

7.4.A Projekte/Ausgleichsgerade – Arbeitsblatt

1 Problemstellung

Es ist ein C-Programm zu entwickeln, welches komfortabel die Ausgleichsgerade einer Menge von Punkten entwickelt und ausgibt.

Es sei eine gewisse Anzahl von Messwerten gemessen worden (z.B. der Strom I in Abhängigkeit von der Spannung U). Bedingt durch Ablesefehler wird sich das Ohm'sche Gesetz nie zu 100% bestätigen lassen. Gesucht ist also eine „möglichst gute“ Gerade durch die gemessenen Punkte, die so genannte Ausgleichsgerade.

Eine Ausgleichsgerade ist dadurch definiert, dass die Summe der Fehlerquadrate (wahrer Wert $U_W = 2\text{ V}$, abgelesener Wert $U_A = 1,7\text{ V}$, dann ist der Fehler $U_F = 0,3\text{ V}$ und das Fehlerquadrat $U_F^2 = 0,09\text{ V}^2$) zu einem Minimum wird.

2 Beispiel

In einer Mathematiksammlung findet sich folgendes Beispiel:

Aus einer Messung kennt man die drei Punkte P1(1,0;1,0), P2(2,0;2,0) und P3(3,0;4,0). Zur Berechnung der Ausgleichsgeraden wird die folgende Tabelle angelegt:

i	x_i	y_i	$x_i \cdot y_i$	x_i^2	y_i^2
1	1,0	1,0	1,0	1,0	1,0
2	2,0	2,0	4,0	4,0	4,0
3	3,0	4,0	12,0	9,0	16,0
Summe	$\sum x = 6,0$	$\sum y = 7,0$	$\sum x \cdot y = 17,0$	$\sum x^2 = 14,0$	$\sum y^2 = 21,0$

Die erste Spalte ist der Index; da die Punkte P1 bis P3 genannt wurden, läuft i (hier) von 1 bis 3. Die nächsten beiden Spalten beinhalten die x - und y -Koordinaten jedes Punktes. Die letzten drei Spalten enthalten nacheinander die Rechenwerte xy , x^2 und y^2 für jeden Punkt. In der untersten Zeile werden die Summen der Werte jeder Spalte angegeben.

Eine Gerade hat bekannterweise die Gleichung $y = m \cdot x + b$. Für den y -Achsenabschnitt b und die Steigung m werden nun folgende Formeln angegeben (die Nenner der beiden Brüche sind übrigens gleich):

$$b = \frac{\sum y \cdot \sum x^2 - \sum x \cdot \sum xy}{n \cdot \sum x^2 - (\sum x)^2}$$

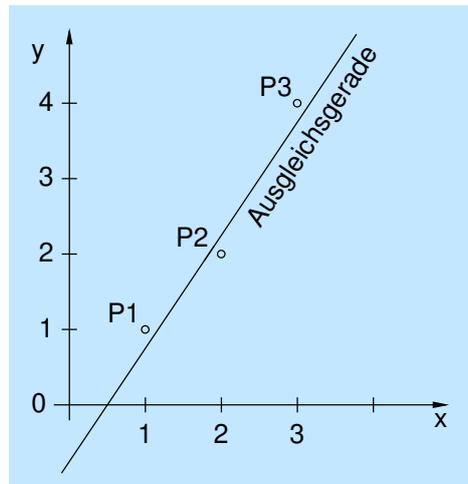
$$m = \frac{n \cdot \sum xy - \sum x \cdot \sum y}{n \cdot \sum x^2 - (\sum x)^2}$$

n ist in diesen Formeln die Anzahl der Punkte, bei unserem Beispiel $n = 3$. Für das Zahlenbeispiel ergibt sich dann:

$$b = \frac{7 \cdot 14 - 6 \cdot 17}{3 \cdot 14 - 6^2} = \frac{98 - 102}{42 - 36} = \frac{-4}{6} = -0,67$$

$$m = \frac{3 \cdot 17 - 6 \cdot 7}{6} = \frac{51 - 42}{6} = \frac{9}{6} = 1,5$$

Damit ist die Gleichung der Ausgleichsgeraden: $y = 1,5x - 0,67$.



3 Benutzeroberfläche

a) Das Programmmenü soll folgenden Bildschirmaufbau bewirken:

```

Terminal
schueler@debian964:~$ ausgleichsgerade
Berechnung einer Ausgleichsgeraden
=====
(1) Daten aufnehmen
(2) Daten ausgeben
(3) Daten korrigieren
(4) Daten speichern
(5) Daten laden
(6) Ausgleichsgerade ermitteln
(0) Ende
Treffen Sie eine Wahl: 1
...

```

b) Daten aufnehmen

```

Terminal
(1) Daten aufnehmen: (Ende: Ersten Punkt noch einmal eingeben)
-----
1. Punkt: x : 1,2
           y : 8,4
2. Punkt: x : 1,44
           y : 9,6
3. Punkt: x : 2,7
           y : 10,7
4. Punkt: x : 3,5
           y : 12,1
5. Punkt: x : 1,2
           y : 8,4
4 Punkte wurden aufgenommen.

```

c) Daten ausgeben

```

Terminal
(2) Daten ausgeben:
-----
1. Punkt: x: 1,2

```

```

                y: 8,4
2. Punkt: x: 1,44
                y: 9,6
3. Punkt: x: 2,7
                y: 10,7
4. Punkt: x: 3,5
                y: 12,1
Zur Fortsetzung <ENTER> druecken

```

d) Daten korrigieren

```

----- Terminal -----
(3) Daten korrigieren:
-----
Welchen Punkt wollen Sie korrigieren (0 fuer Ende): 1
Alter x-Wert = 1,42
Neuer x-Wert : 1,43
Alter y-Wert = 38,4
Neuer y-Wert : 83,4
Welchen Punkt wollen Sie korrigieren (0 fuer Ende): 0

```

e) Daten speichern

```

----- Terminal -----
(4) Daten speichern:
-----
Pfadname (Default: ag.txt): werte.xyz
Es wurden 4 Wertepaare gespeichert.

```

f) Daten laden

```

----- Terminal -----
(5) Daten laden:
-----
Pfadname (Default: ag.txt): werte.xyz
Es wurden 4 Wertepaare geladen.

```

g) Ausgleichsgerade ermitteln

```

----- Terminal -----
(6) Ausgleichsgerade ermitteln
-----
b = -0,67
m = 1,50
y = -0,67 + 1,5 x

```

Die Funktion `menue()` gibt den oben angegebenen Bildschirm aus und wartet auf eine (durch `Enter` abgeschlossene) Eingabe. Nach einer Eingabe wird die entsprechende Funktion ausgeführt. Danach wird wieder auf eine Tastatureingabe gewartet. Erst bei Eingabe einer 0 wird die Funktion und damit das Programm verlassen. Bei Fehleingabe wird eine geeignete Meldung ausgegeben.

Die Funktionen `speichern()` und `laden()` schreiben und lesen Textdateien, bei denen in jeder Zeile genau ein Wertepaar steht, zuerst der x-, dann der y-Wert, getrennt durch ein Leerzeichen.

Bei der Funktion `ausgabe()` soll nach jeweils Punkten und am Ende der Ausgabe auf das Betätigen der `Enter`-Taste gewartet werden. Die übrigen Funktionen `eingabe()`, `korrektur()` und `berechnung()` werden durch die Bildschirmausgaben erklärt.

4 Hinweise

- Wichtig: Die Aufgabe ist mit Hilfe von Funktionen und zweier in `main()` definierter Felder `fx` und `fy` zu lösen. Die Verwendung globaler Felder ist nicht erlaubt!
- Entwerfen Sie Top-Down. Sinnvoll ist also ein Anfang mit `main()` und `menue()`.
- Überlegen und dokumentieren Sie bei allen Funktionen (am einfachsten mit den Ihnen bekannten Templates), ob und welchen Rückgabewert (`return`) diese Funktionen haben sollten und welche Parameter sie benötigen.
- Um zu testen, ob `main()` und `menue()` richtig arbeiten, müssen Sie für die übrigen Funktionen erst einmal Dummies schreiben, die zwar schon die richtigen Funktionsköpfe haben, aber nur ihren Namen ausgeben, etwa so:

```
_____ Terminal _____  
Hier ist die Funktion speichern()  
Zur Fortsetzung <ENTER> druecken
```