Praktikum: "Mathematik am Computer" HS 2021
Prof. H. Harbrecht
P. Barendrecht, J. Bopp, M. Fallahpour, C. Santos Universität Basel

---

Serie 8

Matlab - Kontrollstrukturen, Repetition

zur 46. KW (15.11. – 19.11.2021)

Aufgabe 8.1 (2 Punkte): In dieser Aufgabe sehen wir, dass es für eine Problemstellung ganz verschiedene Lösungen geben kann. Dazu wollen wir uns anschauen, wie man alle geraden Zahlen von 1 bis 100 aufsummieren kann.

a) Verwende die Gaußsche Summenformel, d.h. es gilt:

$$\sum_{k=1}^{n} 2k = n(n+1).$$

Hinweis: Um alle geraden Zahlen von 1 bis 100 aufzusummieren, muss n in der Gaußschen Summenformel richtig gewählt werden und ist nicht 100.

b) Verwende eine `for`-Schleife.

c) Verwende eine `while`-Schleife.

d) Verwende den Matlab-Befehl `sum`.

e) Verwende eine geschickte Multiplikation mit einem `ones`-Vektor.

f) Du darfst dir gerne noch weitere Möglichkeiten überlegen.

Aufgabe 8.2 (2 Punkte): Schreibe ein Programm, das mit einem Intervall $[a, b]$ startet und dieses *solange* halbiert, bis der Durchmesser kleiner als eine angegebene Toleranz ist. Unterscheide hierbei zwei Fälle:

a) Ersetze jeweils den Endpunkt des Intervalls, d.h. wenn $[a, b_{n-1}]$ das alte Intervall ist, ist das neue von der Form $[a, b_n]$ mit

$$b_n = \frac{1}{2}(b_{n-1} + a).$$

Wiederhole dies *solange*, bis $b_n - a < 10^{-3}$ ist. Setze dabei als Startwerte $a = 0$ und $b = 1$. Zähle auch die Anzahl der Schritte, die dein Programm benötigt hat, und gebe diese aus.

b) Auf die gleiche Weise ersetze nun den Anfangspunkt des Intervalls, d.h. für jedes alte Intervall $[a_{n-1}, b]$ ist das neue von der Form $[a_n, b]$ mit

$$a_n = \frac{1}{2}(a_{n-1} + b).$$

Wiederhole dies *solange*, bis $b - a_n < 10^{-3}$ ist. Setze auch hier die Startwerte $a = 0$ und $b = 1$ und gib die Anzahl der benötigten Schritte aus.

Aufgabe 8.3 (3 Punkte): Betrachte folgendes Glücksspiel:

1. Zu Beginn zahlst du dem Veranstalter einmal einen Einsatz von 10 Franken.
2. Der Veranstalter wirft nun so lange mit einer fairen Münze, bis zum ersten Mal Kopf erscheint. Sei k die Anzahl der dazu notwendigen Versuche.
3. Der Veranstalter zahlt dir nun 2^{k-1} Franken aus.

a) Simuliere dieses Spiel in einem Matlab-Programm. Um den einzelnen Münzwurf zu simulieren, kannst du den Befehl `randi` benutzen. verwende eine `while`-Schleife, welche die Anzahl der Versuche k zählt und berechne schliesslich Auszahlung und Gewinn.

b) Nun wiederhole das Spiel 10^7 mal (`for`-Schleife). Zeichne sowohl die ausbezahlte Summe pro Spiel als auch den Gesamtgewinn in eine Abbildung. Dabei entspricht der Gesamtgewinn vom i-ten Spiel dem addierten Gewinn der Spiele 1 bis i. Was siehst du?

Aufgabe 8.4 (3 Punkte): Löse folgende Kurzaufgaben:

a) Schreibe eine Matlab-Funktion `c = Teilbar(a,b)`, die entscheidet, ob die Zahl a durch b teilbar ist. Falls dies zutrifft, soll die Funktion den Faktor a/b zurückgeben, ansonsten die Zahl a selbst. Teste die Funktion.

b) Schreibe eine Matlab-Funktion `C = Teiler(a)`, die den Vektor zurückgibt, der alle Teiler von a enthält. Wenn zum Beispiel $a = 17$ gewählt ist, soll die Funktion $C = (1, 17)$ zurückgeben. Wenn $a = 22$ gewählt ist, soll die Funktion $C = (1, 2, 11, 22)$ zurückgeben. Benutze die Funktion `Teilbar` in deiner Funktion `Teiler`. Teste die Funktion.

c) Schreibe eine Matlab-Funktion `Primzahl(a)`, die anzeigt ob die Zahl a eine Primzahl ist oder nicht. Ist a keine Primzahl, soll die Funktion zusätzlich auch alle Teiler von a anzeigen. Benutze zur Anzeige den Befehl `disp` oder `fprintf`. Benutze die Funktion `Teiler` in deiner Funktion `Primzahl`. Teste die Funktion.

Allgemeine Informationen zum Praktikum befinden sich auf der Webseite
`http://cm.dmi.unibas.ch/teaching/praktikumI/praktikumI.html`