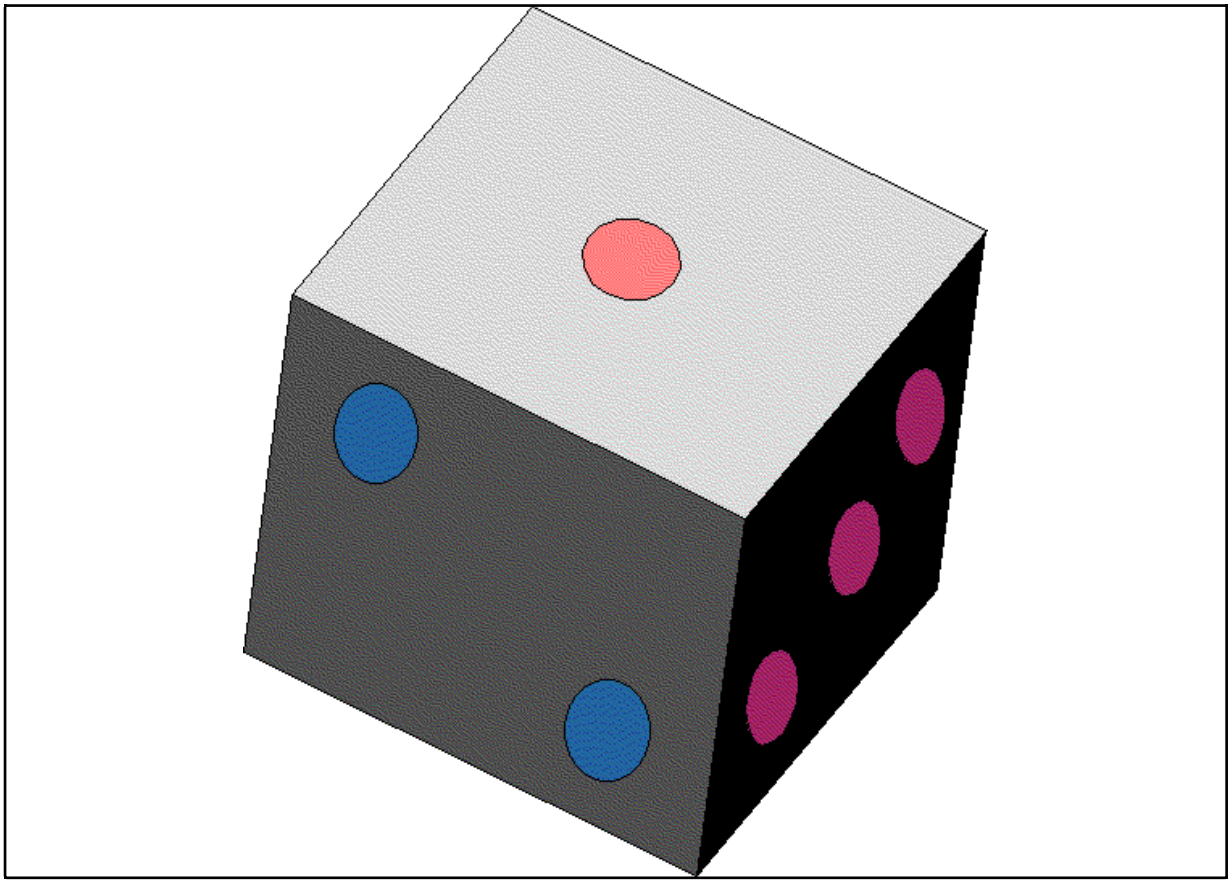
Zufallszahlen erzeugen



Thema: Zufallszahlen erzeugen - Methode der linearen Kongruenz

Schultyp: Mittelschule, technische Berufsschule, Fachhochschule

Vorkenntnisse: Kenntnisse in der Arbeit mit einem Tabellenkalkulations-Programm, Posten „Ganzzahlige Division und Modulo-Funktion“ der Werkstatt „Multiplikation“

Bearbeitungsdauer: 45 - 60 Minuten

Fassung vom: 10. Oktober 1995

Schulerprobung: nein

Übersicht

Ein Computer ist eine deterministische Maschine. Wie kann man trotzdem Zahlensequenzen erzeugen, welche wichtige Kriterien erfüllen, die an zufällig aufeinander folgende Zahlen gestellt werden.

Behandelt wird das Verfahren der linearen Kongruenz, das von D. Lehmer vorgeschlagen und später von D. Knuth verfeinert wurde. Zuerst studieren die Schülerinnen und Schüler die Theorie und erzeugen anschliessend mit einem Tabellenkalkulations-Programm eine einfache Folge von Zufallszahlen. Sie lernen auch die Grundzüge der statistischen Tests kennen, mit denen die Eigenschaften von Folgen von Pseudo-Zufallszahlen untersucht werden können.

Lernziele

Zufallszahlen werden in Simulationsprogrammen, in statistischen Umfragen, in Spielen und weiteren Problemstellungen verwendet. Anhand der folgenden Aufgaben kann den Schülerinnen und Schülern gezeigt werden, wie stark ein Computer ein deterministisches Instrument ist und wie trotzdem Zahlenfolgen erzeugt werden können, die wichtige Eigenschaften von Zufallszahlen aufweisen.

Nach Bearbeiten des Postens kennen die Schülerinnen und Schüler die Methode der linearen Kongruenz als Vertreter einer ganzen Reihe von Algorithmen zum Erzeugen von Pseudo-zufälligen Zahlenfolgen. Sie haben auch erfahren, dass die statistische Qualität der Zahlen, welche diese Algorithmen liefern, stark von den Einstellungen, die man den Parametern des Algorithmus gibt, abhängt.

Ausserdem haben sie an einem einfachen Beispiel erprobt, wozu man Zufallszahlen, die ein Computer erzeugt, einsetzen kann.

Material

- *Theorie*: Zufallszahlen-Generator - Die Methode der linearen Kongruenz
- *Computer mit einem Tabellenkalkulationsprogramm*

Quellen

Sedgewick, Robert: *Algorithmen.* Deutschland 1991 (Addison-Wesley)

Knuth, Donald: *The Art of Computer Programming, Volume 2 / Seminumerical Algorithms.* USA 1981 (Addison-Wesley),

Hinweise, Lösungen

Lösung für die Tabellenkalkulationsaufgabe

Das Tabellenkalkulations-Arbeitsblatt sollte im wesentlichen wie folgt aussehen:

	A	B	C	D	E
1	**Methode der linearen Kongruenz zur**				
2	**Erzeugung von Pseudo-Zufallszahlen**				
3					
4	**Parameter**				
5	Divisor m:	64	Multiplikator a:	9	
6					
7	**Erzeugte Folge**				
8	Startwert X_0:	21			
9	Erster Wert X_1	= REST((B8*D5+1); B5)			
10	…	= REST((B9*D5+1); B5)			
11		= REST((B10*D5+1); B5)			
12		= REST((B11*D5+1); B5)			
13		= REST((B12*D5+1); B5)			

In dieser Lösung wird die Version mit der EXCEL-Funktion REST zur Berechnung der Modulo-Funktion gezeigt. Diese ist wie folgt definiert:

REST (Zahl; Divisor)

Die Formel wird sinngemäss über ca. 100 bis 200 Zeilen nach unten kopiert.

Frage 1 Länge der Folge

Die Folge wiederholt sich nach 64 Zeilen.

Das entspricht der erwarteten Länge der Folge (m = 64).

Frage 2 Ungünstige Parameter

Die Länge der Folge beträgt weiterhin 64 (m = 64).

Hingegen ist die Folge trotzdem keine Pseudo-Zufallsfolge, weil es sich um eine Reihe stetig wachsender Zahlen handelt.

Lehrer-Lernkontrolle / Test

Aufgabe 1

Sie kennen die *Methode der linearen* Kongruenz, die in Computerprogrammen zum Erzeugen von Folgen von Pseudo-Zufallszahlen gebraucht wird. Die Methode besteht in der wiederholten Anwendung der Formel (1), die Sie im Theorieteil kennengelernt haben.

Können Sie sagen, wieviele verschiedene Zahlen diese Formel für einen gegebenen Divisor m maximal liefern kann? Welches ist die kleinste und welches die grösste Zahl, die theoretisch auftreten kann? Begründen Sie Ihre Antwort.

Aufgabe 2

Nehmen Sie an, dass Sie einen Zufallszahlen-Generator nach der Methode der linearen Kongruenz erstellt haben, welcher ganze Pseudo-Zufallszahlen im Bereich <0 .. 32767> liefert. Nun wollen Sie diesen Generator so verändern, dass er Zahlen zwischen <1 .. 6> liefert. Jede Zahl der Folge kann also als Wurf eines Würfels aufgefasst werden. Wie können Sie dieses Ziel erreichen?

Lösungen und Taxierung

Aufgabe 1

Grösste Zahl ist m-1, kleinste Zahl ist 0. Maximal werden m verschiedene Zahlen geliefert.

Begründung: Die Modulo-Funktion in Formel (1) bildet alle Werte auf den Bereich <0 .. m-1> ab.

Hier geht es darum, Stoff der in der Theorie behandelt wurde, nochmals mit anderen Werten anzuwenden. Die Taxierung ist deshalb K2.

Aufgabe 2

Die Zufallszahlen, die der Generator liefert, werden durch die Konstante m (m = 32768) dividiert und mit der Konstante 6 multipliziert. Vom Resultat werden die Stellen nach dem Komma abgeschnitten. Dadurch werden die Zahlenwerte, welche der Zufallszahlen-Generator liefert, auf die ganzen Zahlen 0 bis 5 abgebildet. Addiert man nun noch die Konstante 1 zu den Zahlen, erhält man Zahlen, die im gewünschten Wertebereich liegen.

Hier muss das Gelernte um eine Überlegung erweitert werden, die nicht im Stoff behandelt wurde. Die Aufgabe ist als K3 zu taxieren.

Was soll ich hier tun?

Ein Computer ist eine Maschine, die nach strengen Regeln arbeitet. Es gibt keine Abläufe, die vom Zufall bestimmt sind.

Nun gibt es aber Aufgabenstellungen, die man mit einem Computer bearbeiten möchte, zu deren Bewältigung man Zufallszahlen braucht. Solche Aufgaben treten zum Beispiel dann auf, wenn man eine statistische Umfrage plant und dazu eine zufällige Stichprobe aus einer grossen Anzahl von Individuen zusammenstellen muss. Weil ein Computer nur nach strengen Regeln arbeitet, muss man spezielle Verfahren anwenden, welche zwar keine Zufallszahlen aber doch sogenannte Pseudo-Zufallszahlen erzeugen.

An diesem Posten lernen Sie ein Verfahren kennen, das Folgen von Pseudo-Zufallszahlen erzeugen kann. Es wird auch gezeigt, wie die statistische Qualität der Folge von Zufallszahlen überprüft werden können. Der Posten besteht aus den folgenden Teilaufgaben:

(1) Zuerst wird das Verfahren von D. Lehmer erklärt, nach dem die meisten praktisch implementierten Zufallszahlen-Generatoren arbeiten. Dieser Theorieteil besteht aus einer Lektüre von 10 - 15 Minuten.

(2) Anschliessend untersuchen Sie das Verfahren von D. Lehmer mit einem Tabellen-Kalkulations-Programm. Sie berechnen eine kurze Folge von Zufallszahlen und kontrollieren die statistische Güte der Folge. Die genaue Aufgabenstellung finden Sie auf den Blättern mit dem Titel "Praktische Aufgabe". Sie sollten dafür etwa 30 - 45 Minuten brauchen.

Zufallszahlen-Generator: Die Methode der linearen Kongruenz

Computer sind Maschinen, die nichts dem Zufall überlassen. Sie funktionieren ohne Launen immer genau gleich, berechnen beliebig oft aus denselben Rohdaten genau die gleichen Resultate, ziehen aus bestimmten Prämissen immer zuverlässig dieselben Schlüsse. Deshalb bezeichnet man Computer als deterministische Maschinen. Die Beobachtung, wonach Personal Computer oft völlig überraschend mitten aus einer noch nicht gespeicherten Arbeit abstürzen, widerspricht dieser Aussage nicht. Solche unerfreulichen Ereignisse sind ebenfalls genau vorhersehbare Reaktionen der Maschine auf Fehler, die in den ausgeführten Programmen versteckt sind.

Wie kann man eine solche deterministische Maschine dazu bringen, zufällige Resultate zu erzeugen? Könnte ein Computer zum Beispiel als Würfel herhalten und in dieser Eigenschaft eine völlig zufällige Folge von Zahlen zwischen eins und sechs produzieren? Die vorweggenommene Antwort lautet, er kann es streng genommen nicht. Das Erzeugen von zufälligen Folgen von Zahlen ist aber eine derart wichtige Aufgabe, dass Lösungen gesucht wurden, Folgen von Zahlen zu erzeugen, die möglichst viele Eigenschaften einer zufälligen Folge aufweisen. Soche Folgen heissen pseudo-zufällige Folgen.

Ein Zufallszahlen-Generator

In seinem Buch „The Art of Computer Programming“ hat Donald Knuth viele Grundlagen zusammengetragen, auf denen die Berechnungsmethoden der heutigen Computerprogramme aufbauen. Er behandelt darin auch mehrere Vorschläge, wie pseudo-zufällige Zahlenfolgen erzeugt werden können. Auf diesen Vorschlägen basieren die Zufallszahlen-Generatoren, die in Programmiersprachen, Tabellenkalkulations-Programmen und anderer Software eingesetzt werden.

Am besten hat Knuth die *Methode der linearen Kongruenz* untersucht, die er von D. H. Lehmer übernimmt. Diese Methode wird an diesem Posten behandelt.

Grundidee

Die Methode berechnet die nächste Zahl einer Zahlenfolge mit dieser Formel aus der zuletzt berechneten Zahl:

$$X_{i+1} = (a * X_i + 1) \bmod m \qquad (1)$$

Die Aufgabe des in Klammern stehenden Terms

$$(a * X_i + 1)$$

ist es, aus der letzten berechneten Zahl X_i eine neue Zahl zu berechnen, die wegen der Multiplikation mit a eine grosse Zahl sein wird. Damit die Folge nicht abbricht, wenn $X_i = 0$ ist, wird immer die Konstante Eins zum Produkt addiert.

Anschliessend wird der Klammerterm ganzzahlig durch die Konstante m geteilt und der Rest der Division berechnet. Dazu dient die Modulo-Funktion mod. Das hat einen ähnlichen Effekt,

wie wenn man eine Roulettekugel in die rasch drehende Schüssel wirft. Die Schüssel dreht sich mehrere Male unter der Kugel durch, bis die Kugel schliesslich in eines der Löcher fällt.

Einfluss der Konstanten a und m

Die Formel (1) liefert nicht automatisch Zahlenfolgen, die als zufällig bezeichnet werden können. Was sie schliesslich liefert, hängt von den Werten ab, die den Konstanten a und m gegeben werden. Weiter muss festgelegt werden, mit welchem X_0 die Folge beginnen soll.

Probieren wir die Formel (1) mit den folgenden Werten aus:

a = 19

m = 381

X0 = 0

Dann kommt die Zahlenfolge

0, 1, 20, 0, 1, 20, 0, 1, 20, ...

heraus. Es werden nur drei Werte geliefert, was nicht als zufällige Folge von Zahlen bezeichnet werden kann. Der Wertebereich, der sich aufgrund der Modulo-Funktion von 0 bis 380 erstrecken sollte, wird praktisch nicht ausgenutzt.

Was kann vorgekehrt werden, um bessere Folgen zu erhalten?

Knuth hat diese Fragen theoretisch und praktisch untersucht. Hier werden einige vereinfachte Regeln angegeben, die sich aus seinen Erkenntnissen herleiten lassen.

Divisor m: Diese Konstante bestimmt den maximalen Wertebereich, den die Zahlen in der Folge annehmen. Setzt man m = 64, erhält man Zahlen, die maximal zwischen 0 und 63 liegen. Das ergibt sich aus der Modulo-Funktion in Formel (1). Für optimale Zufallszahlen-Generatoren bestimmt man m aus der Anzahl Bits, mit der die Maschine, auf der der Generator eingesetzt werden soll, ganze Zahlen darstellt.

Für grosse Werte von m kann man nicht sicher sein, dass wirklich alle Werte zwischen 0 und m-1 in der erzeugten Folge vorkommen.

Muliplikator a: Empfohlen wird ein Wert, der um eine Grössenordnung kleiner ist als m. Er soll also eine Stelle weniger aufweisen. Für m = 64 ergibt diese Regel eine Zahl zwischen 1 und 9. Wenn m vier oder mehr Stellen hat (also grösser 1000 ist), wird empfohlen, a mit den Ziffern 21 aufhören zu lassen.

Startwert $X_{0:}$ Für diesen Wert gibt es keine Regeln. Verschiedene Anfangswerte können jedoch zu verschiedenen Folgen von Zufallszahlen führen.

Test der Zufallsfolgen

Die Formel (1) liefert also eine Folge von ganzen Zahlen. Damit diese Folge als pseudo-zufällige Folge gelten kann, muss sie gewisse Bedingungen erfüllen. Zur Überprüfung dieser Bedingungen werden statistische Verfahren angewendet. An diesem Posten werden aber nur die grundsätzlichen Ideen, die hinter den Tests stehen, behandelt.

Test der Gleichverteilung

Dieser Test untersucht, ob die Pseudo-Zufallszahlen, die erzeugt werden, gleichmässig über das Intervall <0 .. m-1> verteilt sind. Es ist leicht einzusehen, dass dies eine erste wichtige Forderung ist. Die Pseudo-Zufallszahlen dürfen nicht an einer bestimmten Stelle im Wertebereich häufiger auftreten als an einer anderen.

Allerdings ist dieser Test nicht ausreichend. Wenn der Zufallszahlen-Generatorbeispielsweise die Folge <0, 1, 2, 3, ... , m-1> liefert, sind diese Zahlen sicher absolut gleichmässig über das Intervall verteilt, da jede Zahl genau einmal vorkommt. Trotzdem ist diese Folge natürlich keine pseudo-zufällige Folge.

Hier müssen andere Tests eingesetzt werden, auf die hier aber nicht näher eingegangen wird.

Literatur

Sedgewick, Robert: *Algorithmen.* Deutschland 1991 (Addison-Wesley)

Knuth, Donald: *The Art of Computer Programming, Volume 2 / Seminumerical Algorithms.* USA 1981 (Addison-Wesley),

Praktische Aufgaben

Erzeugen einer Folge von Pseudo-Zufallszahlen

Mit einem Tabellenkalkulations-Programm kann die Arbeitsweise der Methode der linearen Kongruenz gut untersucht werden. Sie finden hier einige Hinweise, wie Sie das Arbeitsblatt eines solchen Programms auslegen können, um die folgenden praktischen Fragen zu beantworten.

Legen Sie also ein neues, leeres Arbeitsblatt an und plazieren Sie darauf die folgenden Texte und Zahlen.

	A	B	C	D	E
1	**Methode der linearen Kongruenz zur**				
2	**Erzeugung von Pseudo-Zufallszahlen**				
3					
4	**Parameter**				
5	Divisor m:	64	Multiplikator a:	9	
6					
7	**Erzeugte Folge:**				
8	Startwert X_0:	21			
9	Erster Wert X_1				
10	…				

Setzen Sie in der Zelle B9 die Formel (1) ein. Sorgen Sie dafür, dass die Parameter m und a und der Startwert X_0 richtig übernommen werden. Stellen Sie sicher, dass die Bezüge auf die Parameter m und a vom Tabellenkalkulations-Programm absolut eingesetzt werden.

Dann kopieren Sie die Formel aus der Zelle B9 in der Spalte B nach unten. Füllen Sie einige Dutzend bis einige hundert der folgenden Zeilen mit der Formel. Wenn Sie die Bezüge richtig formuliert haben, sollte immer noch jede kopierte Formel die Parameterwerte aus den Zellen B5 und D5 beziehen, der Startwert X_i sollte aber immer genau aus der darüberliegenden Zeile kommen.

Betrachten Sie nun die Zahlenfolge, die Sie in den Zellen von B9 an abwärts erhalten haben. Beantworten Sie die folgenden Fragen:

Frage 1 Verteilung der erzeugten Zahlen:

Sie haben die Parameter wie folgt eingesetzt:

m = 64
a = 9

Prüfen Sie, nach wievielen Zeilen sich die Folge der erzeugten Zahlen wiederholt.

Entspricht dieses Resultat der maximalen erwarteten Länge der Folge?

Frage 2 Ungünstige Werte für die Parameter:

Setzen Sie nun anstelle der Original-Parameter die folgenden Werte ein:

m = 64
a = 1

Nach wievielen Zeilen wiederholt sich nun die Folge der erzeugten Zahlen?

Weshalb ist die mit diesem Parametern erzeugte Folge trotzdem keine Folge von Pseudo-Zufallszahlen?

Versuchen Sie selber noch weitere Parameter-Einstellungen. Wenn Sie über einen Rechner mit viel Speicherplatz verfügen, können Sie auch versuchen, wesentlich grössere Werte für m und a einzusetzen. Sie sollten dann anhand der in der Theorie erwähnten Regeln Folgen erzeugen können, die wesentlich länger sind.