Praktikum: "Mathematik am Computer" HS 2023
Prof. H. Harbrecht
M. Fallahpour, E. Gajendran, G. Ohlmann, M. Ramm Universität Basel

Serie 6

Matlab - Kontrollstrukturen II (if, for), Vektorisierung

zur 44. KW (30.10. – 2.11.2023)

Aufgabe 6.1 (1 Punkt): Schau dir die Folien genau an, überlege dir dann von Hand, was Matlab für folgende Code-Stücke ausgibt und beantworte die Fragen:

a)
```
x = 2; y = 3;
if x>y
     disp('x ist groesser als y')
else
     disp('y ist groesser oder gleich x')
end
```

b)
```
A = [0 1 2; -3 4 2; 3 2 1];
A(2:3, [1 3]) = [1 1; 2 2], A(A >= 2) = 8
```

c) Was bedeutet es eine Variable zu initialisieren? Wozu ist das nützlich?

Aufgabe 6.2 (1 Punkt): Benutze `if-else`-Verzweigungen, um ein Programm zu schreiben, das einer Variable s den Wert $s = -1$ zuordnet, falls $x < 0$, und $s = 1$, falls $x > 0$. Wenn $x = 0$, sei $s = 0$. In jedem anderen Fall, d.h. wenn s nicht eine reelle Zahl ist, dann soll "`Falsche Eingabe`" angezeigt werden.
Nützlich können dabei die Funktionen `isreal` und `isnumeric` und der Junktor & bzw. && sein. Teste dein Programm für verschiedene Werte von x.

Aufgabe 6.3 (2 Punkte): Zur Erinnerung: Mit der `if`-Verzweigung wird eine bedingte Ausführung von Befehlen erreicht. Mit `for`-Schleifen kann der Rechner angewiesen werden, bestimmte Dinge wiederholt auszuführen.

a) Schreibe ein Skript `PassFail.m`, welche die Note am Anfang als Argument erhält und als Ergebnis 1 für Noten grösser oder gleich 4 beziehungsweise 0 für Noten kleiner als 4 zurückgibt.

b) Schreibe ein Skript, das einen Vektor mit 100 zufälligen Noten zwischen 1 und 6 erstellt (Hinweis: `rand`). Zähle, wieviele Studierende bestanden haben.
Extra 1: Erstelle die zufälligen Noten wirklich mit `rand` statt mir `randi`.
Extra 2: Zähle die Anzahl der Studierenden die bestanden haben in einer Zeile.

Aufgabe 6.4 (4 Punkte): Erzeuge eine 100×100-Matrix $\mathbf{A}$, deren Einträge nur aus Nullen bestehen, und führe nacheinander die folgenden Operationen aus (du kannst im Workspace-Fenster auf die Matrix doppelklicken, um sie dir anzeigen zu lassen und um einfach zu überprüfen, ob du die Operation richtig ausgeführt hast). Löse die Aufgaben **ohne** `for`- oder `while`-Schleifen.

1. Setze die Einträge $a_{43,4}$ und $a_{91,100}$ auf 33.
2. Setze alle Einträge der 90. Zeile von $\mathbf{A}$ auf 42.
3. Setze ersten, vierten, siebten, etc. Eintrag der 35. Spalte auf 65.
4. Weise den Einträgen der 66. Zeile absteigend die Werte 199 bis 100 zu.
5. Setze jeden Eintrag $a_{i,j}$, wobei i und j beide durch 4 teilbar sind, auf 77.
6. Setze jeden Eintrag $a_{i,j}$, wobei i und j beides Primzahlen sind, auf 11.
7. Weise jedem Eintrag $a_{i,66}$, wobei i durch 3 und/oder durch 5 teilbar ist, aufsteigend die Werte 2 bis 48 zu.
8. Addiere zum ersten, dritten, etc. Eintrag auf der Hauptdiagonalen 22 hinzu.

Hinweis: Zur Lösung dieser Aufgabe sind der Junktor `|` und die Befehle `isprime`, `mod` und `eye` hilfreich. Hast du alles richtig gemacht, sollte `sum(A,'all')` die Zahl 78525 ergeben.

Aufgabe 6.5 (2 Punkte) Häufig führt die Benutzung von vielen `for`-Schleifen in Matlab zu einem Zeitverlust. Deshalb ist es vorteilhaft, dass man gewisse Schleifen durch Operationen mit Vektoren ersetzt. Siehe dazu die Anmerkungen zur vektoriellen Rechnung in Matlab in den Folien.

a) Schreibe ein Skript, welches das Skalarprodukt zweier Vektoren gleicher Länge berechnet. Im Gegensatz zu Aufgabe 5.3 a), soll hier keine `for`-Schleife oder die Befehle `dot` oder `sum` benutzt werden. Teste auch hier für die beiden Vektoren aus Aufgabe 5.3 a), ob dein Programm das Richtige tut.

b) Mit den Befehlen `tic` und `toc` kann in Matlab die Zeit für den Durchlauf eines Programms gemessen werden. Zum Beispiel wird für

```
tic
(1:1000).^2;
time=toc
```

die Zeit gemessen, wie lange es dauert, den Vektor $(1, 2, \ldots, 1000)$ eintragsweise zu quadrieren. Die verstrichene Zeit wird in der Variablen `time` gespeichert.
Vergleiche mit Hilfe dieser Befehle die Laufzeiten der beiden Programme zur Berechnung des Skalarprodukts (das eine Program mit `for`-Schleife aus Aufgabe 5.3 a) und das Andere ohne). Setze $n = 10^3, 10^4, 10^5, 10^6, 10^7, 10^8$ und betrachte die Vektoren $\mathbf{x} = (1, 2, \ldots, n)^\top$ und $\mathbf{y} = 3\mathbf{x}$. Speichere die Zeiten zur Berechnung des Skalarprodukts für die verschiedenen n in einer Variable ab und zeichne diese in einer Abbildung. Benutze die Plot-Optionen `'-o'` und `'-*'`. Füge einen Titel und eine Legende hinzu. Was stellst du fest?
Hinweis: Es werden 6 verschiedene Werte für n betrachtet. Initialisiere dementsprechend die Vektoren, in denen die Laufzeiten abgespeichert werden.

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