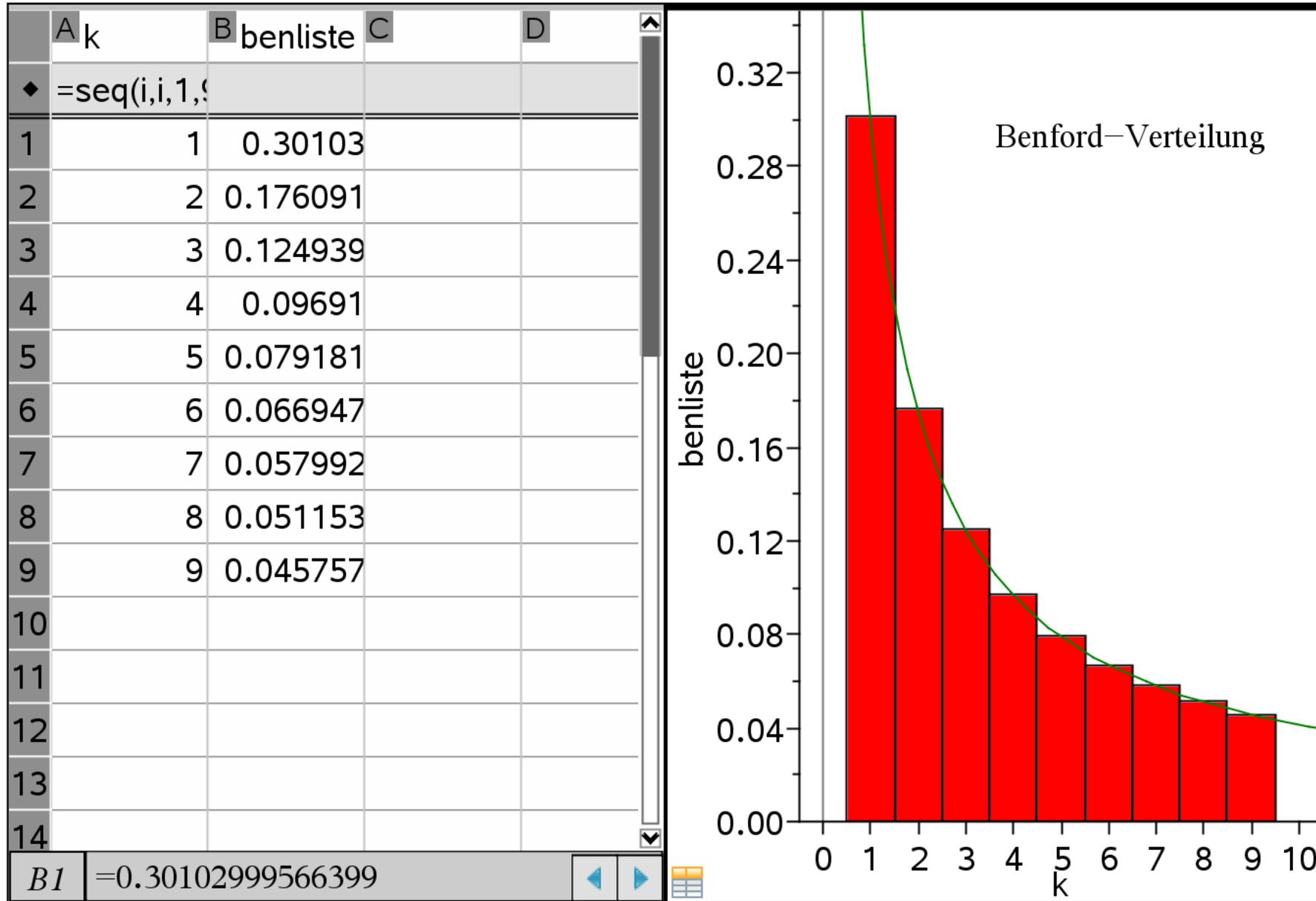
Theoretische Benford-Verteilung

$\text{ben}(x) = P(x \text{ ist führende Ziffer})$

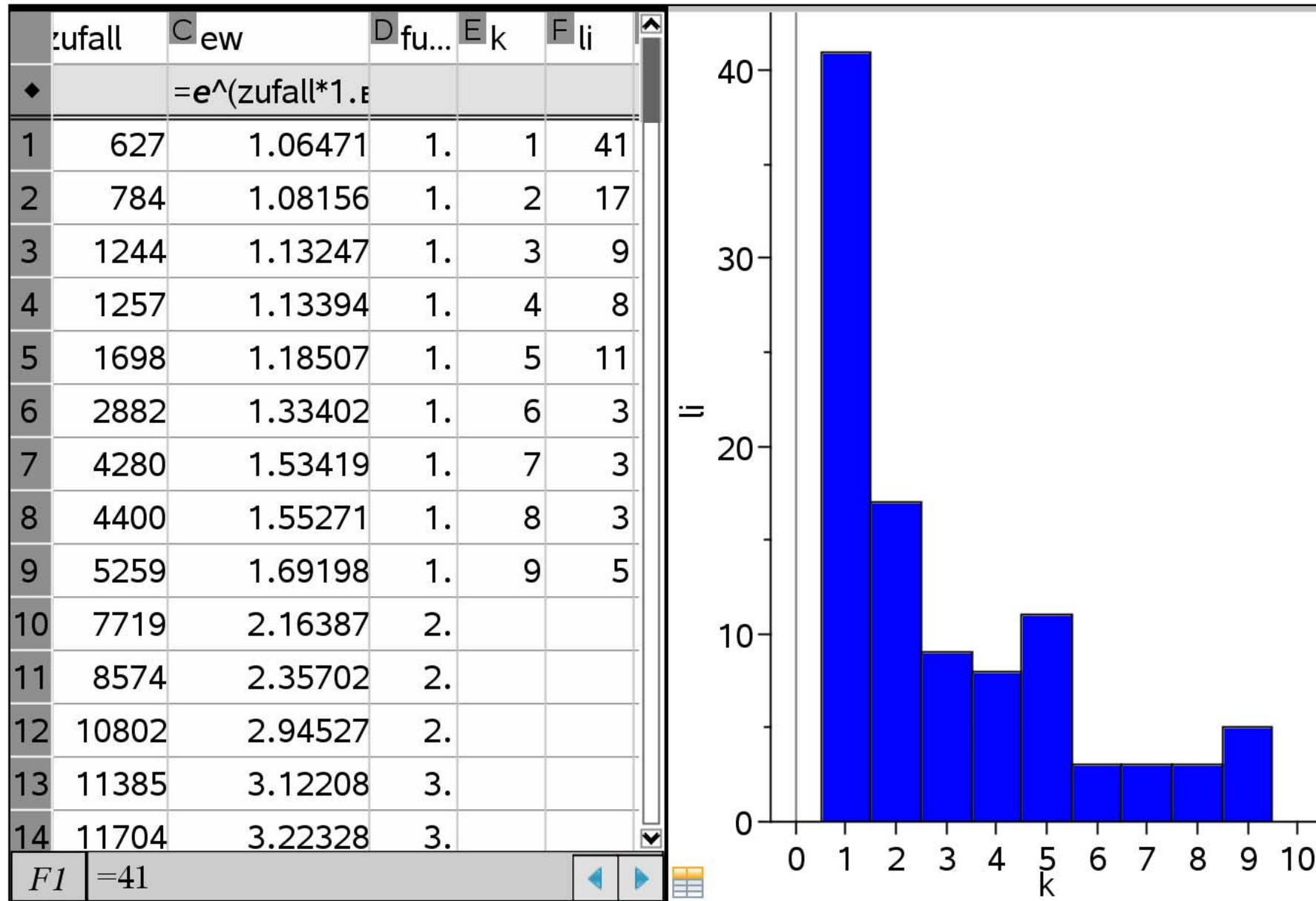
$$\text{ben}(x) := \log_{10} \left(1 + \frac{1}{x} \right) \quad \blacktriangleright \textit{Fertig}$$

$\text{benliste} := \text{approx}(\text{seq}(\text{ben}(x), x, 1, 9))$

$\blacktriangleright \{0.30103, 0.176091, 0.124939, 0.09691, 0.079181, 0.066947, 0.057992, 0.051153, 0.045757\}$



1.2



1.3

Für die Simulation wird zunächst ein Zufallsliste mit 100 Werten zwischen 1 und 10000 erzeugt.

zufall:=seq{randint(1,10000-1)} Mit **sortA zufall** wird diese Liste sortiert.

Anmerkung: sortA gibt die genannte Liste sortiert wieder zurück, man kann also die ursprüngliche Liste nicht bewahren. Dieses Verhalten ist didaktisch höchst ungeschickt. Dabei wird auch die ursprüngliche Definition der Liste gelöscht.

Mit **ew:=e^{0.0001·zufall}** werden e-Funktionswerte zwischen 1 und etwa 2500 erzeugt.

Die auf der letzten Seite definierte Funktion **fzif(x)** gibt die führenden Ziffern dieser Werte zurück. Sie werden in der Liste **fuehrend** abgelegt.

Leider wirkt meine Funktion **fzif(ew)** nicht, ist also nicht "listable".

daher ist **fzif(c1)** nach unten gezogen. Mit **seq** wäre es auch gegangen.

Schließlich ist mit **li:=seq{countIf{fuehrend,j·1.,j,1,9}}**

die Häufigkeitsliste der Ziffern erzeugt. |

Mit Re-Maus Ergebnisdiagramm ist sie als Histogramm dargestellt.

Chi-Quadrattest

Es bietet sich an, hier einen Chi-Quadrattest durchzuführen.

Die Rechnungen (lt. meinem Statistigheft Seite 30) sind auf der nächsten Seite durchgeführt.

Es ergibt sich **chiq** ▶ 10.7415 und damit $\chi^2\text{Cdf}(\text{chiq}, 1000, 8)$ ▶ 0.21678

Die Simulation ist also mit der Benford-Verteilung verträglich, was nicht überrascht.

```
 $\chi^2\text{GOF li,hi,8: stat.results}$  ▶ 

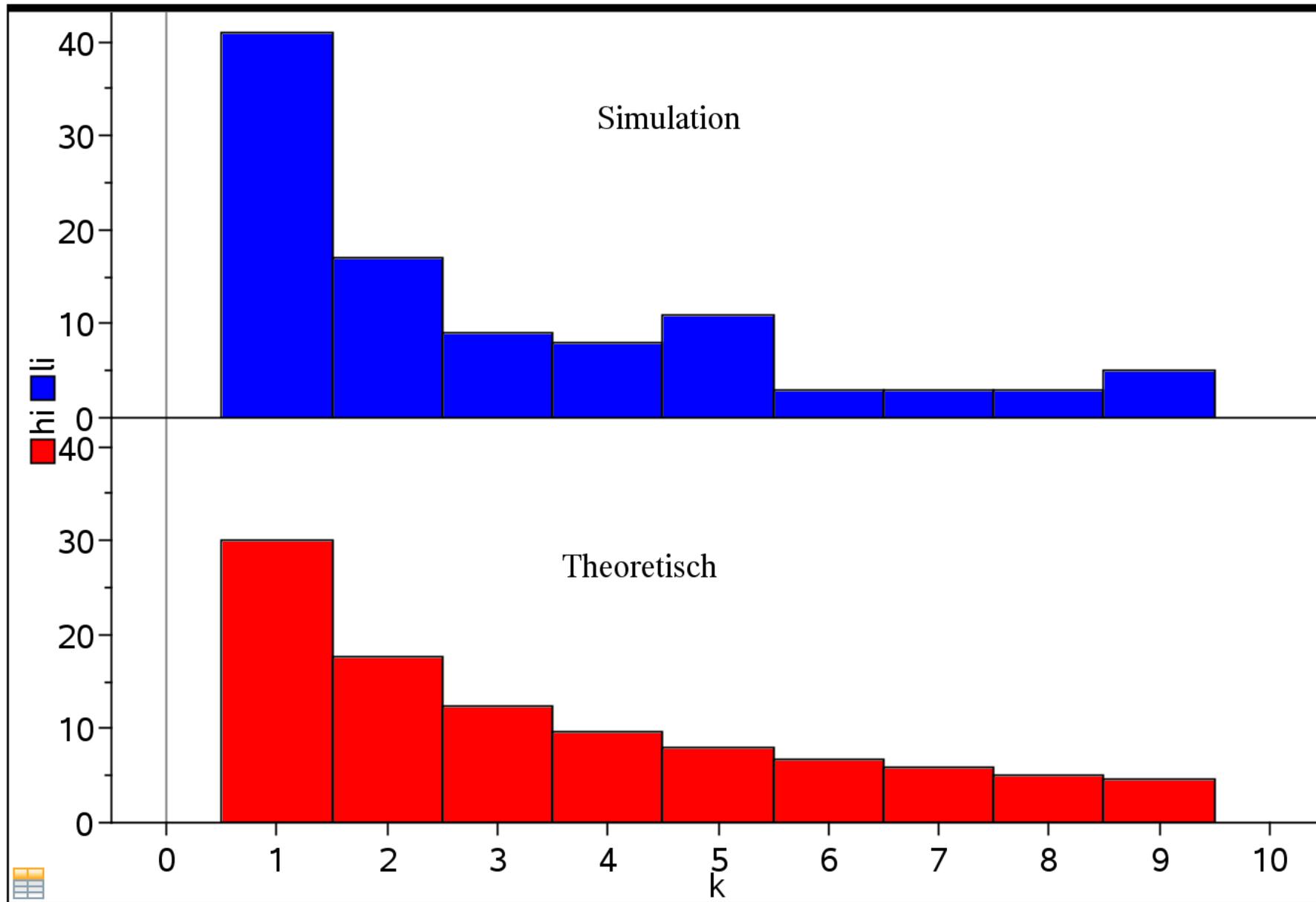
|              |                 |
|--------------|-----------------|
| "Titel "     | " $\chi^2$ GOF" |
| " $\chi^2$ " | 10.7415         |
| "PVal"       | 0.21678         |
| "df"         | 8.              |
| "CompList "  | "{...}"         |


```

Also die Auswertung geht auch in einem Rutsch.

	A k	B li	C benliste	D hi	E delta	F deltaq	G deltaqre	H	
◆				=benliste*	=hi-li	=delta^2	=deltaq/hi		
1	1	41	0.30103	30.103	-10.897	118.745	3.94461		
2	2	17	0.176091	17.6091	0.609126	0.371034	0.021071		
3	3	9	0.124939	12.4939	3.49387	12.2072	0.977051		
4	4	8	0.09691	9.691	1.691	2.85949	0.295066		
5	5	11	0.079181	7.91812	-3.08188	9.49796	1.19952	chiq	
6	6	3	0.066947	6.69468	3.69468	13.6507	2.03903	10.7415	
7	7	3	0.057992	5.79919	2.79919	7.83549	1.35113	0.21678	
8	8	3	0.051153	5.11525	2.11525	4.47429	0.874696		
9	9	5	0.045757	4.57575	-0.424251	0.179989	0.039335		
10									
11									
12									
13									
14									
H7		=χ ² cdf(h6,1000,8)							

1.6



1.7

"fzif" erfolgreich gespeichert	fzif (78.34) ▶ 7.
Define fzif (x)=	fzif (678686876876) ▶ 6
Func	fzif (2.E-6) ▶ 2.
© bestimmt die führende Ziffer	fzif (1) ▶ 1
Local z,xx	fzif (0) ▶ 0
xx:= x	fzif (-654) ▶ 6
While xx>10	
xx:= $\frac{xx}{10}$	
EndWhile	
If xx≥1 Then	
z:=int(xx)	
ElseIf xx=0 Then	
z:=0	
Else	
While xx<1	
xx:=xx·10	
EndWhile	
z:=int(xx)	
EndIf	

1.8