

## Thema, Ziele: Zufallszahlengeneratoren

### Aufgabe 1: Verteilung der Zufallswerte durch `rand()`

`rand()` liefert gleichverteilte Werte zwischen 0 und inklusive `RAND_MAX`, d.h. im Intervall  $[0, \text{RAND\_MAX}]$ . Untersuchen Sie, wie gut diese Gleichverteilung eingehalten wird. Dazu soll das Testprogramm zufällige Zahlen im Bereich von  $[1, 20]$  liefern. Speichern Sie die Anzahl in einem Array und stellen Sie das Resultat in einem GUI oder mit Hilfe von Excel dar.

Hinweis: die man-Page für die Funktion `rand()` erhalten Sie mit Hilfe des Befehls `man 3 rand`

### Aufgabe 2: Güte von Zufallszahlengeneratoren

Oft werden für Tests Zahlen benötigt, welche zufällig aus einem gewissen Intervall sind. Ihre Aufgabe ist es, ein Programm zu entwickeln, welches Zufallszahlen zur Verfügung stellt. Selbstverständlich kann eine deterministische (eindeutig bestimmte) Maschine wie der Computer nicht echt zufällige Zahlen erzeugen sondern nur sogenannte Pseudozufallszahlen.

Die Zufallszahlen sollten möglichst gleichverteilt sein. Diese Anforderung könnte auch mit einem simplen Zähler erreicht werden, allerdings würde diese Zahlenreihe kaum als zufällig bezeichnet werden. Nebst der Gleichverteilung ist deshalb auch wichtig, dass die nächste Zahl nicht erraten werden kann.

Zufallszahlen werden oft als Zahlenreihen implementiert, die mit der folgenden Formel beschrieben werden können:

$$r_{i+1} = (a \cdot r_i + c) \bmod m,$$

Die Formel besteht aus einem ganzzahligen Startwert  $r_0$  und mit ganzzahligen Konstanten  $a$  (Multiplikator),  $c$  (Inkrement) und  $m$  (Modulus). Diese Konstanten sind sorgfältig zu wählen, damit die Zahlen zufällig werden. So erzeugte Zahlenfolgen haben immer eine Periode  $\leq m$ . Bei gegebenen Konstanten und Startwert  $r_0$  ist die Folge eindeutig festgelegt. Mit Hilfe der C-Funktion `srand()` wird der Startwert  $r_0$ , der random seed, festgelegt.

Zur Beurteilung der Güte von Zufallszahlen existieren in der Statistik verschiedene Verfahren. In dieser Aufgabe sollen Sie mit Hilfe des Computers einen graphischen Gütetest entwickeln und damit verschiedene Zufallszahlen-Generatoren testen. Verwenden Sie dazu nach freier Wahl eine der Plattformen, bzw. Bibliotheken:

- Visual Studio
- Eclipse mit Qt
- ARM 9 mit .NET  $\mu$ Framework

- a) Im graphischen Fenster soll ein Quadrat gezeichnet werden. Mittels zweier aufeinanderfolgender Zufallszahlen, die geeignet skaliert werden, wird ein Koordinatenpaar in diesem Quadrat festgelegt (1. Zufallszahl: x-Wert, 2. Zufallszahl: y-Wert) und an der entsprechenden Stelle ein Punkt gezeichnet. Dieser Vorgang wird beliebig wiederholt. Durch die Beobachtung der graphischen Anordnung der Punkte kann ein Eindruck über die Qualität des Generators gewonnen werden. Schlechte Generatoren zeigen im allgemeinen ein allzu ausgeprägtes Muster.

In Ihrem Programm sollen die Werte  $r_0$ ,  $a$ ,  $c$  und  $m$  eingegeben werden können.

Damit das fortlaufende Zeichnen der Punkte auch verfolgt werden kann, wird ein Timer eingesetzt. Als Timerintervall wird 10 ms gewählt, d.h. der Timer meldet sich dann alle 10 ms. Jedesmal wenn sich der Timer meldet, soll ein Punkt gezeichnet werden.

b) Testen Sie z.B. die folgenden Zufallszahlengeneratoren:

<b>a</b>	<b>r<sub>0</sub></b>	<b>c</b>	<b>m</b>
2	0	1	28
8	0	1	7
125	1	0	8192
21	1	3	64
25	1	1	64
3	1	1	32766
130	1	1	32766
1331	1	1	327661

c) Testen Sie den Zufallszahlengenerator des Systems: rand()