

Prof. Dr. Marcin Grzegorzek, Juniorprofessor – Fakultät IV, Elektrotechnik und Informatik
Hölderlinstr. 3, 57068 Siegen

Einführung in die Informatik II

SoSe 2012

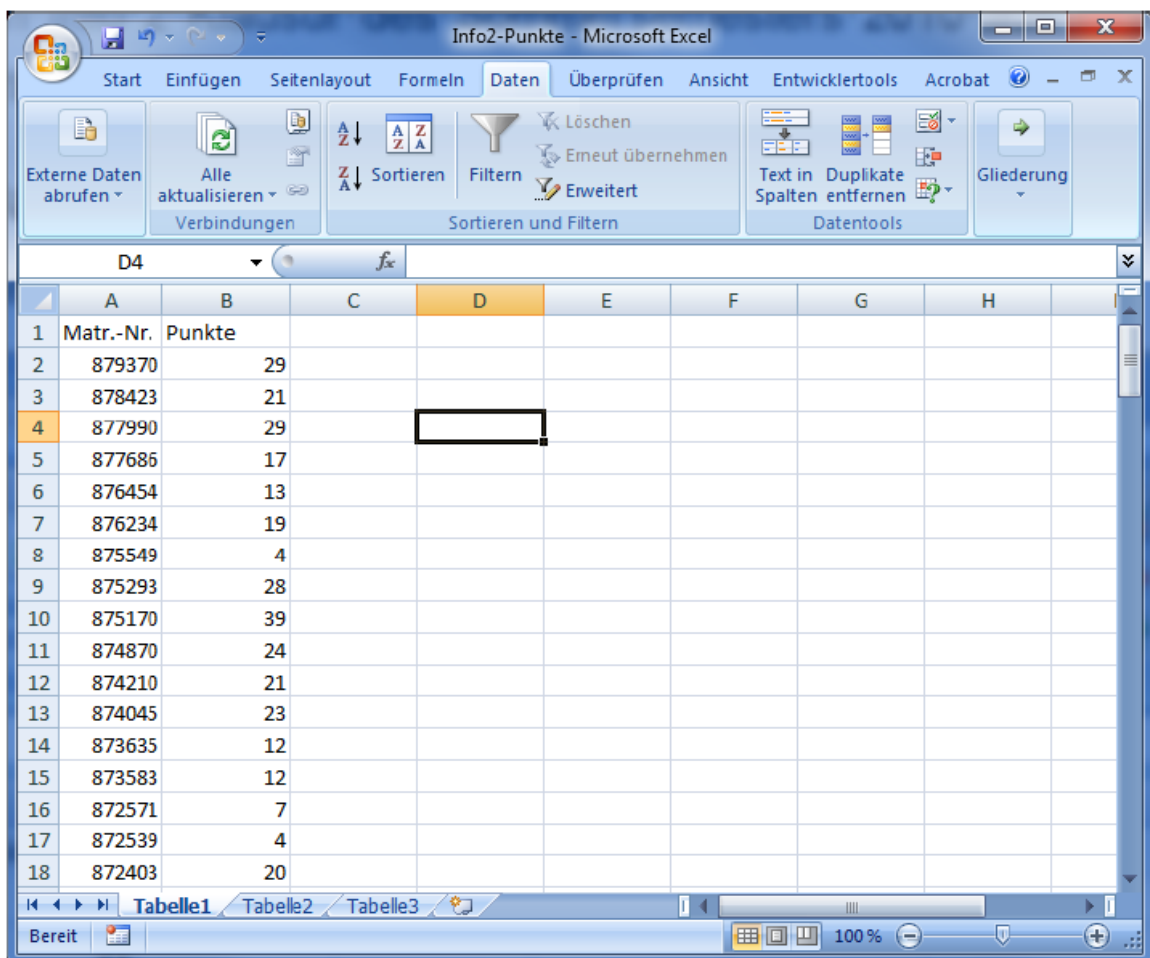
Übung 2

In dieser Übung sollen Matlab Funktionen geschrieben werden, welche die Ergebnisse einer Info 2 Klausur aus einer Excel-Tabelle einliest, auswertet und eine Textdatei für das Prüfungsamt erstellt.

Aufgabe 1

In der Datei 'Info2-Punkte.xlsx' stehen die Matrikelnummern in Spalte A und die erreichten Punkte in Spalte B.

Hinweis: Die Matrikelnummern sind frei erfunden und völlig willkürlich und haben nichts mit reellen Ergebnissen zu tun!



	A	B
1	Matr.-Nr.	Punkte
2	879370	29
3	878423	21
4	877990	29
5	877686	17
6	876454	13
7	876234	19
8	875549	4
9	875293	28
10	875170	39
11	874870	24
12	874210	21
13	874045	23
14	873635	12
15	873583	12
16	872571	7
17	872539	4
18	872403	20

Schreiben Sie eine Matlab-Funktion `Info2Ergebnisse`, welche die Daten aus der Excel-Datei einliest und die Matrikelnummern und Punkte als Vektoren wieder zurückgibt.

```
function [Punkte, MNummern] = Info2Ergebnisse()
```

Hinweis: Kopieren Sie die Excel-Datei in das gleiche Verzeichnis wie die Info2Ergebnisse. Damit vereinfachen Sie sich die Pfadangabe zu der Excel-Datei.

Aufgabe 2

Schreiben Sie eine Matlab-Funktion, welche die Punkte der einzelnen Teilnehmer mit Hilfe einer gegebenen Funktion (PunkteZuNoten.m) auswertet.

```
function Info2Auswertung()  
% Ergebnisse aus der Excel-Datei lesen  
[Punkte, MNummern] = Info2Ergebnisse();
```

Für diese Aufgabe wird folgende Funktion bereitgestellt:

```
function [Noten, moeglicheNoten]  
= PunkteZuNoten(Punkte, PMax, Note1, Note4)
```

Übergabeparameter der Funktion:

- Punkte: Vektor mit den Punkten
- PMax: Skalar mit der Maximalpunktzahl. In diesem Fall 40.
- Note1: Skalar mit dem Prozentsatz für die Note 1, bspw. 0.95
- Note4: Skalar mit dem Prozentsatz für die Note 4, bspw. 0.5

Rückgabeparameter der Funktion:

- Noten: Vektor mit den Noten zu den gegebenen Punkten
- moeglicheNoten: Vektor mit den zulässigen Noten, also 1.0, 1.3, 1.7 ... 5.0

Die Matlab-Funktion `hist()` berechnet die Werte für ein Histogramm. Beispiel:

```
histNoten = hist(Noten, moeglicheNoten);
```

Der Vektor `histNoten` hat so viele Elemente wie der Vektor `moeglicheNoten`. Das erste Element entspricht der Häufigkeit der Noten 1,0 und das letzte Element entspricht der Häufigkeit der Note 5,0 (durchgefallen).

Aufgaben:

- a) Berechnen Sie die Durchfallquote und den Anteil der Noten, welche besser oder gleich 2,0 sind.
- b)
 - a. Bestimmen Sie zuerst experimentell die Parameter `Note1` und `Note4` so, dass die Durchfallquote etwa 50 % beträgt und etwa 10% der Noten besser oder gleich 2,0 sind.
 - b. Setzen Sie die Werte für `Note1` und `Note4` auf ihre ursprünglichen Werte zurück. Programmieren Sie nun mit Hilfe von Schleifen und Bedingungen eine automatisierte Berechnung der in *b)a.* gestellten Anforderungen. Geben sie am Ende dann das Ergebnis wie folgt aus:

Wert von `Note4`: 0.46 - Durchfallquote liegt bei 50.38 Prozent

Wert von `Note1`: 0.91 - Anteil der Noten die 2,0 und besser sind liegt bei 9.92 Prozent

Aufgabe 3

Erweitern Sie die `Info2Auswertung()`.

Die Funktion soll die Matrikelnummern und den Status der Prüfung ('bestanden' bei besser oder gleich 4,0, 'nicht bestanden' bei 5,0) durch Semikolon getrennt in eine Textdatei namens "`MeldungPruefungsamt.txt`" schreiben.

Die Datei soll eine Kopfzeile haben und wie folgt aussehen:

```
Matrikelnummer; Status
875293; bestanden
875549; nicht bestanden
876234; bestanden
876454; nicht bestanden
877686; nicht bestanden
877990; bestanden
878423; bestanden
879370; bestanden
...
```