Zufall und Bytes

Semesterarbeit Mathematik

Tobias Tommila

Inhalt

Seite

3 Einleitung

4 Definitionen

4 Quellen

5 Mathematische Grundlagen
5 Modulo
6 Gleichverteilung

7 Research
7 Linear congruential generator
9 Lagged Fibonacci generator
9 Blum Blum Shub
9 Zufallszahl aus „Formeln und Tafeln“
9 Mersenne twister

10 TTgenerator
10 Der erste Schritt
10 Der zweite Schritt
11 Der dritte Schritt
11 Erste Tabellen zu TTgenerator
12 Der letzte Schritt

13 Schlusstext

14 Anhang
14 Mail vom Bundesamt
15 Tabelle mit Zahlenreihe von TTgenerator
16 Diagramm zu TTgenerator

Einleitung

Was ist Zufall? Gibt es Zufall? Schicksal? Chaos?

Abgesehen von diesen Fragen interessierte mich, wie man einer Maschine Zufall beibringen kann. Vielen Orten ist Zufall anzutreffen, so bei der zufälligen Wiedergabe von Lieder auf der Stereoanlage, in Spielen auf PC und Handy sowie natürlich im Casino.

Als erstes interessierte mich die Frage, wie die Spielautomaten im Casino funktionieren. Von Herstellern erhielt ich jedoch keine Informationen dazu.
Vom Bundesamt erhielt ich freundlicherweise ein Mail, in welchem ich über die Vorschriften für einen Spielautomaten aufgeklärt wurde. Das Mail beinhaltet ein paar interessante Aspekte, aus diesem Grund habe ich es im Anhang angehängt.

Als Ziel dieser Arbeit habe ich mir gesetzt, selber einen funktionierenden Zufallsgenerator für Würfelaugen (die Zahlen 1-6) zu programmieren. Um dieses Ziel zu erreichen, musste ich mich mit den mathematischen Anforderungen an eine Reihe von „Zufallszahlen“ auseinandersetzen.

Die folgenden Erkenntnisse, auf welche ich noch näher eingehen werde, waren für mich die wichtigsten: das Rechnen mit dem Restwert, die Gleichverteilung und die Tatsache, dass die nächste „Zufallszahl“ bei den einfacheren Algorithmen immer Bezug auf die letzten Zahlen nimmt. Für mich war selbstverständlich, dass eine möglichst hohe Periodizität erreicht werden muss, da es in der Natur wohl kaum eine grössere Reihe von Würfelaugen wiederholen wird. Ein weiteres Problem stellte der Initialwert (der erste Wert in einer Reihe von Zufallszahlen) dar. Denn mit einem vorprogrammierten Algorithmus und dem selben Initialwert gibt es immer und überall die selbe Reihe von Zufallszahlen.

Während dieser Arbeit bin ich auf etliche Themen der Mathematik gestossen. Ich werde mich aber auf die hierfür relevanten Themen beschränken.

Alle „Zufallsgeneratoren“, welche ich in diesem Dokument vorstelle, sind im Grunde genommen „pseudorandom number generators“, wie es auf Englisch so schön heisst. Der Einfachheit halber werde ich ab jetzt nur noch von Zufallszahlen schreiben, ohne pseudo oder Gänsefüsschen.

An dieser Stelle möchte ich Herr Staehli Markus, Jürg Tschirren und Jasmin Furrer herzlich für ihre Unterstützung danken.

Definitonen

Um die (von mir erstellten) Algorithmen und Bezeichnungen besser zu verstehen, definiere ich folgende Werte:

Z_n steht für eine Zufallszahl (an beliebiger Stelle in einer Reihe, eben an n-ter Stelle)
(in Excel-Tabellen steht „Zn“ für den selben Wert, Excel kennt den Formel-Editor nicht)

Z_I steht für den Initialwert (der erste Wert in einer Reihe von Zufallszahlen)

A_n steht für die Werte der Augenzahlen des Würfels
(Excel: An)

Quellen:

- www.wickipedia.com
- Formeln und Tafeln, Orell Füssli
- Bronstein Taschenbuch der Mathematik, Verlag Harri Deutsch

Mathematisch Grundlagen
Modulo

Quelle: www.wickipedia.com

Modul (mit Betonung auf der ersten Silbe) heißt die Zahl, deren Restklassen in einer Kongruenz betrachtet werden. Man sagt dann auch, die Kongruenz gelte **modulo** dieser Zahl; modulo ist dabei die lateinische Ablativform von modulus (Modul).

In der Mathematik und Informatik steht **Mod** für den Modulo-Operator bzw. die Modulo-Funktion. Sie liefert den Rest bei der Ganzzahl-Division.

So ist z. B. (5 mod 3) = 2, als Äquivalenz geschrieben:

$$5 \equiv 2 \bmod 3$$

oder kurz

$$5 \equiv 2 \quad (3)$$

(gesprochen: "5 ist kongruent zu 2 modulo 3").

In der Schule lernt man den Modus als **Rest einer Teilung**, wenn die Teilung nicht ganzzahlig aufgeht.

Beispiele

- 17 / 3 = 5,666666... = 5 Rest 2
- 17 / 4 = 4,25 = 4 Rest 1
- 9 / 2 = 4,5 = 4 Rest 1
- 2 / 3 = 0,666666... = 0 Rest 2 (Hinweis: 3 "passt null mal in 2", daher "bleiben 2 übrig" - der Rest ist also 2)

Das Rechnen mit Restwerten kommt in fast jedem Zufallsgenerator vor, da die Zahlen einer Reihe sonst immer grösser würden. So stiessen die meisten Computer aber schon bald mal an ihren Grenzen. Aus diesem Grund kann man die maximale Grösse der Zahlen durch modulo beschränken.

Weiterhin brauche ich modulo, um die grossen Zahlen einer Zufallsreihe auf meine Würfelzahlen herunter zu brechen. Der Rest wird für meine Augenzahl stehen.

Mathematisch Grundlagen
Gleichverteilung

Quelle: www.wickipedia.com

Diskrete Gleichverteilung

Die **diskrete Gleichverteilung** ist eine statistische Wahrscheinlichkeitsverteilung. Ein diskrete Zufallsvariable X mit endlich vielen Ausprägungen hat eine diskrete Gleichverteilung, wenn die Wahrscheinlichkeit für jede ihrer Realisationen xi (i=1, ... , n) gleich ist.

Ihre Wahrscheinlichkeitsfunktion ist

$$P(X=x)=f(x)=\begin{cases}\frac{1}{n} & \text{fuer } x=x_i(i=1,...,n)\\ 0 & \text{sonst}\end{cases}$$

Sie besitzt den Erwartungswert

$$EX=\frac{1}{n}\sum_{i=1}^{n}x_i$$

Typischerweise findet diese Wahrscheinlichkeitsverteilung Anwendung bei Zufallsexperimenten, deren Ergebnisse gleichhäufig sind.

Beispiel

Sechsseitiger Würfel

Das Zufallsexperiment ist: Ein Würfel wird einmal geworfen. Die möglichen Ausprägungen der Zufallsvariablen X: resultierende Augenzahl sind 1, 2, ..., 6. Nach der klassischen Wahrscheinlichkeitsauffassung ist die Wahrscheinlichkeit für jede Ausprägung gleich. Sie hat dann die Wahrscheinlichkeitsfunktion

$$P(X=x)=f(x)=\begin{cases}\frac{1}{6} & \text{fuer } x=x_i(i=1,...,6)\\ 0 & \text{sonst}\end{cases}$$

mit dem Erwartungswert

$$EX=\frac{1}{6}(1+2+...+6)=3,5$$

Wie durch Zufall ist auf Wickipedia der Würfel als Beispiel aufgeführt. Zusammengefasst heisst das oben aufgeführte, dass jede Augenzahl in einer zufälligen Reihe etwa gleich oft vorkommen muss. Dies garantiert, dass jede Zahl bei jedem „Wurf“ die gleich hohe Chance hat vorzukommen.

Research
Linear congruential generator

$Z_n = A \times Z_{n-1} + B \pmod M$

Z steht für die Zufallszahl (Z_n ist die neue Zahl, Z_{n-1} steht für die letzte Zufallszahl)
A & B stehen für Konstanten, das heisst, eine beliebige Zahl, welche aber immer gleich bleibt
M jeder Wert in der Reihe wird kleiner als M (modulo)

Was passiert:
Um eine weitere Zufallszahl zu erhalten wird die letzte Zahl (oder der Initialwert) mit *A* multipliziert und *B* dazugezählt.

Dieser sehr einfache Zufallsgenerator wird in Apparaten mit geringer Rechenleistung (z.B. Handys) eingesetzt. Zuerst machte ich den Fehler, dass ich für M meine 6 eingesetzt habe. So erhielt ich eine Augenzahlenreihe mit einer Periodizität von höchstens 4 Zahlen. Wichtig ist, dass man für die Zufallszahl einen möglichst hohen Restwert einsetzt, diese grossen Zahlen dann erst auf die Augenzahlen herunterrechnet (siehe Tabelle auf folgender Seite).

Beispiel:

Die Zahlen für A, B und M habe ich aus wickipedia.com und sind sicherlich auf eine hohe Periodizität optimiert.

Z_I	A	B	M
13	1664525	1013904223	2^{32}

Die ersten 40 Werte der Reihe sind in der Tabelle auf der folgenden Seite zu sehen.

Zu der Augenzahl:
Die Zahlen für Z_n waren zu gross für Excel (konnte die Augenzahl nicht berechnen. Aus diesem Grund musste ich die Zahlen zuerst mit mod 60'000 (ein vielfaches von 6, sollte darum die Augenzahl nicht beeinflussen).

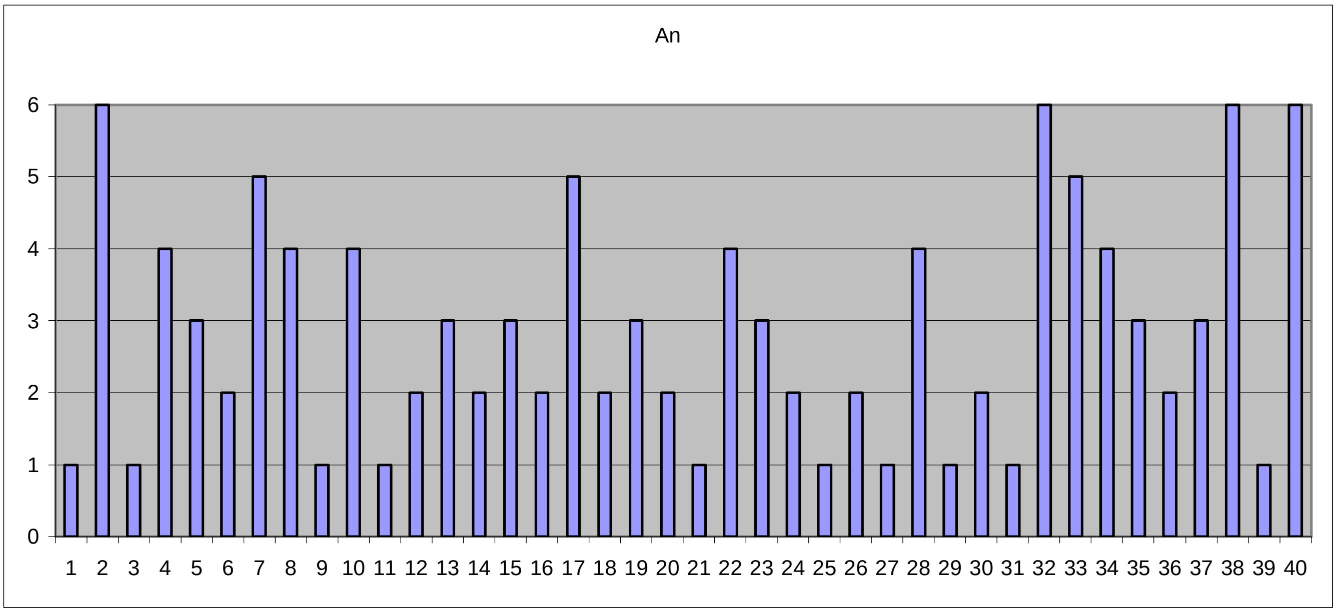
Eine weitere Möglichkeit die Augenzahl (A_n) zu berechnen, wäre folgende gewesen:
M ist 2^{32}, aus diesem Grund wird keine Zufallszahl (Z_n) grösser als 2^{32} sein. Wenn ich jetzt Z_n geteilt durch 2^{32} rechne, muss es eine Reihe von Zahlen zwischen 0 und 1 geben. Da ich aber Zahlen von 0 bis 5 möchte, multipliziere ich mit 5. Das sieht dann mathematisch ausgedrückt so aus:

$$A_n = \frac{Z_n \times 5}{2^{32}}$$

Die Reihe gäbe aber keine ganzen Zahlen, diese müssten darum gerundet werden.

(Algorithmus Zn: siehe vorherige Seite)

	Zufallsreihe			**Augenzahl A_n**	
n	**Zn**	**Zn mod 60'000 = Zn2**		**(Zn2 mod 6)+1**	**(Zn*6)/2^32**
1	1035543048	3048		1	1.45
2	1965874631	34631		6	2.75
3	3095560314	40314		1	4.32
4	640292241	32241		4	0.89
5	206754236	54236		3	0.29
6	1469088235	48235		2	2.05
7	4068224590	44590		5	5.68
8	2362335573	15573		4	3.30
9	520012464	52464		1	0.73
10	1411446351	6351		4	1.97
11	2985684834	24834		1	4.17
12	2749310809	50809		2	3.84
13	1344432356	12356		3	1.88
14	1111301875	41875		2	1.55
15	4187576246	56246		3	5.85
16	2465262493	42493		2	3.44
17	3001055320	35320		5	4.19
18	2187337687	37687		2	3.06
19	1140800330	20330		3	1.59
20	742289953	29953		2	1.04
21	2186077452	37452		1	3.05
22	3687144699	24699		4	5.15
23	486780446	446		3	0.68
24	770490085	30085		2	1.08
25	2313216768	36768		1	3.23
26	38566495	46495		2	0.05
27	3327788082	8082		1	4.65
28	2509182441	42441		4	3.51
29	4214154804	54804		1	5.89
30	681401347	41347		2	0.95
31	4217426310	26310		1	5.89
32	166394669	14669		6	0.23
33	3839271592	51592		5	5.36
34	1116480999	999		4	1.56
35	674621978	41978		3	0.94
36	2667328177	28177		2	3.73
37	3199798364	58364		3	4.47
38	3686611979	31979		6	5.15
39	2819252718	32718		1	3.94
40	2016985205	25205		6	2.82



Research

weitere Zufallsgeneratoren

Um nicht den Rahmen dieser Arbeit zu sprengen, werde ich zu den folgenden Algorithmen keine Tabellen und Grafiken mehr aufzeigen. Ich habe mit allen Algorithmen gerechnet, die Resultate kamen bei allen gut heraus.

Einige weitere Zufallsgeneratoren:

Lagged Fibonacci generator

$$Z_n = Z_{n-1} + Z_{n-2} \pmod{M}$$

Hier werden ganz einfach die zwei letzten Werte miteinander addiert, ergeben so die neue Zufallszahl. Bei einer anderen Version werden diese beiden Zahlen miteinander multipliziert um die neue Zufallszahl zu erhalten. Auf diese Weise kann die Periodizität verlängert werden, da nicht nur die letzte, sonder die zwei letzten Werte genau die selben sein müssen, dass sich die Reihe wiederholt.

Blum Blum Shub

$$Z_n = (Z_{n-1})^2 \pmod{M}$$

Bei diesem Generator wird einfach die letzte Zahl quadriert um eine weitere Zufallszahl zu erhalten.
Es wird empfohlen für M ein Produkt aus zwei grossen Primzahlen zu wählen. Wieso das so ist, übersteigt meine Vorstellungskraft. Dem nachzugehen wäre wohl eine Semesterabeit für sich...

Zufallszahl aus „Formeln und Tafeln“

$$X_I = 94664704$$
$$X_n = 23 \times X_{n-1} \text{ modulo } 10^8 + 1$$
$$Z_n = X_n \text{ modulo } 10^5$$

Hier werden zwei Reihen erstellt, zuerst eine mit modulo $10^8 + 1$, dann eine mit modulo 10^5.
Z_n ist immer Abhängig vom entsprechenden Wert für X_n.

Mersenne twister

Dieser Generator hat bewiesenermassen eine Periodizität von $2^{19937} - 1$ und rechnet in 623 Dimensionen. Ich glaube da erübriget sich jeglicher Kommentar von einem angehenden Schreiner-Techniker...

TTgenerator

Endlich zu meinem Ziel dieser Arbeit, meinem eigenen Zufallsgenerator.
Als Grundlage wollte ich die mystische Zahl p nehmen. Die Kommastellen dieser Zahl haben ja keine Periodizität, diese Tatsache versuchte ich in meinem Zufallsgenerator zu nutzen.

Der erste Schritt:

$$Z_n = p \times Z_{n-1} \times 10 \quad \text{modulo } 10^8 + 1$$

Um die Augenzahl zu bestimmen, rechnete ich mit modulo 6. So dachte ich mir, dass ich Zahlen zwischen 0 und 5 erhalte, diesen dann 1 addiere, auf diese Weise zu meinen Augenzahlen zwischen 1 und 6 käme.

Ich rechnete eine Reihe von 180 Zahlen und fing an von Hand die Zahlen zu Zählen (ich wusste noch nicht, wie die Funktion „zählenwenn“ im Excel funktioniert), so wollte ich die Gleichverteilung der Reihe überprüfen. Dies machte ich mit zwei verschiedenen Initialwerten. Mir fiel auf, dass die Null und die Fünf nur etwa halb mal so oft vor kamen wie die restlichen Zahlen.
Dies lag natürlich am Runden. Die Null hat ja nur ein Spektrum von 0 bis 0.499..., die Fünf nur von 4.5 bis 5. Die restlichen Zahlen haben ein Spektrum von einer ganzen Zahl (z.B. von 2.5 bis 3.499... gibt eine 3 als Augenzahl).
Aus diesem Grund musste ich ab jetzt mit modulo 7 rechnen. Ein Computer müsste in einem Würfelspiel die Zahlen 0 und 7 in der Zufallsreihe einfach übergehen.

Der zweite Schritt:

Ich rechnete mit verschiedenen modulo und Initialwerten mehrere Reihen mit 240 Zahlen. Wieder zählte ich die Anzahl der verschiedenen Augenzahlen. Erstaunlicherweise kamen sich die Anzahlen immer ziemlich nahe (wenn man die Anzahl der Nullen und Sieben addierte, bekam man in etwa die selbe Anzahl wie von den restlichen Zahlen).
Jetzt kam mir eine weitere Möglichkeit in den Sinn, die Brauchbarkeit der Reihe zu beweisen: der Erwartungswert (siehe Seite 6). Zu meiner Überraschung waren die Werte aller Reihen nahe bei 3.5. Ich hatte aber noch immer eine kleine Reihe und am Initialwert konnte man auch noch arbeiten. Um die Periodizität eventuell etwas zu verlängern, entschied ich mich fürs Fibonacci-Modell, das heisst, ich wollte mit den zwei letzten Zufallswerten die neue Zufallszahl generieren.

Der dritte Schritt:

$$Z_n = p \times (Z_{n-1} + Z_{n-2}) \quad \text{mod } 13^5 + 1$$

$Z_I = jetzt$ „jetzt“ bedeutet im Excel soviel wie das heutige Datum in Sekunden oder sogar einem Bruchteil von Sekunden.
Da der Algorithmus zwei Initialwerte benötigt, ist der zweite Wert „jetzt+1“, was im Excel so viel wie die selbe Zeit am nächsten Tag bedeutet.

M diesen Wert haben ich zufällig gewählt

Diesen Algorithmus habe ich mit einer Reihe von 15’000 Zahlen zu verschiedenen Zeitpunkten getestet. Mit mod 7 habe ich die grossen Zufallszahlen auf die Augenzahlen herunter gerechnet. Die Resultate sind aus den folgenden Tabellen ersichtlich. Mit diesen wäre wohl die Gleichverteilung meines Generators bewiesen.
In der linken Spalte der Tabelle sind die Augenzahlen, rechts ist die Anzahl der Augenzahlen ersichtlich, E(Z) steht für den Erwartungswert:

12.04.2005 21:31	Anzahl
0	1059
1	**2120**
2	**2111**
3	**2173**
4	**2210**
5	**2086**
6	**2150**
7	1091
E(Z) 3.511	Summe 15000

12.04.2005 21:33	Anzahl
0	1071
1	**2169**
2	**2150**
3	**2159**
4	**2197**
5	**2085**
6	**2129**
7	1040
E(Z) 3.482	Summe 15000

12.04.2005 21:36	Anzahl
0	1058
1	**2140**
2	**2156**
3	**2086**
4	**2155**
5	**2184**
6	**2144**
7	1077
E(Z) 3.509	Summe 15000

12.04.2005 21:38	Anzahl
0	1037
1	**2131**
2	**2131**
3	**2138**
4	**2158**
5	**2144**
6	**2155**
7	1106
E(Z) 3.521	Summe 15000

Der letzte Schritt

Jetzt ging es an die Beseitigung der Augenzahlen 0 und 7, welche ziemlich störend in meiner Reihe waren.

Für dies waren vor allem zwei Erkenntnisse wichtig (Dank an Fränzi Stähli):

- solange man nur mit Ganzzahlen rechnet, muss es auch einen ganzzahligen Rest geben.
- man kann im Excel den Befehl „Ganzzahl“ einsetzen. Mit dieser Funktion konnte ich die lästigen Kommastellen (wegen p) eliminieren.

Dank diesen Tatsachen funktioniert nun die Formel (welche ich im „ersten Schritt“ anwenden wollte):

$A_n = (Z_n \bmod 6) + 1$ (ähnlich wie die Notenberechnung in der Schule)

Es gibt jetzt keine gerundeten Zahlen mehr, das Diagram sieht so auch viel schöner aus. Das beste ist natürlich, dass so keine unnötigen Zahlen (die 0 und 7) in der Reihe der Augenzahlen vorkommen.

Die Zahlenreihen sind noch immer gleichverteilt, wie die folgenden beiden Tabellen zeigen:

20.04.2005 19:47	Anzahl
1	**2453**
2	**2578**
3	**2508**
4	**2538**
5	**2463**
6	**2460**
E(z)	Summe
3.491	15000

20.04.2005 19:49	Anzahl
1	**2508**
2	**2483**
3	**2491**
4	**2535**
5	**2516**
6	**2467**
E(z)	Summe
3.498	15000

Im Anhang hat es eine Tabelle und ein Diagramm zu den Reihen der Augenzahlen.
Im Anhang hat es auch eine CD-Rom mit einer Excel-Datei vom TTgenerator. Als Initialwert benutzt dieser „jetzt“ und das Geburtsdatum des Benutzers (Makros aktivieren!).

Schlusstext

Ich glaube, das Ziel meiner Arbeit habe ich erreicht. Für mich haben sich aber mehr neue Fragen gestellt, als dass Fragen beantwortet wurden.
Wie könnte ich zum Beispiel die Periodizität meines TTgenerators überprüfen? Für dies wäre es wohl interessant, Kontakt mit der Prüfstelle des Bundes aufzunehmen (welche die Casinotauglichkeit von Zufallsgeneratoren überprüft).
Im Anhang habe ich einen Ausschnitt aus der Reihe der Augenzahlen und ein Diagramm, gerechnet mit dem TTgenerator am 20.4.2005 um 19:51 Uhr.
Es scheint, als gäbe es, egal mit welchen Initialwerten, immer eine ziemlich hohe Periodizität (über 1000 Werte). Natürlich kann ich das nicht beweisen, aber während meiner arbeit habe ich nie eine Periodizität erkennen können. Dies ist ziemlich faszinierend, denn beim „linear congruential generator" ist die Periodizität sehr abhängig von den Werten für die verschiedenen Variablen. Mir gab es einige Male eine ziemlich kurze Periode (für die Augenzahlen, obwohl ich diese von den grossen Zahlen der Reihe herunter gerechnet habe) mit diesem Generator. Wieso das so ist, weiss ich halt nicht so genau.
Mit p habe ich eine Zahl gefunden, welche auf der ganzen Welt definiert ist, bis ins unedliche nach dem Komma. So könnte der TTgenerator in verschiedenen Geräten, mit verschiedenen Initialwerten eingesetzt werden.

Jedenfalls habe ich viel gelernt und die Sache ist für mich sicherlich nicht abgeschlossen. Gibt fast jeder Algorithmus eine Gleichverteilte Reihe von Zahlen? Eben, viele Fragen sind offen...

Ich könnte mir auch vorstellen, ein kleines Spiel auf Excel zu programmieren, das Geburtsdatum des Spielers als erster, das jetzige Datum als zweiter Initialwert. So könnten Spieler mit ihren „Würfelreihen" gegeneinander spielen.

Zum Glück weiss ich jetzt zumindest, wie man einem Computer „Zufall" beibringen kann.
Was mir aber, vor lauter Mathematik, vergessen ging: selber mal einen Würfel in die Hand nehmen und eine Reihe von etwa 300 Zahlen zu würfeln...

Mail vom 11.3.2005:

Sehr geehrter Herr Tommilia

Zu Ihrer Anfrage vom 9. März 05 bezüglich Zufallsgeneratoren in Glücksspielautomaten kann ich Ihnen Folgendes mitteilen:

1. Vorgeschriebene Anzahl Gewinne

Gemäss Glücksspielverordnung Art. 28 (siehe www.esbk.admin.ch: Regulierung/Verordnungen) muss ein Automat eine theoretische Auszahlungsquote von mindesten 80 Prozent aufweisen.

Anmerkung: Die Glücksspielautomaten in den Casinos weisen eine Auszahlungsquote von ca. 88%-98% auf. Diese hohen Werte müssen jedoch korrekt interpretiert werden. Wenn Sie zum Beispiel von einem Einsatz von CHF 100.- mit einer Auszahlungsquote von 95% ausgehen und den erzielten Gewinn von 95% wiederum riskieren, bleibt Ihnen nach 10 Spielen theoretisch lediglich ein Betrag von ca. CHF 60.- übrig!

2. Wählen die Automaten wirklich zufällig?

Ja. Die Zufallsgeneratoren generieren absolut zufällige Werte. Der Initialwert für die Zufallfunktion ist ebenfalls variabel (z.B. f(Uhrzeit). Die programmierten Zufallsgeneratoren werden durch unabhängige Prüflabors getestet (siehe www.esbk.admin.ch: Spielbankenaufsicht/Technische Aufsicht/ List of Certification Institutes / Zertification Protocol Switzerland)

Anmerkung: In einem Glücksspielautomaten werden die Gewinne mittels Zufallsgenerator und den so genannten Paytabels generiert. Das folgende Beispiel veranschaulicht die Situation:

Ein Zufallsgenerator generiert z.B. Zufallszahlen von 1 bis 10
Eine Paytable, die die Gewinnhöhe definiert (Matrix mit 10 Elementen) enthält folgende 10 Werte: [0, 0, 0, 5, 0, 4, 0, 0, 0, 0]

Dies ergibt einen Gewinn, wenn die Zufallszahl 4 oder 6 generiert wird. Im vorliegenden Beispiel wird somit in 20 % der Fälle ein Gewinn mit einer Auszahlungsquote von 90% realisiert.

Falls Sie weitere Fragen zu diesem Thema haben, können Sie diese auch direkt mir zustellen.

Mit freundlichen Grüssen

G. Bertschy
ESBK

guido.ertschy@esbk.admin.ch

TTgenerator

Ausschnitt aus der Zufallsreihe vom 20.4.2005 um 19:51 Uhr generiert:

n	Zn	„Ganzzahl" Zn	An
1	241672.2221	241672	5
2	137485.3595	137485	2
3	77276.67307	77276	3
4	303400.8238	303400	5
5	82051.62762	82051	2
6	97052.58978	97052	3
7	191378.4936	191378	3
8	163544.9726	163544	3
9	1142.95418	1142	3
10	146088.381	146088	1
11	91246.88112	91246	5
12	3022.716075	3022	5
13	296156.674	296156	3
14	197311.774	197311	2
15	65100.85116	65100	1
16	81805.57548	81805	2
17	90226.1507	90226	5
18	169159.6072	169159	2
19	72296.39138	72296	3
20	15968.39121	15968	3
21	277291.9926	277291	2
22	178716.6673	178716	1
23	318711.4558	318711	4
24	77540.53701	77540	3
25	130980.3494	130980	1
26	283793.685	283793	6
27	189169.0596	189169	2
28	680.2837646	680	3
29	225135.3023	225135	4
30	338126.5861	338126	3
31	284363.4107	284363	6
32	99140.00088	99140	3
33	90929.50032	90929	6
34	225826.9486	225826	5
35	252531.7331	252531	4
36	17632.12024	17632	5
37	106156.7768	106156	5
38	17600.28953	17600	3
39	17500.29042	17500	5
40	110271.7241	110271	4
41	30113.62214	30113	6
42	69739.57241	69739	2
43	313698.0624	313698	1
44	90722.85665	90722	3
45	156643.7883	156643	2
46	34537.23452	34537	2
47	229318.8968	229318	5
48	86340.48376	86340	1
49	249085.191	249085	2
50	311182.8357	311182	5

10106	70375.84338	70375	2
10107	162051.5985	162051	4
10108	358898.3438	358898	3
10109	151436.5113	151436	3
10110	118088.2316	118088	3
10111	104148.9524	104148	1
10112	326884.7046	326884	5
10113	240250.1703	240250	5
10114	296530.7568	296530	5
10115	201171.0172	201171	4
10116	78400.23698	78400	5
10117	135710.9984	135710	3
10118	301356.2843	301356	1
10119	259205.3645	259205	6
10120	275880.3577	275880	1
10121	195845.3741	195845	6
10122	368088.0936	368088	1
10123	286473.2393	286473	4
10124	199895.0749	199895	6
10125	42795.12271	42795	4
10126	19845.74177	19845	4
10127	196792.0797	196792	5
10128	309293.7883	309293	6
10129	104739.6448	104739	4
10130	186842.3916	186842	3
10131	173443.9832	173443	2
10132	17991.02808	17991	4
10133	230116.825	230116	5
10134	36865.80862	36865	2
10135	96162.68052	96162	1
10136	46627.32418	46627	2
10137	77294.02977	77294	3
10138	18016.41521	18016	5
10139	299426.5937	299426	3
10140	254688.6249	254688	1
10141	255628.3	255628	5
10142	118031.9023	118031	6
10143	60006.14652	60006	1
10144	188029.0262	188029	2
10145	36637.47633	36637	2
10146	334516.6337	334516	5
10147	52133.02559	52133	6
10148	100813.7293	100813	2
10149	109202.4015	109202	3
10150	288491.1337	288491	6
10151	135509.0885	135509	6
10152	218153.9829	218153	6
10153	368477.3069	368477	6
10154	357780.5506	357780	1

Ausschnitte aus Augenzahlreihen, generiert mit dem TTgenerator: