

---

MATHEMATIK I FÜR BIOLOGEN, GEOLOGEN UND GEOÖKOLOGEN  
Computer-Übungsblatt 5

**Aufgabe C13.** Zufallszahlen

Wir simulieren  $N = 500$  radioaktive Atome und nehmen an, dass jedes Atom mit einer Wahrscheinlichkeit von  $p = 3,7\% = 0,037$  innerhalb einer Sekunde zerfällt. Für diese Simulation erzeugen Sie eine Zufallszahl  $X$  (siehe Beispiel 7) mit  $X \in [0, 1]$ ; gilt  $X \leq 0,037$ , so soll das simulierte Atom zerfallen, sonst nicht. Das Verfahren wiederholen Sie in einem Programm `decay` so oft, bis das Atom zerfallen ist, und generieren so die zufällige Anzahl Sekunden  $t$  bis zum Zerfall. Lassen Sie dies für  $N = 500$  Atome wiederholen und plotten Sie in ein Histogramm, wie viele Atome wann zerfallen sind. Plotten Sie zum Vergleich ein Histogramm mit den zufalls-unabhängigen Werten (die um so genauer zutreffen, je größer  $N$  ist) für  $Z(t) =$  erwartete Anzahl der Zerfälle in  $[t, t + 1]$

$$Z(t) = Z(0) \exp(-\lambda t)$$

mit  $Z(0) = pN$ , wobei Sie zuerst  $\lambda$  bestimmen aus der Beziehung

$$\exp(-\lambda \cdot 1 \text{ sec}) = 1 - p.$$

(5 Punkte)

*Beispiel 7:* Erzeugung von Zufallszahlen, `while`-Schleife

Der Befehl `A = rand(N)` erzeugt eine  $N \times N$ -Matrix `A` mit Zufallswerten  $A_{ij} \in [0, 1]$ . Für unsere Zwecke reicht es aus, jeweils *eine* Zufallszahl  $X$  zu berechnen, also

```
X=rand(1);
```

In *octave* bezeichnet die Anweisung

```
while (x<10)
x=x+1;
end
```

eine sog. `while`-Schleife. Die Anweisung `x=x+1;` wird wiederholt (also jeweils 1 zu  $x$  addiert), solange die Bedingung `(x<10)` erfüllt ist. Nutzen Sie eine `while`-Schleife und die Zufallszahl-Generierung `rand(1)` für Aufgabe 12. Verwenden Sie hier für die Histogramme den Befehl `hist(s, 20)` (`s` sei hier ein Datenvektor). Dabei werden die Zahlenwerte in `s` in 20 Intervalle sortiert und die Anzahl der Elemente in jedem Intervall in Histogramm-Form ausgegeben.

**Aufgabe C14.**

Schreiben Sie in Programm `toss`, das zu gegebener natürlicher Zahl  $n$  eine Münzwurffolge der Länge  $n$  erzeugt, nämlich  $n$  Zufallszahlen  $X_1, \dots, X_n$ , die (unabhängig voneinander) die Werte 0 und 1 jeweils mit Wahrscheinlichkeit  $p = \frac{1}{2}$  annehmen. Lassen Sie die Anzahl der Einsen,  $N_n$ , bestimmen.<sup>1</sup> Dieser Wert sollte nahe bei  $\frac{n}{2}$  liegen, aber wie nahe eigentlich? Um sich zu

---

<sup>1</sup>Man sagt auch, Sie *simulieren die Zufallsvariable*  $N_n$ , wenn Sie Werte der Variablen  $N_n$  erzeugen lassen.

veranschaulichen, dass  $|N_n - \frac{n}{2}|$  in der Größenordnung von  $\sqrt{n}$  liegt, lassen Sie die simulierten Werte von  $N_n - \frac{n}{2}$  für  $n = 64, 256, 1024, 4096$  ausdrucken. (5 Punkte)

*Beispiel 8: if-Bedingung*

Zur Lösung von Aufgabe 13 benötigen Sie die schon bekannte Zufallszahl-Generierung `rand(1)` sowie ein Bedingungs-Anweisung. Folgende Schleife durchläuft ihre inneren Anweisungen zehnmal und überprüft, ob die Schleifenvariable `i=5`. Ist dies der Fall, dann wird der Text `Hello World` mit Hilfe von `disp` auf der Konsole ausgegeben.

```
for i=1:1:10
if(i==5)
disp('Hello World!');
end
end
```

Beachten Sie, dass nach Befehlen wie `for`-Schleifen, `function`-Deklarationen oder auch `if`-Anweisungen und deren Ende, KEIN Semikolon stehen darf.

**Abgabe:** Donnerstag, 24.11.2005, zu Beginn der Vorlesung.