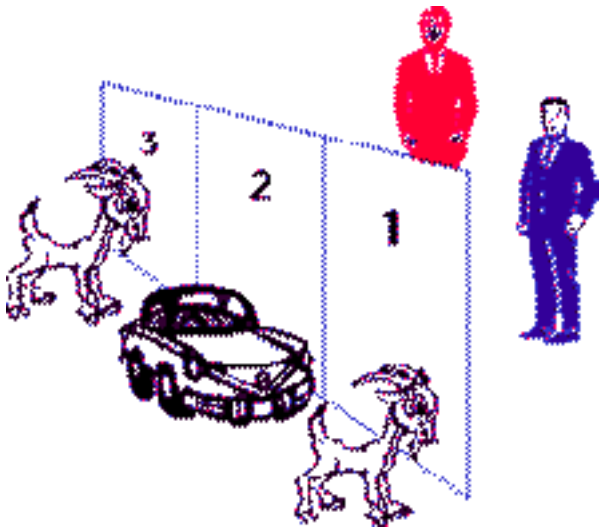
Prof. Dr. Ludwig Paditz 29.10.2002

Zwei Ziegen und ein Auto

In der amerikanischen Spielshow **"Let`s make a deal"** ist als Hauptpreis ein Auto ausgesetzt. Hierzu sind auf der Bühne drei verschlossene Türen aufgebaut. Hinter einer rein zufällig ausgewählten Tür ist der Hauptpreis platziert, hinter den beiden anderen jeweils eine Ziege.



Der Kandidat wählt eine der Türen, beispielsweise Tür 1 aus; diese bleibt aber vorerst verschlossen. Der Spielleiter, der weiß, hinter welcher Tür das Auto steht, öffnet daraufhin mit den Worten **"Soll ich Ihnen mal etwas zeigen?"** eine der beiden anderen Türen, zum Beispiel Tür 3, und eine Ziege schaut ins Publikum. Der Kandidat hat nun die Möglichkeit, bei seiner ursprünglichen Wahl zu bleiben oder die andere verschlossene Tür (in unserem Beispiel Nr. 2) zu wählen. Er erhält dann den Preis hinter der von ihm zuletzt gewählten Tür.

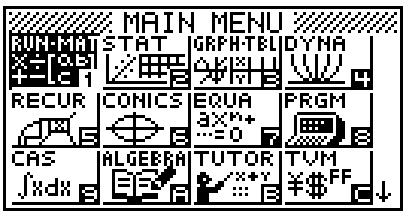
In der Kolumne "Ask Marilyn" des amerikanischen Wochenmagazins "Parade" erklärte die Journalistin Marilyn vos Savant, **dass ein Wechseln des Kandidaten zu Tür 2 dessen Chancen im Vergleich zum Festhalten an Tür 1 verdoppeln würde.**

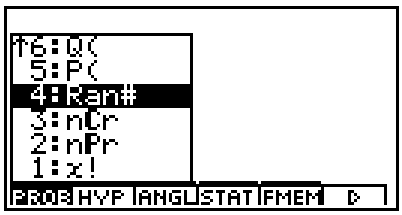
Das war die Geburtsstunde des **"Ziegenproblems"** im Jahre 1991, denn die Antwort von Frau Marilyn bescherte der Redaktion der "Parade" eine Flut von Leserbriefen gegenteiliger Meinung. Das Problem wurde sowohl in der amerikanischen als auch in der deutschen Öffentlichkeit heiß diskutiert. .....

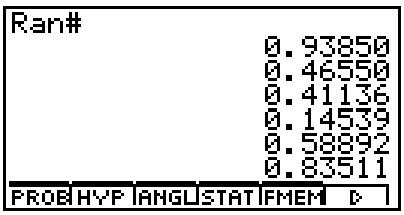
Problemlösungsprozeß für das Ziegenproblem:

Wir wollen dieses Problem experimentell mit dem GTR bearbeiten und benötigen dazu ganzzahlige Zufallszahlen von 1 bis 3.

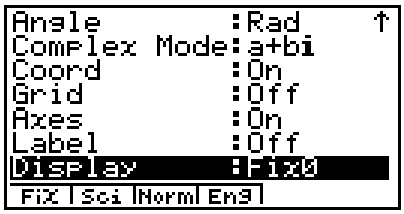
Der GTR stellt gleichmäßig in dem Intervall (0;1) verteilte Zufallszahlen bereit: **Ran#**

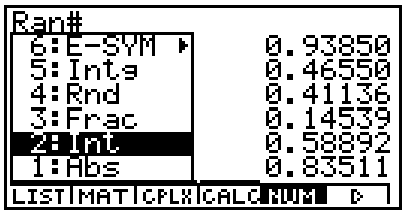


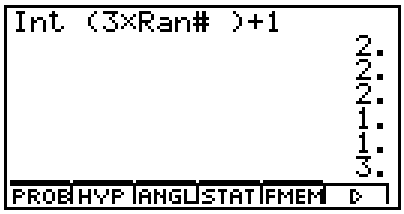




Mit der Transformation **Int(3 × Ran#) +1** wird eine gewünschte Zufallszahl bereitgestellt.







Int(...) berechnet hierbei den Ganzteil (Integer Part) einer Zahl.

Simulation von drei parallelen Folgen von Zufallszahlen:

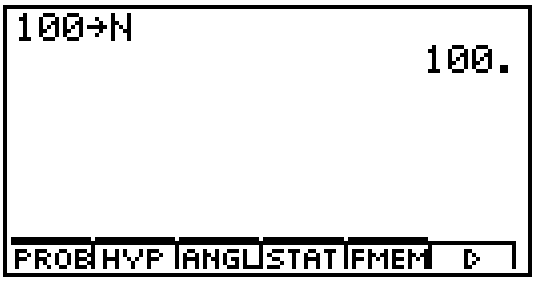
Ran# 1 (Auto) , Ran# 2 (Kandidat), Ran# 3 (Moderator)

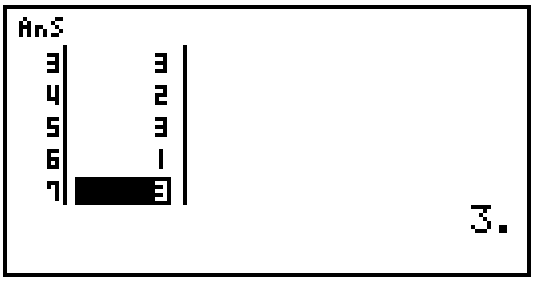
1. Schritt: Nr. der zufälligen Autoplatzierung simulieren: Zufallszahl **a = Int(3 × Ran# 1) +1**
2. Schritt: Nr. der zufälligen Kandidatenwahl simulieren: Zufallszahl **k = Int(3 × Ran# 2) +1**
3. Schritt: Nr. der zufälligen Moderatortür simulieren: **m = f(a, k)** , wobei **f(a, k)** das Verhalten des Moderators gemäß der folgenden Tabelle beschreibt.
4. Schritt: Nr. der endgültig vom Kandidaten gewählten Tür: **w**,
   Spielerstrategie: Tür beibehalten (Kode **s = 0**), Tür wechseln (Kode **s = 1**)

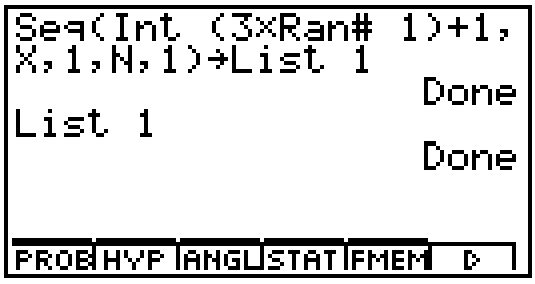
Autotür a Liste 1	Kandidatentür k Liste 2	m = f(a, k) Liste 3	w (s = 0) Liste 4	w (s = 1) Liste 4
1	1	2 v 3	1	3 v 2
1	2	3	2	1
1	3	2	3	1
2	1	3	1	2
2	2	1 v 3	2	3 v 1
2	3	1	3	2
3	1	2	1	3
3	2	1	2	3
3	3	1 v 2	3	2 v 1

5. Schritt: Auswertung der einzelnen Gewinnspiele
6. Schritt: Gesamtbilanz in **N** simulierten Spielen.

Vorbereitung eines Simulationsprogramms im RUN-Menü des GTR und Test einiger Programmschritte:

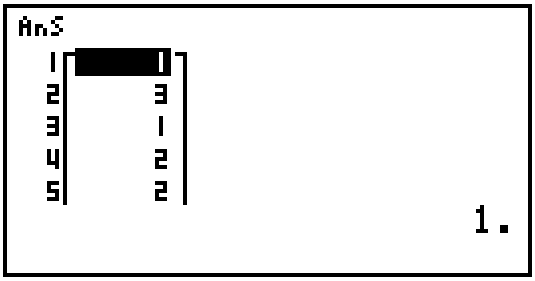


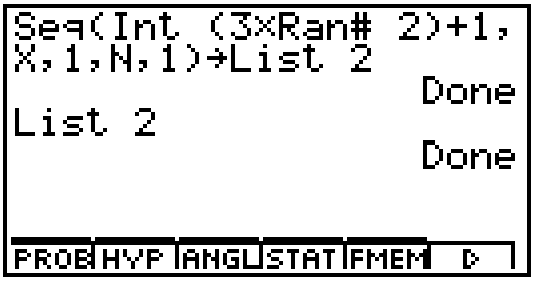




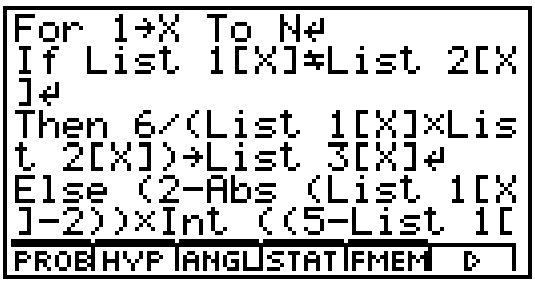
Simulation der Autoplatzierung a

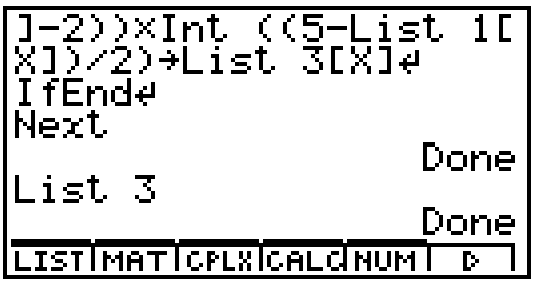
Simulation der Kandidatenauswahl k:



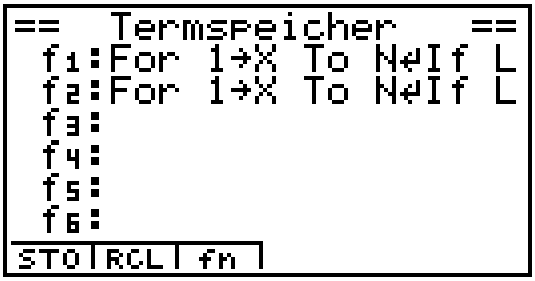


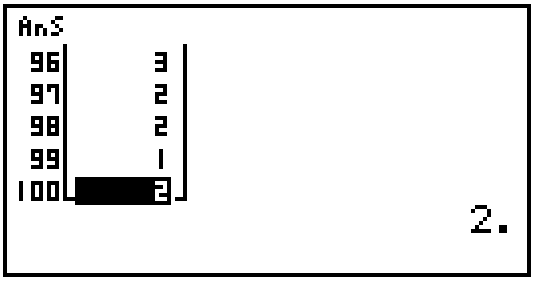
Simulation der Moderatortür m:





Abspeicherung der Befehlsfolge (für eine spätere Programmübernahme) **im Termspeicher:**





Mathematische Modellierung des Verhaltens des Moderators (3. Schritt):

Das (zufällige) Verhalten des Moderators ist eindeutig im Fall $\mathbf{a \neq k}$. Es gilt hier offensichtlich

$$\boxed{\mathbf{a \times k \times m = 6}\,, \quad \text{d.h.} \quad \mathbf{m = f(a, k) = 6 / (a \times k)}}$$

Im Fall $\mathbf{a = k}$ hat der Moderator zwei Möglichkeiten, die mithilfe der Zufallszahl **x** beschrieben werden:

$$\mathbf{x = Int(\ 2 \times Ran\#\ 3\ )\ \in\ \{\ 0\ ;\ 1\ \}}$$

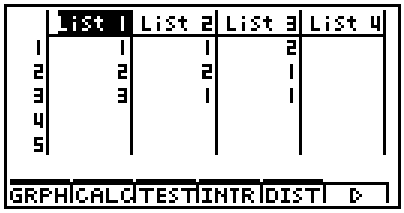
Es gilt hierbei für $\mathbf{m}$ eine lineare Gleichung der Form $\mathbf{m = b \times x + c}$:

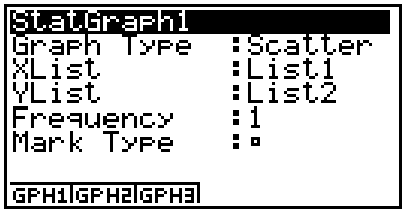
$$\begin{aligned} \mathbf{a = k = 1:} &\qquad \mathbf{m = 1 \times x + 2 \in \{\ 2\ ;\ 3\ \}} \\ \mathbf{a = k = 2:} &\qquad \mathbf{m = 2 \times x + 1 \in \{\ 1\ ;\ 3\ \}} \\ \mathbf{a = k = 3:} &\qquad \mathbf{m = 1 \times x + 1 \in \{\ 1\ ;\ 2\ \}} \end{aligned}$$

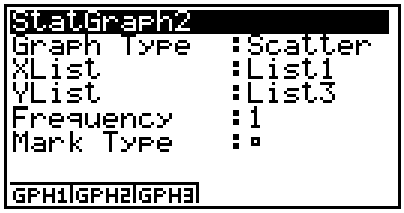
Um alle drei Fälle in einem zusammenzufassen, müssen wir eine einfache Formel $\mathbf{b = b(a)}$ für den Koeffizienten und $\mathbf{c = c(a)}$ für das Absolutglied finden. Es soll also gelten:

$$\mathbf{b(1) = b(3) = 1} \ \text{ und } \ \mathbf{b(2) = 2}, \ \text{ sowie } \ \mathbf{c(2) = c(3) = 1} \ \text{ und } \ \mathbf{c(1) = 2}.$$

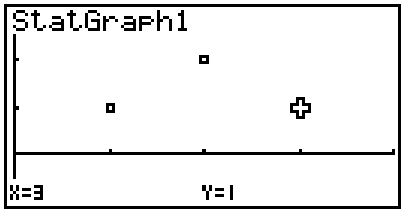
(Die Zahlenpaare skizzieren und mit einem passenden Graphen einer Funktion verbinden.)

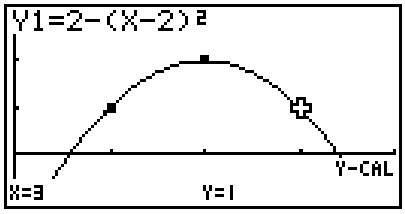


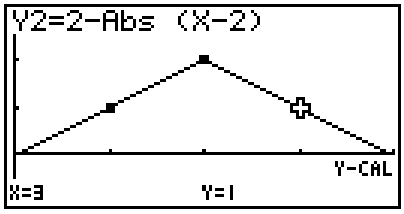




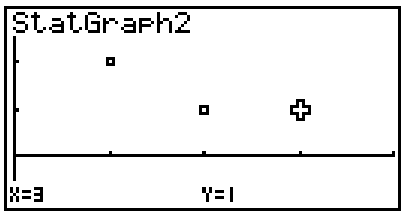
Zahlenpaare als Scatter-Plot darstellen und als Hintergrundbild Pict1 bzw. Pict2 abspeichern.

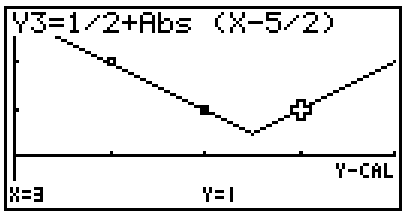


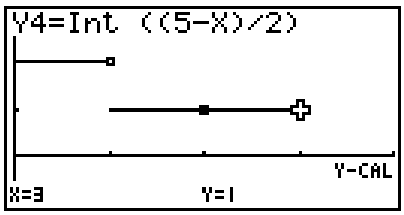




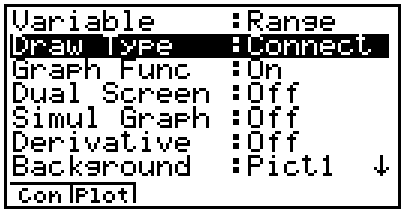
Geeignete Graphen (Kurven) durch die drei Punkte legen: $\mathbf{b = b(a)}$

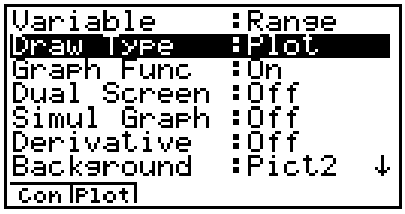


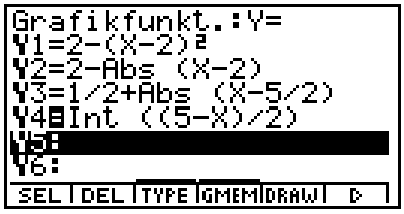




Geeignete Graphen (Kurven) durch die drei Punkte legen: $\mathbf{c = c(a)}$







Das benutzte SET-UP und die Definition der benutzten Funktionen.

Man findet also z.B. folgende Lösung, wobei **Abs(...)** den Absolutbetrag einer Zahl bezeichnet:

$$\mathbf{b(a) = 2 - (\,a - 2\,)^2} \quad \text{oder} \quad \mathbf{b(a) = 2 - Abs(\,a - 2\,)}$$

und

$$\mathbf{c(a) = 1/2 + Abs(\,a - 5/2\,)} \quad \text{oder} \quad \mathbf{c(a) = Int(\,(\,5 - a\,)\,/\,2\,)}\,.$$

Damit gilt im Fall $\mathbf{a = k}$:

$$\boxed{\mathbf{m = f(a) = b \times x + c = (\,2 - Abs(\,a - 2\,)\,) \times Int(\,2 \times Ran\#\,3) + Int(\,(\,5 - a\,)\,/\,2\,)}}$$

Die Fallunterscheidung $\mathbf{a = k}$ bzw. $\mathbf{a \neq k}$ wird mithilfe des **If – Then – Else – IfEnd** - Befehls programmiert.

Mathematische Modellierung der Spielerstrategie (4. Schritt):

Die Spielerstrategie **0** (Tür nicht wechseln) oder **1** (Tür wechseln) wird ebenfalls mithilfe des **If – Then – Else – IfEnd** - Befehls unterschieden, wobei die am Ende vom Kandidaten gewählte Tür **w** folgender Berechnungsvorschrift genügt:

$$\boxed{\mathbf{w = 6 - m - k} \ \text{(Strategie 1)} \quad \text{bzw.} \quad \mathbf{w = k} \ \text{(Strategie 0).}}$$

Die Simulation von **N** Spielverläufen (z.B. **N = 50** oder **N = 100**) wird über eine Laufanweisung mit dem **For – To – Next** - Befehl programmiert.

Für die einfache Erzeugung von **N** Zufallszahlen steht der **Seq(...)** – Befehl zur Verfügung (1. und 2. Schritt).

Das Simulationsprogramm:

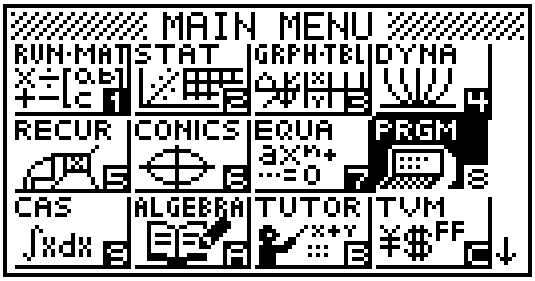
```
Filename:ZIEGE.FX
ClrText↵
ClrList↵
Fix 0↵
" "↵
"  LET'S MAKE A DEAL"↵
" "↵
" "↵
"  ANZAHL DER SPIELE?"↵
" "↵
?→N↵
ClrText↵
"STRATEGIE?"↵
```

```
" "↵
"1 FUER WECHSELN"↵
"0 FUER NICHT WECHSELN"↵
?→S↵
ClrText↵
"AUTO-PLATZIERUNG"↵
Seq(Int (3×Ran#1)+1,X,1,N,1)→List 1↵
List 1◢
"SPIELER-TIPP"↵
Seq(Int (3×Ran#2)+1,X,1,N,1)→List 2↵
List 1+.01List 2→List 7↵
Fix 2↵
List 7◢
"MODERATOR-TIPP"↵
" "↵
"ER OEFFNET EINE TUER"↵
For 1→X To N↵
If List 1[X]≠List 2[X]↵
Then 6÷(List 1[X]×List 2[X])→List 3[X]↵
Else (2-Abs (List 1[X]-2))×Int (2×Ran#3)+Int ((5-List 1[X])÷2)→List 3[X]↵
IfEnd↵
Next↵
Fix 3↵
List 7+.001List 3→List 8↵
List 8◢
ClrText↵
Fix 0↵
Locate 1,2,"SPIELERSTRATEGIE:"↵
Locate 19,2,S↵
If S=1↵
Then 6-List 2-List 3→List 4↵
Else List 2→List 4↵
IfEnd↵
Fix 4↵
List 8+.0001List 4→List 9↵
List 9◢
ClrText↵
" "↵
"AUSWERTUNG DER SPIELE"↵
"1 FUER GEWINNSPIEL"↵
"0 FUER VERLUSTSPIEL"↵
For 1→X To N↵
If List 4[X]=List 1[X]↵
Then 1→List 5[X]↵
Else 0→List 5[X]↵
IfEnd↵
Next↵
```

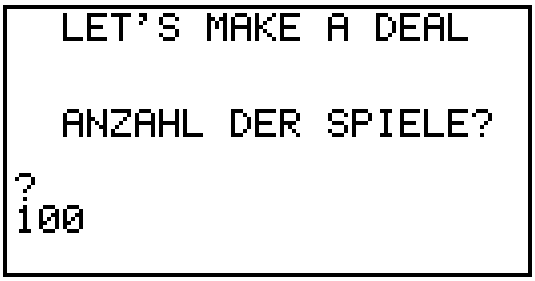
```
ClrText↵
" "↵
"ANZAHL GEWINNSPIELE"↵
Cuml List 5→List 6↵
Fix 0↵
List 6◢
ClrText↵
Locate 3,2,"  AUSWERTUNG"↵
Locate 1,4,"ANZAHL  SPIELE:"↵
Locate 18,4,N↵
Locate 1,5,"SPIELSTRATEGIE:"↵
Locate 18,5,S↵
Locate 1,6,"ANZAHL GEWINNE:"↵
Locate 18,6,List 6[N]↵
Locate 1,7,"ANZAHL VERLUSTE:"↵
Locate 18,7,N-List 6[N]↵
Stop↵
```

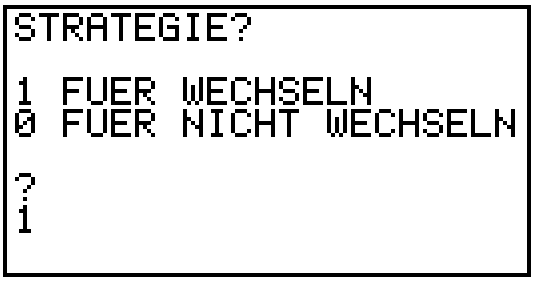
Taschenrechnerbilder:

Programmstart des Programms ZIEGE.FX :

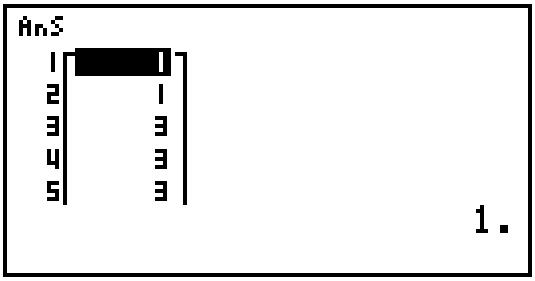


Eingabe der Spieleanzahl N = 100 und der Spielerstrategie S = 1 :

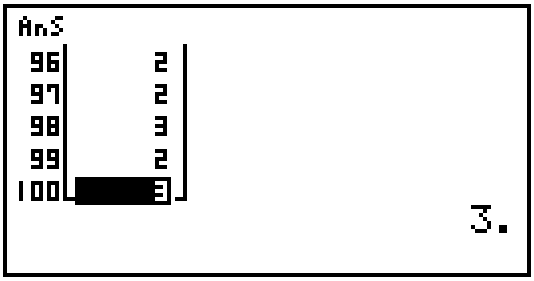




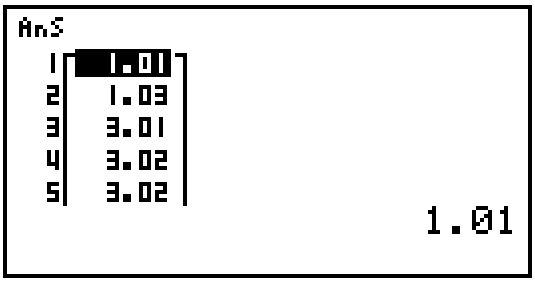
Die simulierte Autoplatzierung a in 100 Spielverläufen:



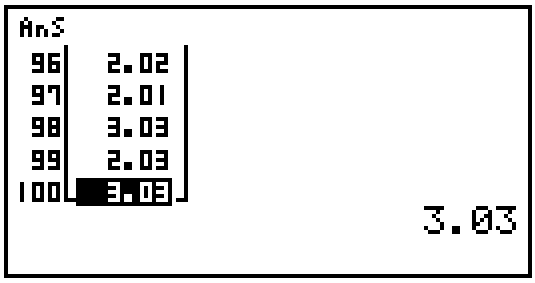
...



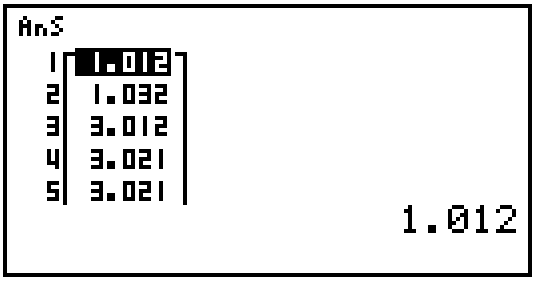
Die simulierte Kandidatenwahl k (als rechte Ziffer) in 100 Spielverläufen:



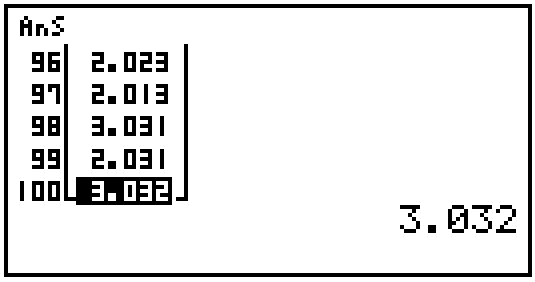
...



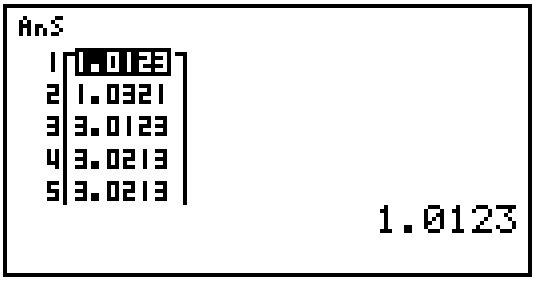
Die simulierte Moderatortür m (als rechte Ziffer) in 100 Spielverläufen:



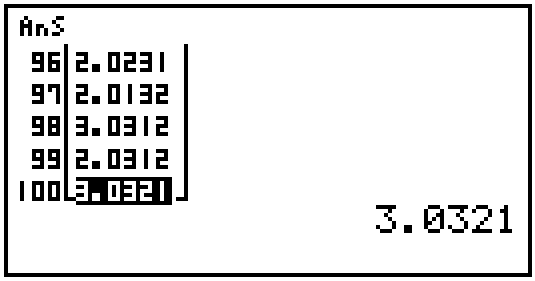
...



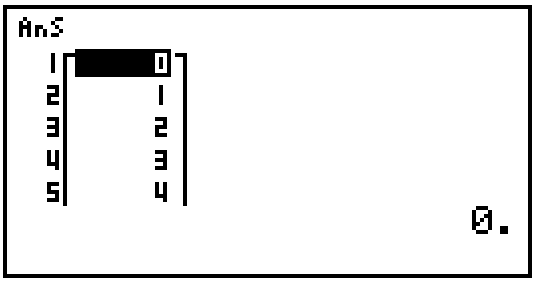
Die simulierte Spielerstrategie w (als rechte Ziffer) in 100 Spielverläufen:



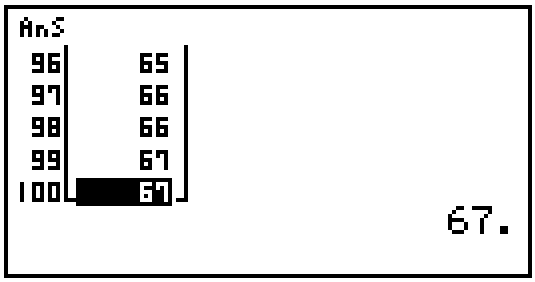
...



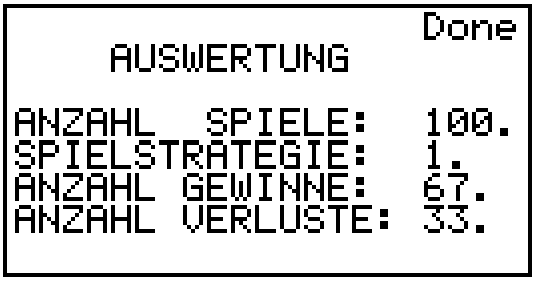
Der kumulierte Spielerfolg in 100 Spielverläufen:



...



Die Endauswertung nach N = 100 Spielen:



Damit zeigt die Simulation deutlich, dass die Strategie des Wechselns der gewählten Tür zu mehr Gewinnspielen führt als das Beharren auf der zuerst gewählten Tür.

Das Ziegenproblem und die Wahrscheinlichkeitsrechnung:

Ereignisse: A_i = { Auto befindet sich hinter der Tür i }, i = 1, 2, 3 .

M_j = { Moderator öffnet die Tür j }, j = 1, 2, 3 .

Wahrscheinlichkeiten: $P(A_i) = 1/3$ für alle i

Kandidat wählt die Tür 1 (o.B.d.A.)

Bedingte Wahrscheinlichkeiten:

$P(M_2 / A_1) = 1 / 2$ $\quad$ $P(M_3 / A_1) = 1 / 2$

$P(M_2 / A_2) = 0$ $\quad$ $P(M_3 / A_2) = 1$

$P(M_2 / A_3) = 1$ $\quad$ $P(M_3 / A_3) = 0$

Formel der totalen Wahrscheinlichkeit:

$$P(M_2) = P(M_2 / A_1) \times P(A_1) + P(M_2 / A_2) \times P(A_2) + P(M_2 / A_3) \times P(A_3)$$

$$= 1 / 6 + 0 + 1 / 3 = 1 / 2 ,$$

analog

$$P(M_3) = P(M_3 / A_1) \times P(A_1) + P(M_3 / A_2) \times P(A_2) + P(M_3 / A_3) \times P(A_3)$$

$$= 1 / 6 + 1 / 3 + 0 = 1 / 2 .$$

Bayes'sche Formel:

$$P(A_1 / M_2) = P(M_2 / A_1) \times P(A_1) / P(M_2) = (1 / 2) \times (1 / 3) / (1 / 2) = 1 / 3,$$

$$P(A_3 / M_2) = P(M_2 / A_3) \times P(A_3) / P(M_2) = (1) \times (1 / 3) / (1 / 2) = 2 / 3,$$

$$P(A_2 / M_2) = 0.$$