

Einführung in die o++o-App

(Stand 12.05.2022)

Table of Contents

1. Die Tastatur.....	1
2. 12 weitere otto-Beispielprogramme.....	3
3. Wesentliche o++o Operationen.....	6
4. Handy-Screenshots dreier Beispiele.....	11

1. Die Tastatur



Oberhalb der Bildschirmeingabe befinden sich 4 Tasten (buttons)

+**op** um ein Operationssymbol auszuwählen.

tbt um den Namen einer auf dem Handy gespeicherte Datei in das Programmfeld zu laden.

prg erlaubt den gesamten Text eines otto-Programms zu laden, das sich auf dem Handy befinden muss, damit es anschließend ausgeführt (run) werden kann. Es kann zuvor auch noch modifiziert werden. Mit der Löschtaste

✗ kann der gesamte Programmtex mit einem Klick gelöscht werden.

Die erste Tastatur besitzt insgesamt 7 Zeilen mit insgesamt 36 Tasten (Buttons).

Dazu zählen

die Taste zur Festlegung des Ausgabeformats, die standardmäßig auf **web** eingestellt ist.

Es folgt der

run button. Es schließt sich das

keyboard Symbol zum Wechsel der Tastatur

Löschtaste für das zuletzt eingetippte Zeichen.

< > sind die Relationssymbole kleiner und größer.

Die nächsten drei Reihen beinhalten zunächst die zehn Ziffern.

(und) sind die gewöhnlichen Klammern. Damit kann die einfache Abarbeitung eines Ausdrucks von links nach rechts modifiziert werden (Klammern zuerst).

% ist das Prozentzeichen.

/ dient als Trenner zwischen rationalen Zahlen (zwischen Zähler und Nenner)

Die nächste Zeile beginnt mit

' dem Hochkomma: Es dient der Gruppierung großer Zahlen.

Mit

hoch (to the power of) ist die Potenzfunktion, mit ihr können Zahlen auch übersichtlich eingegeben werden.

= ist das Gleichheitszeichen.

Die 10 Ziffern müssen nicht erklärt werden.

Es folgen 4 Tasten für die grundlegenden Tabmentoperationen:

+ (Addition)

- (Subtraktion)

* (Multiplikation)

: (Division)

Diese Operationen sind bereits wesentlich allgemeiner als man erwarten würde, da sie auf Listen und ganze Tabmente angewandt werden können.

! dient als Trennzeichen, da das Komma bereits eine (Paar) Operation ist.

l (Liste) und

m (Menge) sind die wichtigsten Kollektionssymbole. Mit ihnen und mit den Spaltennamen können Schemen für strukturierte Tabellen gebildet werden; die einer gib-Anweisung folgen müssen.

gib erlaubt ein Schema für eine strukturierte Tabelle anzugeben. Dadurch kann man Sortier- und Umstrukturierungswünsche ausdrücken. Ferner können Aggregationen wie Summen und Durchschnitte spezifiziert werden.

avec (mit) (französisch) erlaubt (strukturierte) Zeilen in Tabellen zu selektieren bzw. wegzulassen.
Der Punkt

- ist das Trennzeichen für Punktzahlen (Kommazahlen gibt es in keiner Programmiersprache.)
, Das Komma ist ebenfalls eine Operation. Dadurch werden lediglich zwei Tabmente nebeneinander gestellt. Mathe, 1 ist beispielsweise ein Paar, bestehend aus einem Wort und einer Zahl.

= stellt die Gleichheit zweier Werte fest.

Durch ihre Anwendung werden Wahrheitswerte durch die kurzen spanische Worte **si** und **no** ausgegeben. Es folgt die etwas breitere Taste für das

Leerzeichen.

Mit dem gebogenen Pfeil (RETURN) als letzte Taste (rechts unten) bewegt man den Cursor auf einen neuen Zeilenbeginn.

Mit den 8 Operationstasten im engeren Sinn, können weiterhin folgende Operationen durch mehrere Tastendrücke gebildet werden:

(zwei Tasten)

++ **--** ****** **::** **+%** **-%** **..** **'3** **'4**
+m **+l** **-m** **-l** **:m** **:l** ***m** ***l**

,

(drei Tasten)

++1 **++:** **...**

Relationssymbole mit 2 Tasten:

<= **>=** **!=**

Schemaoperationen:

,

(zwei Tasten)

1- **m-**, um abwärts zu sortieren.

D.h., es stehen mit der ersten Tastatur bereits 30 Tabmentoperationen und 6 Relationssymbole zur Verfügung.

Ferner kann man die Zuweisungsoperation (Erweiterung um eine neue Spalte) **:=** und **::=** (Überschreiben einer Spalte) tippen.

Für die restlichen Operationen muss auch keine andere Tastatur benutzt werden. Man kann nach dem Drücken von **+op** das entsprechende Operationssymbol auswählen.

hoch wird durch Klicken ein Leerzeichen voran und nachgestellt. Damit soll bei der ersten Tastatur ein sehr schnelles Tippen einfacher Zahlen ermöglicht werden:

10 hoch 7*1.23

kann daher auch mit 9 Tastenklicken realisiert werden.

2. Typische otto-Beispielprogramme

Die folgenden einfachen Programme sollen die Nutzung demonstrieren.

1. term1.otto: 1+2*3+4

Hier wie generell wird einfach von links nach rechts gerechnet und nicht in der Mitte begonnen.
Ergebnis: 13

2. grosseZahl.otto: 10 hoch 9*1.887

Es muss das Wesentliche zuerst getippt werden:

Basis: 10

Funktion: hoch

Exponent: 9

Damit sind Milliarden gemeint. Es folgen die Details:

1.887

Ergebnis: 1887000000.

Schreibt man die Mantisse (1.887) zuerst

1.887*10 hoch 9, so wird 18.87 hoch 9 berechnet, was in seltneren Fällen gewünscht ist.

3. großeZahlformatiert.otto: 10 hoch 9*1.887 '3

Ergebnis: 1'887'000'000.

4. rationaleAddition.otto: 1/3 + 2/7

Ergebnis: 13/21

Jedes der folgenden Programme sollte wenigstens mit tab und säulen betrachtet werden.

5. liste.otto: Darstellung von 5 Zahlen (hier Noten) (einspaltige tab-Tabelle)

2 4 1 4 3

6. tabelle.otto: einfache zweispaltige Tabelle in einer Zeile eintippen (eine Liste von 5 Paaren)

Mathe, 2 Phy, 4 Deu, 1 Eng, 4 Werken, 3

komplexere (normale) Tabellen würde man allerdings über "Dateien" eingeben. Die einfache Tastatur erlaubt aber auch schon mehrere zusätzliche Möglichkeiten.

7. strukturiertetabelle.otto: strukturierte Tabelle in einer Zeile eintippen

Mathe, [2 3 4 2] Phy, [4 5 1] Deu, [1 3] Eng, [4] Werken, [3]

Hierbei handelt es sich um 5 Paare, wobei die zweite Komponente jeweils eine Liste ist.

8. einfachesdiagramm.otto: Ermittle bzw. stelle die Durchschnitte der Fächer dar.

FACH, NOTE1 1 := Mathe, [2 3 4 2] Phy, [4 5 1] Deu, [1 3] Eng, [4]

DUR:= NOTE1 ++:

gib FACH, DUR m

9. punkte.otto: 5 Punkte durch Koordinaten vorgeben

1,2 1,3 1,5 2,2 2,3

Da die Punkte bei der Repräsentation mittels **bild** sehr klein sind, sind sie schwer zu erkennen.

9. wertetabelle.otto: Berechne 10 Funktionswerte

X1 := 1 .. 10

Y := X sin

10. funktionsbild.otto: Berechne Funktionswerte der Funktion (Geraden) X*0.5 + 1

X1 := 1 ... 10!0.1

X2 := X poly 0.5 1

Verringert man die Schrittweite von 0.1 auf 0.01, so werden nicht 91 sondern 901 Funktionswerte berechnet. Damit bildet sich bereits ein Funktionsverlauf aus. Das Auge kann die einzelnen Punkte nicht mehr klar unterscheiden.

Mit **säulen** kann man Programm 10 gut darstellen. Das modifizierte Programm mit der Schrittweite 0.01 besitzt dagegen zu viele "Säulen".

Es werden dabei sowohl die X-Werte als auch die Y-Werte als Säulen dargestellt. Häufig will man die X-Werte als Unterschrift und den Y-Wert für die Säulenhöhe. Die folgende zusätzliche Programmzeile gewährleistet das:

```
X ::= X wort
```

11. extremwert.otto: Berechne das erste (lokale) Maximum der Sinusfunktion.

```
1 ... 2!0.0001 sin max
```

Das Ergebnis 0.99999999993 ist bereits eine gute Näherung für die 1 . Bei der Schrittweite 0.000001 ergibt sich sogar der exakte Wert 1 .

12. minimumpolynom.otto: Berechne das lokale Minimum einer Funktion dritten Grades.

```
-1 ... 3!0.0001 poly [1 2 -3 5] min
```

Ergebnis: 4.12058025421

Man kann sogar die eckigen Klammern weglassen, um schneller tippen zu können. Nur wenn das Polynom zu einer Konstante mutiert, muss man diese setzen, z.B.:

```
9 poly [2]
```

Dass in dem betrachteten Intervall tatsächlich ein lokales Minimum vorliegt, kann man durch das **bild**-output mit dem folgenden Programm klären.

```
X1 := -1 ... 3!0.01
```

```
Y := X poly [1 2 -3 5]
```

Die betrachtete Funktion hat in alter Schreibweise die Gestalt:

$$x^3 + 2*x^2 - 3*x + 5$$

13. flaeche.otto: Berechne die Fläche unter den beiden ersten "positiven" Sinusbögen.

```
0 ... 2*pi! 0.00001 sin abs *0.00001 ++
```

Dank der Benutzung von "double" in OCaml ist 4.00000000005 offensichtlich eine sehr gute Näherung für 4 .

14. funktionsbilder.otto: Gib 3 Funktionen mit 3 verschiedenen Farben aus.

```
X1 := -5 ... 5!0.01
```

```
RGB1 := 0*X, 0, 1 # blau
```

```
Y1 := X sin
```

```
RGB2 := 0*X, 1, 0 # grün
```

```
Y2 := X abs sqrt
```

```
RGB3 := 1, 0.2, 1 *X sin abs # Farbverlauf
```

```
Y3 := X abs hoch 1.3
```

Eine Farbe (RGB-Wert) wird in o++o durch ein Tripel von drei Zahlen zwischen 0 und 1 beschrieben. Es gibt mehr als 100 Farbnamen. Diese müssen aber mit einer anderen Tastatur eingegeben werden und für diese muss man die englischen Bezeichnungen kennen. Der Code von **blue** ist demnach $0, 0, 1$ und von **darkorange** $1, 0.54901, 0$. Normalerweise schreibt man für die RGB1-Zuweisung **RGB1 := blue at X** oder **RGB1 := 0, 0, 1 at X**. Da für beide Zuweisungen eine weitere Tastatur benötigt wird, haben wir obige gewählt.

Die Zuweisung für **RGB3** beschreibt sogar einen Farbverlauf. Betrachtet man das Ergebnis von Programm 13 als Tabelle, so erkennt man verschiedene Farbwerte in den 3 zugehörigen Spalten. Das liegt daran, dass sich das **X** in $1, 0.2, 1 *X$ auf alle drei vorangehenden Werte bezieht. Für **X=0** ergibt sich also der Farbwert $0, 0, 0$ und für **X=1** der Wert $1, 0.2, 1$. Damit

stets positive Farbwerte Berücksichtigung finden, wurde abschließend noch die `abs`-Operation angefügt.

15. gib.otto: Umstrukturierung, Sortierung einer gegebenen Tabelle mit vorherigen Selektionen.
`fluesse.tabh`

```
avec BUNDESLAND in [Sachsen Thüringen Hessen]
gib BUNDESLAND, DUR, (LAENGE, FLUSS m) m-
    DUR:=LAENGE! ++
avec LAENGE > 280
```

Wenn man diese Datei nicht selbst eintippen will, kann man sie über "Dateien", "+" und "Datei importieren" auf dem Handy speichern.

In Anfrage 12 werden zunächst Flüsse selektiert, die durch obige 3 Bundesländer fließen. Die eckigen Klammern könnte man auch weglassen. Diese Flüsse werden bzgl. BUNDESLAND strukturiert (gruppiert) und die Flüsse, die durch jedes Bundesland fließen werden nach der Länge sortiert. Das `m`- (Menge) garantiert, dass jedes Bundesland nur einmal im Ergebnis auftaucht und nach Bundesländern abwärts sortiert wird. Das `m` davor bezieht sich auf die Spalten `LAENGE` und `FLUSS`. Nach der ersten Spalte (hier `LAENGE`) wird stets zuerst sortiert. `DUR` enthält die Durchschnittslänge aller Flüsse der gegebenen Datei, die durch das entsprechende Bundesland fließen. Von der Ergebnismenge verbleiben dann lediglich die, die länger als 280 km sind. Man erkennt hieran bereits, dass Selektionen (Filter-Operationen) und andere Operationen in beliebiger abwechselnder Reihenfolge angewandt werden können.

3. Wesentliche o++o Operationen

In untenstehender Tabelle benutzen wir folgende Abkürzungen.

TT = Tabment Typ

TT1 = Tabment Typ des ersten Inputtabments

Zahl = ZAHL oder PZAHL oder RATIO

Text = TEXT oder WORT

Operation	Stelligkeit	Output-TT	Bedeutung	Beispiele
+	2	TT1	Addition von Zahlen und Verbindung von Texten	1 3 + 2.1 ergibt 3.1 5.1 xy ab + de ergibt xyde abde
*	2	TT1	Multiplikation	2 3 5 * 2 ergibt 4 6 10
-	2	TT1	Subtraktion	3 - 2 ergibt 1 1'234 - 345 ergibt 889 3 - 2

Operation	Stelligkeit	Output-TT	Bedeutung	Beispiele
				ist dagegen eine Liste und ergibt 3 - 2
:	2	TT1	Division	3 : 4 ergibt 0.75
++	1	Zahl	Summe	2 3 6 ++ ergibt 11
**	1	Zahl	Produkt	1 3 5 ** ergibt 15
--	1	Zahl	Mehrfachsubtraktion	20 5 4 -- ergibt 11
::	1	Zahl	Mehrfachdivisionen	64 2 2 :: ergibt 16
%	2	Zahl	Prozentwert	200 % 5 ergibt 10
+%	2	Zahl	Prozentwert hinzufügen	200 +% 5 ergibt 210
-%	2	Zahl	Prozenwert abziehen	200 -% 5 ergibt 190
++:	1	PZAHL	arithmetischer Durchschnitt	1 2 3 2 ++: ergibt 2.0
++1	1	ZAHLEN	Anzahl	3 4 7 9 ++1 ergibt 4
+m	2	X1,..Xn m	Vereinigung zu Menge	{1 2 3} +m {6 3} ergibt {1 2 3 6}
+l	2	X1,..Xn l	Vereinigung zu Liste	3 5 5 +l 6 3 ergibt 3 5 5 6 3
-m	2	X1,..Xn m	Differenz zu flachen Mengen	{2 4 5} -m {6 2} ergibt {4 5}
-l	2	X1,..Xn l	Differenz zu flachen Listen	1 3 4 -l 5 3 ergibt 1 4
:m	2	X1,..Xn m	mengentheoretischer	{1 3 7} :m {3 9}

Operation	Stelligkeit	Output-TT	Bedeutung	Beispiele
			Durchschnitt	ergibt $\{3\}$
:1	2	X1,..Xn l	Listen-Durchschnitt	2 2 4 :1 3 2 4 ergibt 2 4
*m	2	X1,..Xn m	kartesisches Produkt	{1 2} *m {3 4 6} ergibt {1, 3 1, 4 1, 6 2, 3 2, 4 2, 6}
*1	2	X1,..Xn l	kartesisches Listenprodukt	1 2 *1 2 3 ergibt 1, 2 1, 2 2, 2 2, 2
,	2	TT1,TT2	Komma; Paarbildung	1 2, 3 ergibt ZAHL1, ZAHL 1 2 3
=	2	BOOL	Gleichheit	1 = 2 ergibt si
!=	2	BOOL	ungleich	2 != 1 ergibt no
:=	1	TT1+ neue Spalte	Zuweisung	X := 1 Y := X + 2 ergibt X, Y 1 3
::=	1	TT1	Überschreiben	X ::= X + 3 * Y
<	2	BOOL	kleiner	3 < 4 ergibt si
>	2	BOOL	größer	3 > 4 ergibt no
<=	2	BOOL	kleiner gleich	2 <= 2 ergibt si
>=	2	BOOL	größer gleich	2 >= 4 ergibt no
..	2	Zahl l	von bis	1 .. 4 ergibt 1 2 3 4
...	3	Zahl l	von bis ! schrittweite	0 ... 6 ! 2 ergibt

Operation	Stelligkeit	Output-TT	Bedeutung	Beispiele
				0 2 4 6
'3	1	TT1	große Zahlen in 3er Blöcke formatieren	1234567890 '3 ergibt 1'234'567'890
'4	1	TT1	große Zahlen in 4er Blöcke formatieren	12345.67898 '4 ergibt 1'2345.6789'8
abs	1	TT1	absoluter Betrag	-3 abs ergibt 3 7 abs ergibt 7
avec	1	TT1	Selektion (mit)	fluesse.tabh avec LAENGE >800
gib	1+sche me+..	S2	restrukturiere, transformiere ein Tabment in ein Tabment mit gegebenen Schema bzw. gegebener TTD	aus studenten.tab gib FAK,(ORT,NAMEm m)m
hoch	2	TT1	Potenz	4 hoch 1/2 ergibt 2.
log	2	TT1	allgemeiner Logarithmus	100 log 10 ergibt 2.
max	1	Zahl	Maximalzahl	12.21,2,Hallo max ergibt 12.21
median	1	Zahl	mittlere Zahl	1 2 4,9.9 median ergibt 3.0
min	1	Zahl	Minimalzahl	12.21,2,Hallo min ergibt 2
pi	0	PZAHL	Kreiskonstante	KREISFLAECHE:=R*R*pi
poly	2	TT1	Polynom	3 poly [1 2 3] ergibt 18
rnd	2	TT1	runden	17.678 3.45 zz 8 rnd 1 ergibt 17.7 3.5 zz 8
sans	1+Bed	TT1	Selektion (ohne)	sans ORT=Magdeburg

Operation	Stelligkeit	Output-TT	Bedeutung	Beispiele
				sans Magdeburg sans: ohne die spezifizierten (komplexen) Tupel
sin	1	TT1	Sinusfunktion	3.14159 sin ergibt 2.65358979335e-06
sqrt	1	TT1	Quadratwurzel	4 sqrt ergibt 2.
streu	1	PZAHL	Streuung	[1 2 5 3 5 1] streu ergibt 1.5

Abschließend sollen noch die Operationen für Schemen angegeben werden. Sie werden beispielsweise in gib und igib-Anweisungen benutzt.

Operation	Stelligkeit	Bedeutung	Beispiele
,	2	Komma; Paarbildung auf Schemaebene	NAME , ORT NAME , HOBBY l
b	1	Symbol für Multimengen (bag)	NAME b
m	1	Symbol für Mengen	ORT m NAME , ORT m
l	1	Symbol für Listen	NOTE l
b -	1	Symbol für Multimengen	
m -	1	Symbol um Mengen abwärts zu sortieren	GEHALT , NAME , ORT m -
l -	1	Symbol um die Reihenfolge einer Liste umzukehren	NOTE l -

4. Handy-Screenshots dreier Beispiele

Im ersten Beispiel werden Durchschnitte von Noten für sehr wenige Fächer berechnet und als web-Tabelle und dann als säulen-Diagramm ausgegeben. Die Farbzuweisung der Säulen erfolgt hier durch einen Standard vom System:

Das Programm `durchschnitteeinfach.otto` findet man auf dem deutschen Server.

```
FACH,NOTE1 m := Mathe, [1 2 1] Deu, [1 2]
```

```
DUR:=NOTE1 ++:
```

```
rnd 2
```

liefert als web-Ausgabe:

The screenshot shows a mobile application interface. At the top, there is a black header bar with white icons for time (19:50), signal strength, battery level (95%), and a back arrow. Below the header is a title "Ausgabe" with a left arrow icon. The main content area contains a table and a bar chart.

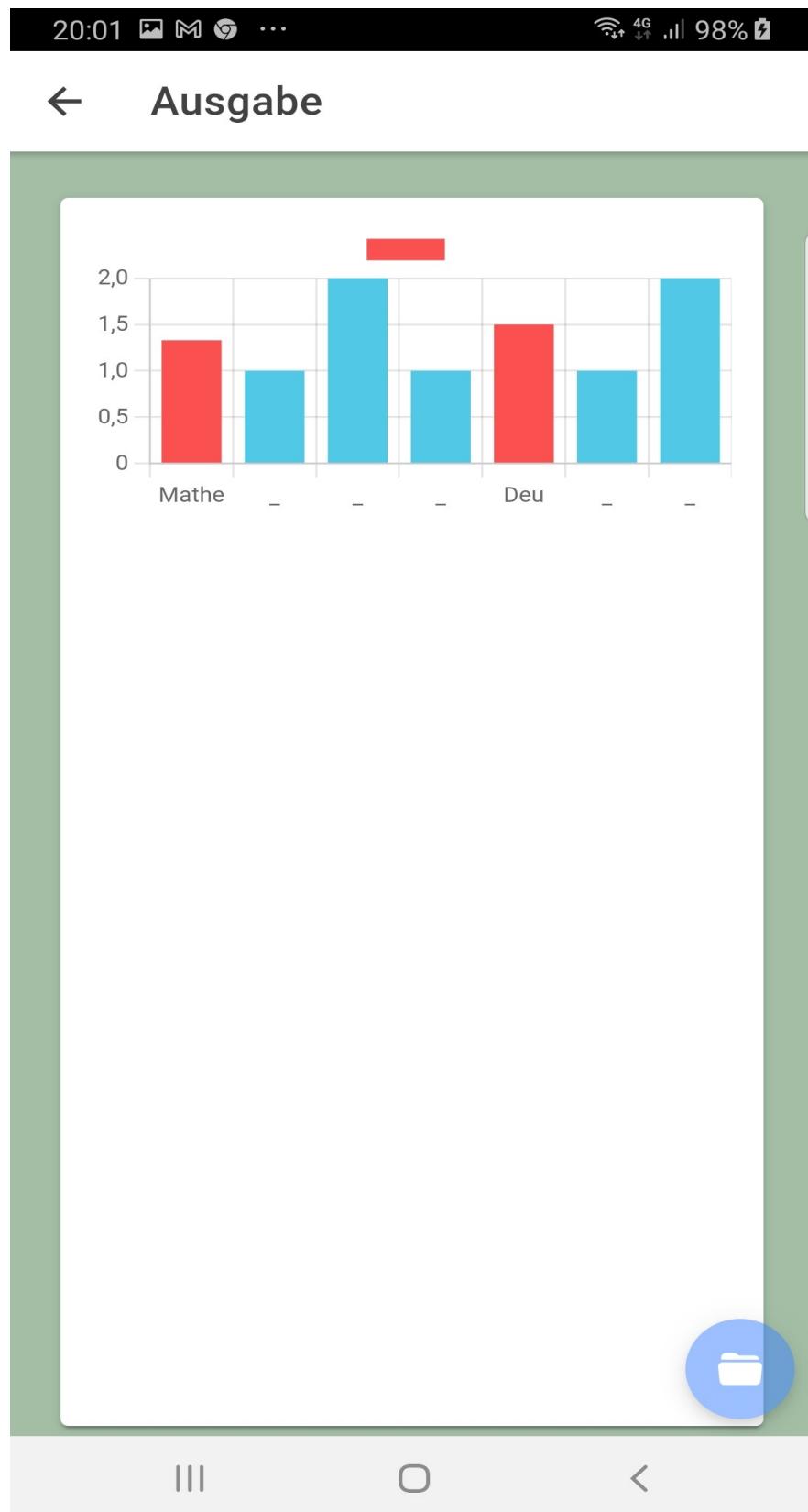
Table Data:

FACH	DUR	NOTE1
Mathe	1.33	NOTE 1 2 1
Deu	1.5	NOTE 1 2

Bar Chart:

A bar chart is displayed below the table, showing the distribution of notes for each subject. The bars are colored according to the standard color scheme: light green for the first bar (Mathe) and light blue for the second bar (Deu). The heights of the bars correspond to the values in the NOTE1 column of the table.

und als Säulendiagramm:



Mit dem zweiten Beispiel `funktionsbilder.otto` werden 3 Funktionen lediglich durch die `bild`-Repräsentation ausgegeben. Der Nutzer hat die Farben selbst ausgewählt, wobei an der Funktion Y3 ein Farbverlauf illustriert werden soll. Wenn "vollständige" Bilder generiert werden sollen, ist diese Möglichkeit erforderlich:

```
X1:= -5 . . . 5!0.002  
RGB1:= 0*X, 0, 1 # blau  
Y1:= X sin  
RGB2:=0*X, 1, 0      # grün  
Y2:=X abs sqrt  
RGB3:=1, 0.549, 0*X abs # darkorange  
Y3:=X abs hoch 1.3
```

In der Zeile von RGB1 ist die "formale" Multiplikation mit 0 erforderlich, damit für jeden X-Wert ein RGB1- Wert generiert wird. Man könnte auch

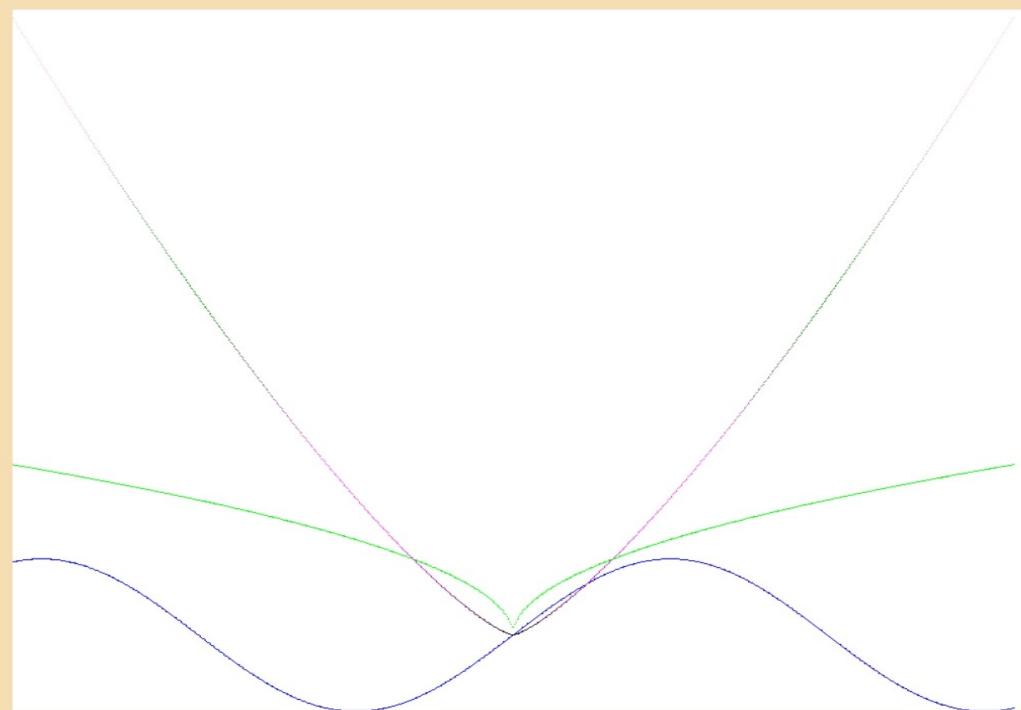
`RGB1:= blue at X`

schreiben. Durch

`Y0 := X*0`

hätte man auch noch eine X-Achse hinzufügen können.

← Ausgabe



|||

O

<

Das Programm fluessesaeulen.otto

fluesse.tabh

avec BUNDESLAND in Sachsen Hessen Thüringen

gib BUNDESLAND, (LAENGE, FLUSS m) m-

DUR:=LAENGE1 ++:

avec LAENGE>280

rnd 1

liefert die unten stehende tab bzw. säulen-Ausgaben. Durch m- werden die Bundesländer abwärts sortiert. Da mittels rnd 1 auf eine Stelle nach dem Komma gerundet wird, ist die Ergebnistabelle auch auf dem Handy noch relativ gut sichtbar:

The screenshot shows a mobile phone interface with a black header bar at the top containing the time (09:23), signal strength, battery level (19%), and other icons. Below the header is a navigation bar with a back arrow and the word "Ausgabe". The main content area displays a table with a light green header row and a light orange body row. The table lists river names and their lengths for three German states: Thüringen, Sachsen, and Hessen. The data is sorted by length in descending order.

BUNDESLAND, DUR, (LAENGE, FLUSS m) m-			
Thüringen	256.5	292	Werra
		413	Saale
Sachsen	606.3	400	Spree
		1165	Elbe
Hessen	374.6	292	Werra
		367	Neckar
		452	Weser
		544	Main



Wenn im Ergebnis mehr Flüsse ausgegeben werden, verschwinden in der Säulen-Repräsentation einige Unterschriften aus Platzgründen.