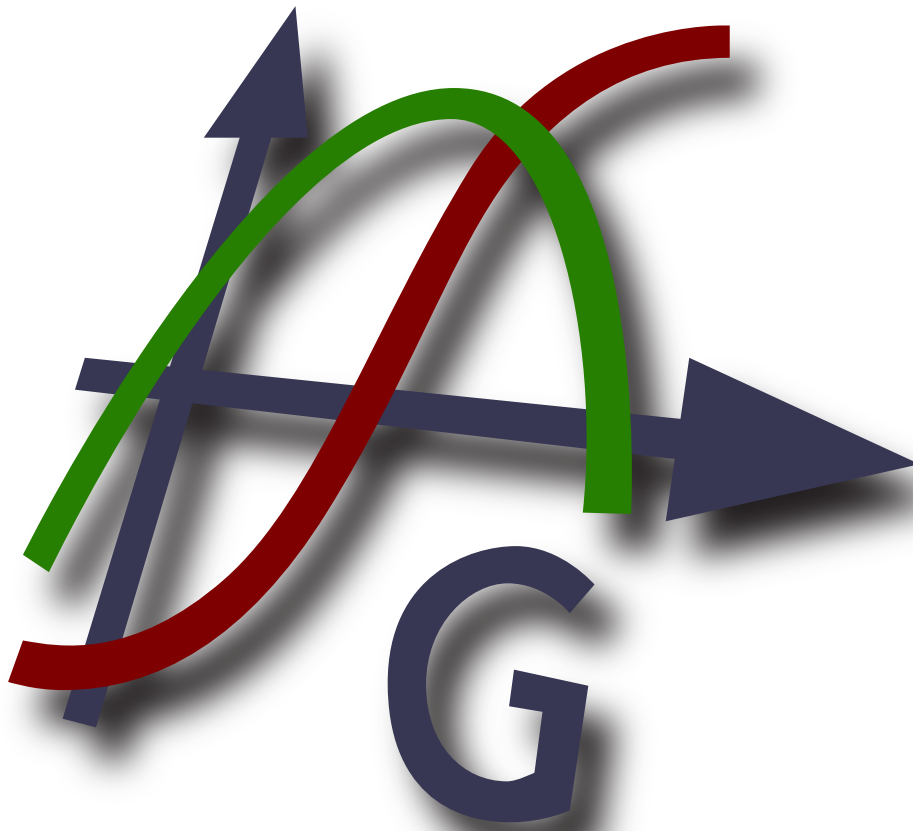


Graph



Version 4.5

Übersetzer:

Michael Bach Ipsen (translations@padowan.dk)
Sebastian Stütz (hauptsache@sonnenkinder.org)
Frank Hüttemeister (frank.huettemeister@padowan.dk)

Copyright © 2019 Ivan Johansen

Inhaltsverzeichnis

Was ist Graph?	1
Zur Benutzung von Graph	2
Installation und erstes Starten	3
Häufig gestellte Fragen	5
OLE-Server/Client	8
Liste der Menüpunkte	9
Fehlermeldungen	13
Funktionen	17
Funktionsliste	17
Konstanten	20
Die Konstante rand	20
Trigonometrisch	20
sin-Funktion	20
cos-Funktion	20
tan-Funktion	21
asin-Funktion	21
acos-Funktion	21
atan-Funktion	21
sec-Funktion	22
csc-Funktion	22
cot-Funktion	22
asec-Funktion	23
acsc-Funktion	23
acot-Funktion	23
Hyperbolicus	23
sinh-Funktion	23
cosh-Funktion	24
tanh-Funktion	24
asinh-Funktion	24
acosh-Funktion	25
atanh-Funktion	25
csch-Funktion	25
sech-Funktion	25
coth-Funktion	26
acsch-Funktion	26
asech-Funktion	26
acoth-Funktion	26
Potenz und Logarithmus	27
sqr-Funktion	27
exp-Funktion	27
sqrt-Funktion	27
root-Funktion	27
ln-Funktion	28
log-Funktion	28
logb-Funktion	28
Komplex	29
abs-Funktion	29
arg-Funktion	29
conj-Funktion	29
re-Funktion	29
im-Funktion	30
Runden	30
trunc-Funktion	30
fract-Funktion	30
ceil-Funktion	30

floor-Funktion	31
round-Funktion	31
Stückweise	31
sign-Funktion	31
u-Funktion	32
min-Funktion	32
max-Funktion	32
range-Funktion	32
if-Funktion	32
Speziell	33
integrate-Funktion	33
sum-Funktion	33
product-Funktion	34
fact-Funktion	34
gamma-Funktion	34
beta-Funktion	35
W-Funktion	35
zeta-Funktion	35
mod-Funktion	35
dnorm-Funktion	36
Dialoge	37
Achsen bearbeiten	37
Einstellungen	39
Funktion einfügen	41
Tangente/Normale einfügen	42
Schraffur einfügen...	43
Punktserie einfügen	46
Trendlinie einfügen	48
Textfeld einfügen	50
Relation einfügen	51
Ableitungsfunktion $f'(x)$ einfügen	52
Benutzerdefinierte Funktionen/Konstanten	53
Evaluiieren	54
Tabelle	55
Animieren	56
Als Bilddatei abspeichern	58
Punktserie importieren	58
Plugins	60
Danksagungen	61
Glossar	64

Was ist Graph?

Das Programm Graph wurde ursprünglich entworfen, um Graphen von mathematischen Funktionen in ein Koordinatensystem zu zeichnen. Graph ist ein normales Windows-Programm mit Menüs und Dialogen und unterstützt das Zeichnen von Standardfunktionen, Parameterfunktionen, Polarfunktionen, Tangenten, Punktserien, Schraffuren und Relationen. Es ist auch möglich, eine Funktion für einen vorgegebenen Punkt zu berechnen, einen Graphen mit der Maus zu verfolgen und noch viel mehr. Weitere Informationen unter [Zur Benutzung von Graph](#).

Graph ist freie Software; Sie können es gemäß der [GNU General Public License](http://www.gnu.org/licenses/gpl.html) weiter verteilen und/oder verändern. Sowohl die neueste Version des Programms wie auch der Quelltext können von <http://www.padowan.dk> heruntergeladen werden.

Das Programm ist unter Windows 7, Windows 8 und Windows 10 getestet worden, kann aber trotzdem noch einige Programmfehler enthalten. Im [Graph-Support-Forum](http://www.padowan.dk/forum) erhalten Sie Unterstützung, können Fehler melden und Verbesserungsvorschläge machen.

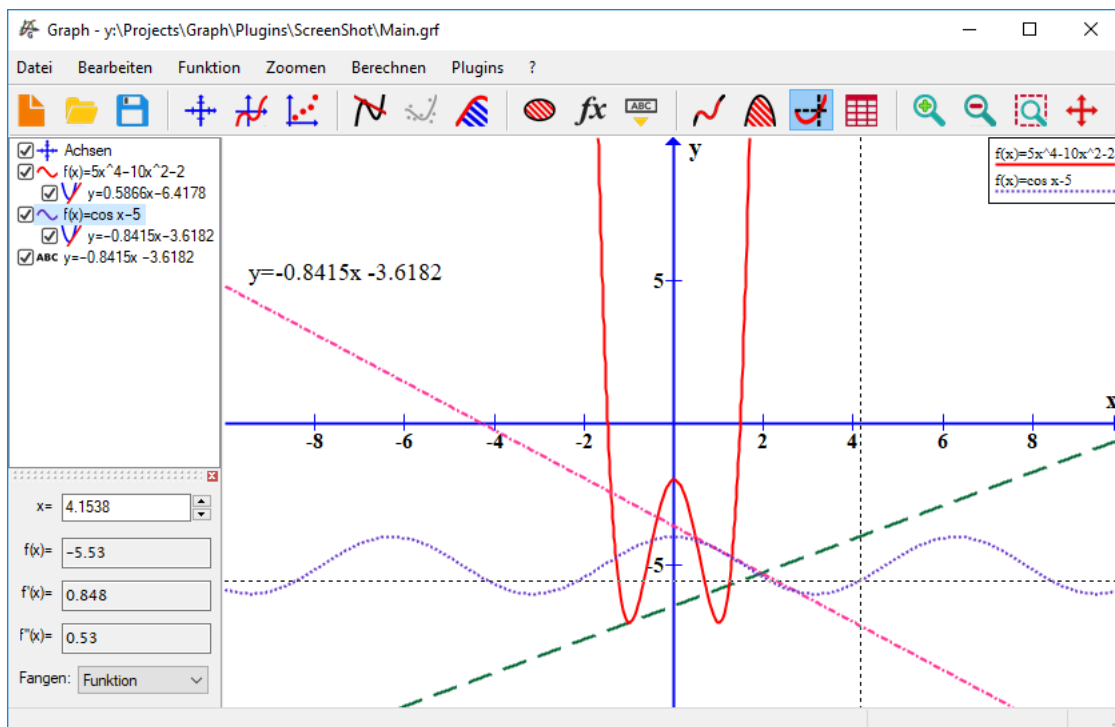
Wenn Sie eine Fehlermeldung senden, beachten Sie bitte die folgenden Aspekte:

- Welche Version verwenden Sie? Dies können Sie im Dialog ? → Über Graph nachsehen. Bitte überprüfen Sie, ob die von Ihnen verwendete Version von Graph auf dem aktuellsten Stand ist, denn der aufgetretene Fehler könnte in der neuesten Version bereits behoben sein.
- Erklären Sie, was passiert und was Sie erwartet haben.
- Erklären Sie genau, wie ich den Fehler reproduzieren kann. Wenn ich nicht dasselbe sehe wie Sie, ist es sehr schwierig für mich, das Problem zu lösen.

Zur Benutzung von Graph

Wenn Graph gestartet wird, erscheint das unten abgebildete Fenster. In diesem Fenster befindet sich auf der rechten Seite eine Zeichnungsfläche, die das Koordinatensystem beinhaltet, in das später alle möglichen Graphen und Objekte eingefügt werden können. Mithilfe von Menü bzw. einzelnen Schaltflächen der Werkzeugleiste können Sie diverse Dialoge aufrufen, um Funktionen neu zu erstellen, bestehende Funktionen zu bearbeiten, zu löschen etc. Hier finden Sie eine [Beschreibung](#) aller verfügbaren Menüpunkte.

Die Werkzeugleiste kann angepasst werden, indem man mit der rechten Maustaste auf die Leiste klickt und **Symbolleiste anpassen...** aus dem Kontextmenü wählt. Verschiedene Befehle können per Drag and Drop auf der Werkzeugleiste platziert und wieder von ihr entfernt werden. Die Statusleiste am unteren Rand des Fensters enthält auf der linken Seite hilfreiche Hinweise und Informationen und zeigt auf der rechten Seite die Koordinatenposition des Mauszeigers.



Mit Hilfe des Funktion Menüs können Sie dem Koordinatensystem neue Elemente hinzufügen. Möchten Sie beispielsweise eine neue Funktion hinzufügen, deren Graph gezeichnet werden soll, wählen Sie den Menüpunkt **Funktion → Funktion einfügen...**

Die links angebrachte *Funktionsliste* zeigt alle von Ihnen angelegten Funktionen, Tangenten, Punktserien, Schraffuren und Relationen. Möchten Sie ein Element aus der Funktionsliste bearbeiten, klicken Sie es einfach an und verwenden Sie dann das Menü **Funktion** oder klicken Sie mit der rechten Maustaste und wählen Sie den entsprechenden Punkt aus dem Kontextmenü aus. Wollen Sie ein Element bearbeiten, können Sie auch einfach darauf einen Doppelklick ausführen.

Das Menü **Berechnen** beinhaltet verschiedene Möglichkeiten, Berechnungen für eine bestimmte Funktion ausführen zu lassen, beispielsweise die Berechnung des Funktionswerts für einen bestimmten Wert für x oder die Aufstellung einer Wertetabelle.

Installation und erstes Starten

Installation

Das Programm wird in einem selbstextrahierendem Installations-Programm ausgegeben, üblicherweise unter dem Namen SetupGraph-x.y.exe, wobei x.y der Versionsname ist. Um Graph zu installieren, führen Sie einfach die Installation aus und folgen Sie den Anweisungen. Dabei werden die folgenden Dateien im ausgewählten Verzeichnis und Unterverzeichnissen installiert:

Datei(en)	Beschreibung:
Graph.exe	Die Programm-Datei.
PDFlib.dll	Bibliothek, die zur Erstellung von PDF-Dateien eingesetzt wird.
Thumbnails.dll	Shell extension für die Vorschau-Anzeige von grf-Dateien im Explorer.
Locale*.mo	Programmübersetzung.
Help*.chm	Hilfe-Dateien in verschiedenen Sprachen.
Plugins*.py	Einige Beispiele für Plug-ins. Benutzerdefinierte Plug-ins können ebenfalls hier gespeichert werden.
Lib*.py	Bibliotheken, die von Plugins benötigt werden.
Examples*.grf	Ein paar Beispiele, die mit Graph geöffnet werden können.

Die Installation erzeugt im Start-Menu eine Verknüpfung, mit der man das Programm starten kann. Während der Installation wählen Sie Ihre bevorzugte Sprache. Diese lässt sich später im [Einstellungen](#)-Dialog ändern.

Falls bereits eine ältere Programmversion installiert ist, wird für die Installation das bisherige Verzeichnis vorgeschlagen. Sie können einfach über die ältere Version installieren und müssen diese nicht erst deinstallieren. Aber vergewissern Sie sich, daß die alte Version während der Installation nicht läuft.

Das Graph-Setup kann die in der untenstehenden Tabelle angegebenen Parameter übernehmen. Diese sind besonders nützlich, wenn Sie die Installation automatisieren wollen.

Parameter	Beschreibung:
<i>/SILENT</i>	Das Setup ist "leise": Der Installations-Fortschritt wird angezeigt, nicht aber der Assistent und das Hintergrundfenster. Alles andere, wie z. B. Fehlermeldungen während der Installation, wird angezeigt. Ist ein Neustart nötig, wird ein Nachrichtenfenster <i>Jetzt Neustart?</i> angezeigt.
<i>/VERYSILENT</i>	Das Setup ist "sehr leise": Dasselbe Verhalten wie "leise", jedoch wird zusätzlich der Installations-Fortschritt nicht mehr angezeigt. Ist ein Neustart nötig, wird das Setup diesen ohne Nachfrage durchführen.
<i>/NORESTART</i>	Das Setup wird nicht neu starten, auch wenn es notwendig ist.
<i>/LANG=language</i>	Legt die Sprache fest. <i>language</i> gibt das englische Wort für die Sprache an. Bei einem gültigen Parameter <i>/LANG</i> wird der <i>Sprache wählen</i> -Dialog unterdrückt.
<i>/DIR=x:\dirname</i>	Überschreibt den voreingestellten Verzeichnisnamen, der im Assistenten unter <i>Wählen Sie den Zielort</i> angezeigt wird. Hier muß ein geeigneter Pfadname angegeben werden.

Deinstallation

Die Deinstallation erfolgt über das Deinstallationsprogramm, das mit Graph installiert wird. Dadurch werden alle Spuren des Programms entfernt. Wenn nach der Installation Dateien in das Installationsverzeichnis aufgenommen wurden, werden Sie gefragt, ob Sie diese löschen möchten. Stellen Sie sicher, dass Graph während der Deinstallation nicht läuft.

Hochfahren

Normalerweise startet man Graph mit dem Link im Start-Menü. Wenn eine .grf-Datei als Parameter übergeben wird, öffnet Graph diese Datei. Zusätzlich können die Parameter in der untenstehenden Tabelle per Kommandozeile an Graph übergeben werden.

Parameter	Beschreibung:
<i>/SI=file</i>	Wird benutzt, um eine geöffnete .grf-Datei als Bild zu speichern. Der Dateityp kann eines der von Graph unterstützten Bildformate sein.
<i>/WIDTH=width</i>	Wird in Kombination mit /SI benutzt, um die Breite des zu speichernden Bildes in Pixeln anzugeben.
<i>/HEIGHT=height</i>	Wird in Kombination mit /SI verwendet, um die Höhe des zu speichernden Bildes in Pixeln anzugeben.

Häufig gestellte Fragen

- F:** Welche Systemanforderungen hat Graph?
- A:** Graph erfordert mindestens Microsoft Windows Vista. Es wurde unter Windows 7, Windows 8 und Windows 10 getestet.
- F:** Läuft Graph unter Linux?
- A:** Graph ist ein Windows-Programm und noch nicht hinreichend unter Linux getestet worden, allerdings haben mich einige Anwender und Übersetzer darüber informiert, dass Graph problemlos mit Hilfe von WINE unter Linux läuft.
- F:** Läuft Graph auf einem Macintosh?
- A:** Wie oben: Graph läuft nicht direkt auf einem Mac. Es ist jedoch ein Graph-Paket mit Wine über den [Web-Seite](http://www.padowan.dk/mac/) [http://www.padowan.dk/mac/] erhältlich.
- F:** Wann kommt die nächste Version?
- A:** Wenn es fertig ist.
- F:** Wie kann ich das Koordinatensystem bewegen?
- A:** Wenn Sie **Ctrl** gedrückt halten, können Sie das Koordinatensystem mit Hilfe der Pfeiltasten verschieben. Mit Hilfe von **Zoomen** → **Koordinatensystem verschieben** können Sie auch die Maus verwenden, um das Koordinatensystem zu bewegen.
- F:** Wie kann man einfach zoomen?
- A:** Wenn Sie **Ctrl** gedrückt halten, können Sie die Ansicht vergrößern und verkleinern (**Zoom**), indem Sie die Tasten **+** bzw. **-** drücken. Denselben Effekt erreichen Sie, wenn Sie stattdessen das Mousrad einsetzen. Graph vergrößert (Mausrad auf) bzw. verkleinert (Mausrad ab) dann das Koordinatensystem und zentriert es auf die Position des Mauszeigers.
- F:** Wie speichert man die Voreinstellungen?
- A:** Die gewünschten Standardeinstellungen können Sie im Dialog [Achsen bearbeiten](#) vornehmen, indem Sie *Als Standard abspeichern* markieren, bevor Sie auf **OK** klicken. Wenn Sie danach ein neues Koordinatensystem anlegen, erstellt Graph es nach den von Ihnen als Standard festgelegten Einstellungen.
- F:** Ist es möglich, dass Graph die Größe und Position des Hauptfensters speichert?
- A:** Wenn Sie [Einstellungen](#) im Dialog *Fenstereinstellungen beim Beenden speichern* auswählen, speichert Graph Position und Größe des Hauptfensters, sobald das Programm beendet wird und startet beim nächsten Mal an entsprechender Position mit dieser Fenstergröße.
- F:** Warum akzeptiert Graph kein Komma bei der Eingabe von Dezimalzahlen?
- A:** In einigen Ländern ist das Komma als Dezimaltrennzeichen etabliert, allerdings verwendet Graph unabhängig von den Ländereinstellungen der Systemsteuerung immer den Punkt als Trennzeichen. Das Komma wird hingegen dafür verwendet, verschiedene Argumente einer Funktion voneinander zu trennen.
- F:** Wie kann man mit Graph eine Senkrechte zeichnen?
- A:** Eine Senkrechte (Normale) kann durch eine parametrische Funktion dargestellt werden. Als *Funktionstyp* wählen Sie *Parameterfunktion* beim Hinzufügen der Funktion. Eine Senkrechte an der

Stelle $x=5$ kann durch $\mathbf{x}(t)=5$, $\mathbf{y}(t)=t$ erstellt werden. Eine andere Möglichkeit besteht darin, sie als Relation hinzuzufügen: $\mathbf{x}=5$.

F: Wie kann man mit Graph eine Funktion $x=f(y)$ zeichnen?

A: Um den Graphen einer Funktion mit y als unabhängiger Variable darstellen lassen zu können, muss die Funktion in parametrischer Form angegeben werden. Wählen Sie *Funktionsstyp* unter *Parameterfunktion*, wenn Sie die Funktion hinzuzufügen. Wollen Sie den Graphen der Funktion $x=\sin(y)$ zeichnen lassen, wäre die entsprechende parametrische Form $\mathbf{x}(t)=\sin(t)$, $\mathbf{y}(t)=t$. Eine andere Möglichkeit besteht darin die Relation $\mathbf{x}=\sin(\mathbf{y})$ einzugeben.

F: Wie lasse ich Graph einen Kreis zeichnen?

A: Mit Hilfe einer Parameterfunktion können Sie Graph einen Kreis zeichnen lassen. Wenn Sie die Funktion anlegen, wählen als *Funktionsstyp Parameterfunktion*. Um Kreis mit dem Punkt (2,3) als Mittelpunkt und mit dem Radius 5 zeichnen zu lassen, könnten Sie beispielsweise diese Parameterfunktion eingeben: $\mathbf{x}(t)=5\cos(t)+2$, $\mathbf{y}(t)=5\sin(t)+3$. Damit der Kreis auch wie ein Kreis aussieht (und nicht wie eine Ellipse), müssen Sie möglicherweise mit Hilfe von **Zoomen** → **Quadrat** die Achsen auf die gleiche Skalenteilung bringen. Einen Kreis um den Koordinatenursprung, also um den Punkt (0,0), kann man auch mit Hilfe einer Polarfunktion zeichnen lassen. Soll dessen Radius 5 betragen, wäre die Polarfunktion $\mathbf{r}(t)=5$ anzulegen. Eine weitere Möglichkeit besteht darin, eine Relation zu verwenden. Einen Kreis mit Mittelpunkt in (2,3) mit Radius 5 würde dann auf Basis der folgenden Relation gezeichnet: $(\mathbf{x}-2)^2+(\mathbf{y}-3)^2=5^2$.

F: Warum sehen Kreise nicht rund, sondern elliptisch aus?

A: Die wahrscheinlichste Erklärung dafür ist, dass die Koordinatenachsen nicht dieselbe Skalierung aufweisen. Entweder passen Sie die Größe des Fensters an, bis die Skalierung für beide Achsen identisch ist oder Sie wählen **Zoomen** → **Quadrat** im Menü, um X- und Y-Achse gleich zu skalieren.

F: Wie lasse ich Graph die Fläche zwischen zwei Funktionsgraphen berechnen?

A: Wenn Sie die von zwei Funktionsgraphen eingeschlossene Fläche bestimmen wollen, gehen Sie wie folgt vor. Angenommen, es handelt sich um die Funktionen $f_1(x)=3x$ und $f_2(x)=x^2$, dann fügen Sie zuerst die Differenzfunktion $f(x)=f_1(x)-f_2(x)=3x-x^2$ der Funktionsliste hinzu. Anschließend können Sie mit **Berechnen** → **Integrieren** die Fläche zwischen den beiden Funktionsgraphen für ein bestimmtes Intervall berechnen lassen.

F: Wie lasse ich Graph die Fläche zwischen zwei Funktionsgraphen berechnen?

A: Hier können Sie eine parametrische Funktion benutzen. Wenn Sie die Inverse von $f(x)=x^2-2x$ zeichnen lassen wollen, verwenden Sie dazu die parametrische Funktion $\mathbf{x}(t)=t^2-2t$, $\mathbf{y}(t)=t$.

F: Wie kann ich Graph den negativen Ast zu $f(x)=\sqrt{x+2}$ zeichnen lassen?

A: Für jeden Wert x des Definitionsbereichs ergibt $f(x)$ genau einen Wert. $\mathbf{f}(x)=\sqrt{x+2}$ ergibt für $f(x)$ entsprechend nur positive Werte. Um bei diesem Beispiel auch einen unteren Parabelast im negativen Bereich zeichnen zu lassen, benötigen Sie zweite separate Funktionen: $\mathbf{f}(x)=\sqrt{x+2}$ und $\mathbf{f}(x)=-\sqrt{x+2}$. Eine andere Möglichkeit besteht darin, die folgende Relation zu verwenden: $\mathbf{y}^2=\mathbf{x}+2$.

F: Wie lasse ich Graph eine komplexe Funktion wie $f(t)=e^{i*t}$ zeichnen?

A: Wahrscheinlich wollen Sie den Realteil auf der X- und den Imaginärteil auf der Y-Achse darstellen. In diesem Fall können Sie die Parameterfunktion $\mathbf{x}(t)=\text{re}(e^{i*t})$, $\mathbf{y}(t)=\text{im}(e^{i*t})$ verwenden. Beachten Sie, dass dafür im Dialog **Achsen bearbeiten** *Komplex berechnen* aktiviert sein muss.

F: Was muss ich tun, damit Graph Funktionen mit senkrechten Asymptoten richtig zeichnet?

- A:** Funktionen, die wie beispielsweise $f(x) = \tan(x)$ vertikale Asymptoten aufweisen, werden möglicherweise nicht immer korrekt dargestellt. Standardmäßig berechnet Graph für jeden Wert der X-Achse einen zugehörigen Funktionswert, was bei sehr steilen (gegen unendlich gehenden) Steigungen zu Darstellungsproblemen führen kann. Um Funktionsgraphen mit derartigen Steigungen korrekt zeichnen zu lassen, kann eingestellt werden, wie viele Werte Graph für die graphische Darstellung berechnen soll. Im Feld *Schritte* des Dialogs [Funktion einfügen](#) wird diese Anzahl festgelegt. Ein Wert um 100.000 sollte das Problem normalerweise aus dem Weg schaffen und den Funktionsgraphen korrekt darstellen.
- F:** Wie kann ich aus Graph heraus PDF-Dateien exportieren?
- A:** Im Dialog [Als Bilddatei abspeichern](#) ist es möglich, das Koordinatensystem als PDF-Datei abzuspeichern.

OLE-Server/Client

OLE-Server

Graph ist als OLE-Server (Object Linking und Embedding) implementiert; d. h. daß Graph-Objekte in andere OLE-fähige Applikationen eingebettet werden können, z. B. auch in Microsoft Word.

Mit **Bearbeiten** → **Koordinatensystem kopieren** läßt sich der aktuelle Inhalt in die Zwischenablage kopieren. Später können Sie das Graph-Objekt aus der Zwischenablage einfügen, z. B. mit **Einfügen** in Word (o. ä. in einem anderem OLE-Client). Beim Doppelklick auf das Objekt startet eine neue Instanz von Graph, mit der Sie das Objekt bearbeiten können. Wenn Sie die Daten nicht als Graph-Objekt einfügen wollen, sondern als Bild, können Sie dazu in Word **Einfügen** → **Spezielles Einfügen...** verwenden.

Sie können eine neues Graph-Objekt in Word erzeugen, indem Sie bei **Objekt** als *Objekttyp Graph-Koordinatensystem* wählen. Mit demselben Dialog läßt sich ein eingebettetes Graph-Objekt aus einer grf-Datei erzeugen. Mit *Mit Datei verknüpfen* erhalten Sie ein verlinktes statt eines eingebetteten Objekts. Auf diese Art werden alle Änderungen des Objekts auch in der ursprünglichen grf-Datei umgesetzt. Ist die grf-Datei nicht verfügbar, können Sie das Objekt zwar nicht bearbeiten, aber das Bild ist in Word noch sichtbar.

Um ein Graph-Objekt zu bearbeiten, muß Graph installiert sein. Wenn Graph nicht installiert ist, können Sie zwar noch das Bild sehen, nicht aber bearbeiten.

OLE-Client

Graph kann als OLE-Client arbeiten, da ein Textfeld in Graph ein OLE-Container ist. Dies bedeutet, daß Sie in den Editor auch Bilder und OLE-Objekte einfügen können. Wie in allen anderen OLE-Containern läßt sich das Objekt per Doppelklick bearbeiten. Vom Kontext-Menü aus können Sie mit **Objekt einfügen...** ein neues OLE-Objekt im Textfeld erzeugen. Mit demselben Dialog läßt sich ein Objekt aus einer Datei erzeugen. Sie können so z. B. eine Bild-Datei einfügen. Um ein OLE-Objekt zu bearbeiten, muß der Server installiert sein, ansonsten können Sie das Objekt nur sehen, nicht aber bearbeiten.

Liste der Menüpunkte

Das Folgende ist eine Liste aller Menüpunkte im Programm:

Datei → Neu (Strg+N)

Verwenden Sie dies, um ein ein neues Koordinatensystem zum Zeichnen von Graphen zu erzeugen.

Datei → Öffnen... (Strg+O)

Liest ein schon gespeichertes Koordinatensystem aus einer .grf-Datei ein.

Datei → Speichern (Strg+S)

Speichert das Koordinatensystem in eine Datei.

Datei → Speichern unter...

Speichert das Koordinatensystem mit neuem Namen in eine Datei.

Datei → Als Bilddatei abspeichern... (Strg+B)

Speichert das Koordinatensystem als Bilddatei (*.emf, *.bmp oder *.png).

Datei → Importieren → Graph-Datei...

Importiert den Inhalt einer anderen Graph-Datei ins aktuelle Koordinatensystem.

Datei → Importieren → Punktserie(n)...

Importiert eine oder mehrere Punktserien aus einer Textdatei, wobei die Daten durch Tabs, Kommata oder Semikola getrennt werden. Die erste Spalte muss die X-Koordinaten, die folgenden Spalten Y-Koordinaten enthalten. Graph erzeugt sovieler Punktserien, wie die Datei Spalten mit Y-Koordinaten enthält. Die Anzahl der Punktserien in der Datei ist nicht begrenzt, solange sie dieselben X-Koordinaten gemeinsam haben.

Datei → Drucken... (Strg+P)

Sendet das Koordinatensystem und die Graphen an einen Drucker.

Datei → Beenden (Alt+F4)

Beendet das Programm. Ggfs. bittet es Sie, die Datei zu speichern.

Bearbeiten → Rückgängig (Strg+Z)

Hiermit läßt sich Ihre letzte Aktion rückgängig machen. Sie können die maximale Anzahl der im [Einstellungen](#)-Dialog gespeicherten Rückgängig-Schritte einstellen.

Bearbeiten → Wiederherstellen (Strg+Y)

Hiermit stellen Sie Ihre letzte rückgängiggemachte Aktion wieder her; vorausgesetzt, sie haben **Bearbeiten → Rückgängig** ausgewählt.

Bearbeiten → Ausschneiden (Strg+X)

Dies kopiert das ausgewählte *Graph-Element* in die Zwischenablage. Danach wird das Element gelöscht.

Bearbeiten → Kopieren (Strg+C)

Dies kopiert das ausgewählte *Graph-Element* in die Zwischenablage.

Bearbeiten → Einfügen (Strg+V)

Dies fügt ein vorher kopiertes *Graph-Element* von der Zwischenablage ins Koordinatensystem.

Bearbeiten → Koordinatensystem kopieren (Strg+I)

Kopiert das gezeigte Koordinatensystem als Bild in die Zwischenablage. Sie können es dann ein anderes Programm einfügen, z. B. Microsoft Word.

Bearbeiten → Achsen... (Strg+A)

Bearbeitet die Eigenschaften der Achsen, z. B. Maßstab, Farben, Platzierung der Legende usw.

Bearbeiten → Einstellungen...

Ändert allgemeine Einstellungen für Graph, z. B. Zuordnung von .grf Dateien, Anzeigen von Tooltips, maximale Anzahl gespeicherter Rückgängig-Schritte usw.

Funktion → Funktion einfügen... (Einf)

Fügt eine Funktion ins Koordinatensystem ein. Die Funktionen können unterschiedliche Breite und Farbe haben, Sie können den Graphen in einem angegebenen Intervall anzeigen und auch andere Dinge einstellen.

Funktion → Tangente einfügen... (F2)

Dieser Dialog dient dazu, eine Tangente an eine bestehende Funktion in einem bestimmten Punkt anzulegen. Die Tangente wird in der *Funktionsliste* als separate Funktion abgelegt.

Funktion → Schraffur einfügen... (F3)

Mit diesem Menüpunkt läßt sich zur selektierten Funktion eine Schraffur hinzufügen. Sie können verschiedene Schraffur-Arten und -Farben wählen. Die Schraffuren lassen sich über der Funktion, unter der Funktion, zwischen Funktion und X-Achse, zwischen Funktion und Y-Achse, innerhalb der Funktion und zwischen zwei Funktionen hinzufügen.

Funktion → Ableitungsfunktion $f'(x)$ einfügen... (F7)

Mit diesem Dialog läßt sich die erste Ableitung der selektierten Funktion erstellen.

Funktion → Punktserie einfügen... (F4)

Fügt eine neue Punktserie ins Koordinatensystem ein, das ist eine unendliche Anzahl von Punkten, definiert durch ihre X- und Y-Koordinaten. Farbe, Größe und Art der Punktserie lassen sich auswählen.

Funktion → Trendlinie einfügen... (Strg+T)

Fügt eine Trendlinie ein; das ist die Funktion, die am besten zur ausgewählten Punktserie paßt. Sie können zwischen verschiedenen Funktionsarten für die Trendlinie wählen.

Funktion → Relation einfügen... (F6)

Fügt eine Gleichung oder Ungleichung ins Koordinatensystem ein. Mit Gleichungen und Ungleichungen drückt man Relationen zwischen X- und Y-Koordinaten mit denselben Operatoren aus, wie z. B. für Funktionsgraphen. Relationen können mit verschiedenen Schraffurtypen und -Farben hinzugefügt werden.

Funktion → Textfeld einfügen... (F8)

Dies öffnet einen Dialog, mit dem man ein formatiertes Textfeld erzeugen kann. Das Textfeld wird immer in der Mitte der Zeichenfläche erzeugt, kann aber später mit der Maus verschoben werden.

Funktion → Bearbeiten... (Eingabe)

Dies öffnet einen Dialog, in dem Sie das ausgewählte *Graph-Element* in der *Funktionsliste* ändern können.

Funktion → Löschen (Entf)

Dies löscht das ausgewählte *Graph-Element* in der *Funktionsliste*.

Funktion → Benutzerdefinierte Funktionen... (Strg+F)

Dies öffnet einen Dialog, in dem Sie - zusätzlich zu den eingebauten - benutzerdefinierte Funktionen und Konstanten erzeugen können.

Zoomen → Vergrößern (Strg++)

Hiermit zoomen Sie in die Mitte der Zeichenfläche, so dass Sie $\frac{1}{4}$ der vorigen Zeichenfläche sehen.

Zoomen → Verkleinern (Strg+-)

Damit zoomen Sie soweit heraus, dass der Bildausschnitt nun um den Faktor 1,23 mehr anzeigt als zuvor. Wenn Sie die **Umschalt**-Taste gedrückt halten, vergrößert sich der Bildausschnitt um den Faktor 4.

Zoomen → Fenster (Strg+W)

Halten Sie die linke Maustaste gedrückt, während Sie den Bereich wählen, der die ganze Zeichenfläche ausfüllen soll. Mit einem Rechtsklick oder **Esc** wird der Befehl abgebrochen.

Zoomen → Quadrat (Strg+Q)

Hiermit erhält die Y-Achse denselben Maßstab wie die X-Achse. Ein Kreis wird nun korrekt dargestellt statt als Ellipse. Die Achsenskalierung bleibt bis zur nächsten Änderung bestehen.

Zoomen → Standard (Strg+D)

Setzt die Achsen-Einstellungen auf die Voreinstellungen zurück (mit denen man ein neues Koordinatensystem erstellt).

Zoomen → Koordinatensystem verschieben (Strg+M)

Wenn ausgewählt, wird der Cursor zur Hand. Sie können mit der Maus das Koordinatensystem umherschoben. Mit einem Rechtsklick, **Esc** oder erneutes Anwählen des Menüpunkts kehren sie in den Normalmodus zurück. Alternativ zu diesem Menüpunkt lässt sich das Koordinatensystem auch bei gedrückter **Shift**-Taste umherschoben.

Zoomen → Passend

Die Achseneinstellungen werden so angepaßt, daß alle Teile des ausgewählten *Graph-Element* angezeigt werden.

Zoomen → Passend

Die Achseneinstellungen werden so angepaßt, daß alle Elemente in der *Funktionsliste* komplett angezeigt werden.

Berechnen → Länge eines Kurvenstücks

Berechnet die Entfernung entlang der Kurve zwischen zwei Punkten des ausgewählten Graphen.

Berechnen → Integrieren

Berechnet das bestimmte Integral für ein gegebenes Intervall. Das entspricht der Fläche (Vorzeichen beachten!) zwischen Funktionsgraph und X-Achse.

Berechnen → Evaluieren (Strg+E)

Die ausgewählte Funktion wird an einem gegebenen Punkt untersucht. Bei Standard-Funktionen werden $f(x)$, $f'(x)$ und $f''(x)$ ausgewertet. Bei Parameterfunktionen werden $x(t)$, $y(t)$, dx/dt , dy/dt und dy/dx ausgewertet. Bei Polar-Funktionen werden $r(t)$, $x(t)$, $y(t)$, dr/dt und dy/dt ausgewertet.

Berechnen → Tabelle...

Dieser Dialog bewertet einen vom Benutzer angegebenen Wertebereich und schreibt diese Werte und deren Auswertung in eine Tabelle.

Berechnen → Animieren

Mit diesem Dialog können Sie eine Animation der Daten im Koordinatensystem erzeugen, indem Sie eine bestehende benutzerdefinierte Konstante ändern. Damit sieht man leicht, was passiert, wenn sich die Konstante ändert. Die Animation kann in eine Datei gespeichert werden.

Plugins

An dieser Stelle werden typischerweise Menüpunkte zur Aktivierung des Plugins platziert. Das Menü ist unsichtbar, wenn keine Plugins vorhanden sind oder das **Plugin System** nicht verfügbar ist.

? → Inhalt und Index (F1)

Zeigt Inhalt und Index der Hilfe-Datei an.

? → Funktionsliste (**Strg+F1**)

Zeigt eine Liste der Funktionen und Konstanten an, die für das Zeichnen von Graphen benutzt werden können.

? → Häufig gestellte Fragen

Zeigt eine Liste der häufig gestellten Fragen und deren Antworten an.

? → Internet → Graph-Homepage

Geht mit Ihrem voreingestellten Browser auf die Graph-Website.

? → Internet → Unterstützung

Geht mit Ihrem voreingestellten Browser ins Graph-Support-Forum.

? → Internet → Spenden

Geht mit Ihrem voreingestellten Browser auf die Website, auf der Sie das Graph-Projekt mit einer Spende unterstützen können.

? → Internet → Nach Aktualisierung suchen

Prüft, ob eine neue Version von Graph verfügbar ist. Falls ja, werden Sie gefragt, ob Sie die Graph-Website für einen Download der neuen Version besuchen wollen.

? → Über Graph (**Alt+F1**)

Zeigt Versionsnummer, Copyright und Lizenzinformationen für Graph an.

Tastenkürzel

Umschalt+Ziehen

Damit bewegen Sie das Koordinatensystem, im Grunde genau so wie Sie Zoomen → Koordinatensystem verschieben im Menü auswählen.

Mausrad

Benutzen Sie das Mausrad um aus dem Koordinatensystem auf Position des Mauszeigers hinein- und herauszuzoomen.

Strg+Pfeiltaste

Wenn Sie die **Strg**-Taste gedrückt halten, können Sie das Koordinatensystem mit Hilfe der Pfeiltasten in kleinen Schritten verschieben. Für größere Schritte halten Sie zusätzlich die Taste **Umsch** gedrückt.

Strg+Pos 1

Damit vergrößern Sie die X-Achse in kleinen Schritten, in größeren Schritten, wenn Sie zusätzlich die **Umschalt**-Taste gedrückt halten.

Strg+Ende

Damit verkleinern Sie die X-Achse in kleinen Schritten, in größeren Schritten, wenn Sie zusätzlich die **Umschalt**-Taste gedrückt halten.

Strg+Bild-Auf

Damit vergrößern Sie die Y-Achse in kleinen Schritten, in größeren Schritten, wenn Sie zusätzlich die **Umschalt**-Taste gedrückt halten.

Strg+Bild-Ab

Damit verkleinern Sie die Y-Achse in kleinen Schritten, in größeren Schritten, wenn Sie zusätzlich die **Umschalt**-Taste gedrückt halten.

F11

Zeigt oder verbirgt den [Python-Interpreter](#), wenn er verfügbar ist.

Fehlermeldungen

Fehler 01: Bei der Berechnung der Potenzfunktion trat ein Fehler auf.

Dieser Fehler tritt auf, wenn die Berechnung einer Potenz einen nicht definierten Wert ergibt. Beispielsweise führt $(-4)^{(5.1)}$ zu diesem Fehler, weil eine negative Zahl nicht mit einer anderen negativen Zahl potenziert werden darf, wenn mit *reelle Zahlen* gerechnet wird.

Fehler 02: Tangens für $90^\circ + p180^\circ$ ist nicht definiert.

Der Tangens $\tan(x)$ ist für $x = \pi/2 + \pi p = 90^\circ + p180^\circ$, wobei p eine ganze Zahl ist, nicht definiert.

Fehler 03: Die Fakultät kann nur für positive Zahlen berechnet werden.

Den Ausdruck, den Sie definieren wollen, geben Sie in der zweiten Spalte ein. Der Ausdruck kann die Argumente, die in der ersten Spalte definiert wurden, sowie alle eingebauten Funktionen und übrigen benutzerdefinierten Funktionen und Konstanten beinhalten und sich sogar selbst rekursiv aufrufen. Durch das Symbol # kann am Ende eines Ausdrucks auch ein Kommentar hinzugefügt werden.

Fehler 04: Logarithmus ist für Zahlen kleiner gleich 0 nicht definiert.

Die Logarithmusfunktionen $\ln(x)$ und $\log(x)$ sind für $x \neq 0$ nicht definiert, wenn reelle Zahlen die Rechengrundlage bilden. Werden komplexe Zahlen verwendet, sind die Funktionen lediglich für $x=0$ nicht definiert.

Fehler 05: Die Quadratwurzel ist für negative Zahlen undefiniert.

Die Wurzelfunktion \sqrt{x} ist für $x < 0$ nicht definiert, sofern mit reellen Zahlen gerechnet wird. Bei Berechnungen mit komplexen Zahlen besteht diese Beschränkung nicht.

Fehler 06: Ein Teil der Berechnung ergab eine Zahl mit Imaginärteil.

Dieser Fehler kann bei Berechnungen auftreten, wenn nur reelle Zahlen zulässig sind. Ergibt ein Teil der Berechnung eine Zahl mit Imaginärteil, kann die weitere Berechnung nicht fortgesetzt werden. Beispiel: $\sin(x+i)$

Fehler 07: Division durch 0.

Bei der Berechnung wurde versucht, durch null zu dividieren. In einem solchen Fall ist der Funktionswert nicht definiert. Beispiel: $f(x)=1/x$. Diese Funktion ist an der Stelle $x=0$ nicht definiert (Pol mit Vorzeichenwechsel).

Fehler 08: Inverse trigonometrische Funktion außerhalb des Intervalls $[-1;1]$

Die trigonometrischen Umkehrfunktionen Arcussinus $\arcsin(x)$ und Arcuscosinus $\arccos(x)$ sind nur auf dem Intervall $[-1;1]$ definiert. Ferner sind sie für beliebige Zahlen mit Imaginärteil nicht definiert. Die Funktion Arcustangens $\arctan(x)$ ist für alle Zahlen ohne Imaginärteil definiert. Dieser Fehler kann auch auftreten, wenn versucht wird, $\arg(0)$ zu berechnen.

Fehler 09: Die Funktion ist für diesen Wert nicht definiert.

Dieser Fehler kann bei Funktionen auftreten, die an einer bestimmten Stelle nicht definiert sind. Beispiele: $\sin(x)$ und $u(x)$ sind an der Stelle $x=0$ nicht definiert.

Fehler 10: atanh wurde für einen undefinierten Wert berechnet.

Die hyperbolische Umkehrfunktion Areatangens Hyperbolicus $\operatorname{atanh}(x)$ ist für $x=1$ nicht und nur dann definiert, wenn komplexe Zahlen außerhalb des Intervalls $]-1;1[$ verwendet werden.

Fehler 11: acosh wurde für einen undefinierten Wert berechnet.

Die hyperbolische Umkehrfunktion Areacosinus Hyperbolicus $\operatorname{acosh}(x)$ ist für *reelle Zahlen* nur für $x \geq 1$ definiert. $\operatorname{acosh}(x)$ ist für alle Zahlen definiert, wenn mit *Komplexe Zahlen* gerechnet wird.

Fehler 12: $\arg(0)$ ist undefiniert.

Das Argument von Null ist undefiniert, da die Null (bei der Polardarstellung komplexer Zahlen) keinen eindeutigen Winkel besitzt.

Error 13: Berechnung fehlgeschlagen.

Dieser Fehler tritt auf, wenn eine kompliziertere Funktion wie $W(z)$ berechnet wird und die Berechnung kein genaues Ergebnis liefert.

Fehler 14: Argument erzeugte ein unbestimmtes Funktionsergebnis.

Ein Funktionsargument hat einen Überlauf verursacht. Beispielsweise tritt dies bei $\sin(1E70)$ auf. Hier erhält man ein zufälliges Ergebnis aus dem Intervall $[-1;1]$.

Fehler 15: Die benutzerdefinierte Funktion/Konstante '%s' wurde nicht gefunden oder hat die falsche Anzahl Argumente.

Eine benutzerdefinierte Funktion oder Konstante ist nicht mehr verfügbar. Sie können sie entweder neu definieren oder von der Verwendung des Symbols absehen. Dieser Fehler kann auch dann auftreten, wenn eine benutzerdefinierte Konstante in einer Funktion umgewandelt wurde oder umgekehrt oder wenn die Anzahl der von einer benutzerdefinierten Funktion erwarteten Argumente verändert wurde.

Fehler 16: Zu viele rekursive Aufrufe

Es wurden zu viele rekursive Aufrufe ausgeführt. Der Fehler wird meist von einer Funktion erzeugt, die sich selbst rekursiv unendlich oft aufruft; z. B. $(x)=2*\text{foo}(x)$. Wenn Sie zu viele Funktionen rekursiv aufrufen, kann der Fehler ebenfalls auftreten.

Fehler 17: Überlauf: Der Rückgabewert einer Funktion war zu groß.

Ein Funktionsaufruf ergab einen zu großen Rückgabewert. Dies passiert z. B. bei der Berechnung $\sinh(20000)$.

Error 13: Eine Plugin-Funktion ist fehlgeschlagen.

Eine benutzerdefinierte Funktion (Python Plugin) führte zu keinem Ergebnis. Im Fenster des Python-Interpreters finden sich möglicherweise genauere Informationen.

Fehler 50: Unerwarteter Operator. Operator %s können hier nicht stehen.

Ein Rechenoperator $+$, $-$, $*$, $/$ oder $^$ wurde falsch gesetzt. Beispiel: $f(x)=^2$. Meist deutet dies darauf hin, dass vor dem Operator ein Element fehlt.

Fehler 55: Rechte Klammer fehlt.

Eine Klammer fehlt. Stellen Sie sicher, dass Sie jede Klammer, die Sie öffnen auch wieder korrekt schließen.

Fehler 56: Ungültige Parameteranzahl für die Funktion '%s'.

Sie haben der angegebenen Funktion eine falsche Zahl an Argumenten übergeben. Schauen Sie unter [Funktionsliste](#) nach, um die Anzahl der von der Funktion erwarteten Argumente herauszufinden.
Beispiel: $\sin(x,3)$

Fehler 57: Vergleichsoperator falsch gesetzt.

Es sind nur zwei Vergleichsoperatoren in Folge erlaubt. Beispiel: " $\sin(x) < y$ %lt; $\cos(x)$ " ist akzeptabel, während " $\sin(x) < x < y < \cos(x)$ " unzulässig ist, da hier drei $<$ -Operatoren in Folge gesetzt wurden.

Fehler 58: Ungültige Zahl gefunden. Benutzen Sie das Format -5.475E-8.

Es wurde eine Zeichenkette gefunden, die nur wie die Beschreibung einer Zahl aussieht, aber keine Zahl ist. Beispiel: 4.5E. Eine reelle Zahl hat allgemein die Form $nnn.fffEeee$, wobei nnn den ganzzahligen Teil der Zahl darstellt, welcher auch negativ werden darf. fff stellt die Nachkommastellen dar, die - entgegen der in Deutschland üblichen Konvention - vom ganzzahligen Teil mit einem Punkt ('.') getrennt werden. Der Nachkommateil kann auch weggelassen werden, wenn es sich um eine ganze Zahl handelt. Umgekehrt kann der Teil vor dem Dezimaltrenner weggelassen werden, wenn es sich um ein Dezimalzahl zwischen null und eins handelt. Das 'E' ist ein Trennzeichen, das gesetzt werden kann, falls Dezimalzahlen in Gleitkommaform dargestellt werden sollen. In diesem Fall gibt die Dezimalzahl vor dem 'E' alle Ziffern der darzustellenden Zahl an (Mantisse), die mit der eee . Potenz von 10 multipliziert eben diese Zahl ergibt. Natürlich kann eee auch ein negatives Vorzeichen haben. Beispielsweise lässt sich $5*10^8$ also verkürzt als $5E8$ darstellen. Weitere Beispiele: $-5.475E-8$, -0.55 , $.75$, $23E4$

#Fehler 59: Zeichenfolge ist leer. Sie müssen eine Formel eingeben.

Es wurde nichts eingegeben. Dies ist nicht erlaubt; es muss ein Ausdruck eingegeben werden.

#Fehler 60: Ein Komma ist hier nicht erlaubt; benutzen Sie den Punkt als Dezimaltrenner.

Kommas dürfen nicht als Dezimaltrenner benutzt werden; verwenden Sie stattdessen den Punkt.

Fehler 61: Unerwartete Endeklammer gefunden.

Eine Endeklammer wurde unerwartet gefunden. Überprüfen Sie, ob Anfangs- und Endklammern zusammenpassen.

Fehler 63: Zahl, Konstante oder Funktion erwartet.

Faktor (Zahl, Konstante, Variable oder Funktion) erwartet.

Fehler 64: Parameter nach Konstante oder Variable verboten.

Klammern dürfen nicht hinter Konstante oder Variablen stehen. Ungültig ist z. B.: $f(x)=x(5)$. Schreiben Sie stattdessen $f(x)=x*5$.

Fehler 65: Ausdruck erwartet.

Ein Ausdruck wurde erwartet. Dies kann bei leeren Klammern passieren: $f(x)=sin()$.

Fehler 66: Unbekannte Variable, Funktion oder Konstante: %s

Sie haben etwas eingegeben, das wie Variable, Konstante oder Funktion aussieht, aber keine ist. Beachten Sie, dass "x5" nicht dasselbe wie "x*5" oder "5x" ist.

Fehler 67: Unbekanntes Zeichen: %s

Ein unbekanntes Zeichen wurde gefunden.

Fehler 68: Unerwartetes Ende des Ausdrucks.

Der Ausdruck wurde unerwartet beendet.

Fehler 70: Fehler bei der Analyse des Ausdrucks

Beim Analysieren des Textes ist ein Fehler aufgetreten. Die Zeichenfolge entspricht keiner gültigen Funktion.

Fehler 71: Eine Berechnung hat einen Überlauf verursacht.

Während der Berechnung ist ein Überlauf aufgetreten. Dies kann passieren, wenn die Zahlen zu groß werden.

Fehler 73: In der Berechnung wurde ein ungültiger Wert verwendet.

Ein ungültiger Wert wurde in einer Berechnung verwendet.

Fehler 74: Nicht genug Punkte für die Berechnung.

Zum Berechnen der Trendlinie gab es nicht genug Punkte. Ein Polynom braucht mindestens einen Punkt mehr als die Ordnung des Polynoms. Ein Polynom 3. Grades braucht mindestens 4 Punkte. Alle anderen Funktionen brauchen wenigstens 2 Punkte.

Fehler 75: Ungültiger Name %s für eine benutzerdefinierte Funktion oder Konstante.

Namen für benutzerdefinierte Funktionen und Konstanten müssen mit einem Buchstaben beginnen und dürfen nur Buchstaben und Ziffern enthalten. Es lassen sich keine Namen benutzen, die schon von eingebauten Funktionen und Konstanten verwendet werden.

Fehler 76: Rekursive Funktion lässt sich nicht differenzieren.

Es ist nicht möglich, eine rekursive Funktion zu differenzieren, weil die Ergebnisfunktion unendlich lang würde.

Fehler 79: Die Funktion %s lässt sich nicht differenzieren.

Die Funktion lässt sich nicht differenzieren, weil ein Funktionsteil keine erste Ableitung hat. Dies ist z. B. der Fall für $\arg(x)$, $\text{conj}(x)$, $\text{re}(x)$ und $\text{im}(x)$.

Fehler 86: Unbekannter Fehler in der Berechnung.

Während der Berechnung entstand ein Fehler, dessen genaue Ursache nicht bekannt ist. Bei diesem Fehler informieren Sie bitte den Programmierer mit einer Beschreibung, wie der Fehler reproduziert werden kann. Evtl. ist er in der Lage, den Fehler zu vermeiden oder die Fehlermeldung zu verbessern.

Fehler 87: Keine Lösung gefunden. Versuchen Sie es mit einer anderen Schätzung oder einem anderen Modell.

Die angegebene Schätzung (evtl. die voreingestellte) führte zu keiner Lösung. Grund dafür kann eine schlechte Schätzung sein, und eine bessere Schätzung kann eine Lösung bringen. Vielleicht passt

aber auch das angegebene Modell nicht zu den Daten; in diesem Fall sollten Sie ein anderes Modell versuchen.

Fehler 88: Kein Ergebnis gefunden.

Kein gültiges Ergebnis gefunden. Bspw. kann der Versuch, eine Trendlinie aus einer Punktserie zu erzeugen, fehlgeschlagen sein oder eine zu berechnende Größe könnte ins Unendliche streben.

Fehler 89: Es kann kein exaktes Ergebnis ermittelt werden.

Graph konnte kein genaues Ergebnis berechnen. Dieses Problem kann beim numerischen Integrieren bei hohen Schätzfehlerwerten vorkommen.

Fehler 99: Innerer Fehler. Bitte teilen Sie dies dem Programmierer mit möglichst vielen Informationen mit.

Ein interner Fehler ist aufgetreten. In diesem Fall hat Graph etwas Unvorhergesehenes getan. Setzen Sie sich bitte mit dem Programmierer in Verbindung und geben Sie ihm so viele Informationen wie nötig, so daß er den Fehler reproduzieren und beheben kann.

Funktionen

Funktionsliste

Im Folgenden ist eine Liste aller Variablen, Konstanten, Operatoren und Funktionen zusammengestellt, die von Graph unterstützt werden. Die aufgeführten Operatoren sind gemäß ihrer Präzedenz absteigend geordnet, d.h. diejenigen Operatoren werden zuerst genannt, die bei der Berechnung Vorrang vor anderen Operatoren haben, die weiter unten stehen (Stichwort: "Punktrechnung vor Strichrechnung"). Durch das Setzen von Klammern (), { } und [] sind zulässig kann auf die Reihenfolge der Berechnung zusätzlich Einfluss genommen werden. Beachten Sie, dass Graph bei den verwendeten Ausdrücke nicht nach Groß- und Kleinschreibung unterscheidet. Die einzige Ausnahme von dieser Regel stellt das e (Euler'sche Zahl) dar. Bei der Gleitkommadarstellung einer *reelle Zahl* ist es zwingend erforderlich, Mantisse und Exponenten durch ein großes 'E' zu trennen.

Konstante	Beschreibung:
x	Die in Standardfunktionen verwendete unabhängige Variable.
t	Die unabhängige Variable bei Parameterfunktionen (Parameter) und bei Polarfunktionen (Polarwinkel).
e	Euler'sche Zahl. In Graph ist sie definiert als $e=2.718281828459045235360287$
π	Die Kreiszahl π , die in Graph als $\pi=3.141592653589793238462643$ definiert ist.
undef	Ergibt immer einen Fehlerwert. Wird benutzt, um anzuzeigen, dass ein Teil einer Funktion nicht definiert ist.
i	Die imaginäre Einheit. Sie ist definiert als $i^2 = -1$ und wird nur dann benötigt, wenn mit komplexen Zahlen gearbeitet wird.
inf	Zeichen für "unendlich". Nur sinnvoll als Argument der integrate Funktion.
rand	Erzeugt eine Zufallszahl zwischen 0 und 1.

Operator	Beschreibung:
Exponentiation (^)	Exponentialfunktion. Beispiel: $f(x)=2^x$
Negation (-)	Negativer Wert eines Faktors. Beispiel: $f(x)=-x$
Logisch NOT (not)	not a ergibt 1, falls a 0 ist, und liefert sonst 0.
Multiplikation (*)	Multipliziert zwei Faktoren. Beispiel: $f(x)=2*x$
Division (/)	Dividiert zwei Faktoren. Beispiel: $f(x)=2/x$
Addition (+)	Addiert zwei Terme. Beispiel: $f(x)=2+x$
Subtraktion (-)	Subtrahiert zwei Terme. Beispiel: $f(x)=2-x$
Größer als (>)	Zeigt an, ob ein Ausdruck größer als ein anderer Ausdruck ist.
Größer gleich (>=)	Zeigt an, ob ein Ausdruck größer oder gleich einem anderen Ausdruck ist.
Kleiner als (<)	Zeigt an, ob ein Ausdruck kleiner als ein anderer Ausdruck ist.
Kleiner gleich (<=)	Zeigt an, ob ein Ausdruck kleiner oder gleich einem anderen Ausdruck ist.
Gleich (=)	Zeigt an, ob zwei Ausdrücke exakt denselben Wert ergeben.
Ungleich (<>)	Zeigt an, ob zwei Ausdrücke nicht exakt denselben Wert ergeben.
Logisches AND (und)	$a \text{ and } b$ ergibt 1, falls a und b ungleich Null sind, und liefert sonst 0.
Logisches OR (oder)	$a \text{ or } b$ ergibt 1, falls a oder b oder beide ungleich Null sind, und liefert sonst 0.
Logisches XOR (exklusiv oder)	$a \text{ xor } b$ ergibt 1, falls entweder a oder b ungleich Null ist, und liefert sonst 0.

Funktion	Beschreibung:
<i>Trigonometrisch</i>	
sin	Liefert den Sinus des Arguments, das die Einheit rad oder Grad haben kann.
cos	Liefert den Cosinus des Arguments, das die Einheit rad oder Grad haben kann.
tan	Liefert den Tangens des Arguments, das die Einheit rad oder Grad haben kann.
asin	Liefert den Arcussinus des Arguments in rad oder Grad.
acos	Liefert den Arcuscosinus des Arguments in rad oder Grad.
atan	Liefert den Arcustangens des Arguments in rad oder Grad.
sec	Liefert den Sekans des Arguments, das die Einheit rad oder Grad haben kann.
csc	Liefert den Cosekans des Arguments, das die Einheit rad oder Grad haben kann.
cot	Liefert den Cotangens des Arguments, das die Einheit rad oder Grad haben kann.
asec	Liefert den Arcussecans des Arguments in rad oder Grad.
acsc	Liefert den Arcuscosecans des Arguments in rad oder Grad.
acot	Liefert den Arcuscotangens des Arguments in rad oder Grad.
<i>Hyperbolicus</i>	
sinh	Liefert den Sinus Hyperbolicus des Arguments.
cosh	Liefert den Cosinus Hyperbolicus des Arguments.
tanh	Liefert den Tangens Hyperbolicus des Arguments.
asinh	Liefert den Arkussinus Hyperbolicus des Arguments.
acosh	Liefert den Arkuscosinus Hyperbolicus des Arguments.
atanh	Liefert den Arkustangens Hyperbolicus des Arguments.
csch	Liefert den Cosecans Hyperbolicus des Arguments.
sech	Liefert den Secanshyperbolicus des Arguments.
coth	Liefert den Cotangens Hyperbolicus des Arguments.
acsch	Liefert den Arkuscosecans Hyperbolicus des Arguments.
asech	Liefert den Arkussecans Hyperbolicus des Arguments.
acoth	Liefert den Arkuscotangens Hyperbolicus des Arguments.
<i>Potenz und Logarithmus</i>	
sqr	Liefert das Quadrat des Arguments zurück, also hoch zwei.
exp	Erhebt die Euler'schen Zahl e zur angegebenen Potenz.
sqrt	Liefert die Quadratwurzel des Arguments.
root	Liefert die n-te Wurzel des Arguments.
ln	Liefert den natürlichen Logarithmus des Arguments zurück.
log	Liefert den 10er-Logarithmus des Arguments zurück.
logb	Liefert den Logarithmus mit der Basis n des Arguments zurück.
<i>Komplex</i>	
abs	Liefert den Absolutwert des Arguments zurück.
arg	Liefert den Winkels des Arguments in rad oder Grad zurück.
conj	Ergibt die konjugiert komplexe Zahl des Arguments.
re	Liefert den Realteil des Arguments zurück.
im	Liefert den Imaginärteil des Arguments zurück.

Funktion	Beschreibung:
<i>Runden</i>	
<code>trunc</code>	Liefert den ganzzahligen Anteil des Arguments zurück.
<code>fract</code>	Liefert den gebrochenen Anteil des Argumentes.
<code>ceil</code>	Rundet das Argument zur nächsten Ganzzahl auf.
<code>floor</code>	Rundet das Argument zur nächsten Ganzzahl ab.
<code>round</code>	Rundet das erste Argument auf die Anzahl der Dezimalstellen aus dem zweiten Argument.
<i>Stückweise</i>	
<code>sign</code>	Liefert das Vorzeichen des Arguments zurück: 1, wenn das Argument größer als 0 ist und -1, wenn es kleiner als 0 ist.
<code>u</code>	Einheitssprung: Liefert 1 zurück, wenn das Argument größer gleich als 0 ist, sonst 0.
<code>min</code>	Liefert das kleinste Argument zurück.
<code>max</code>	Liefert das grösste Argument zurück.
<code>range</code>	Liefert das zweite Argument zurück, wenn es im Bereich zwischen dem ersten und dritten Argument ist.
<code>if</code>	Liefert das zweite Argument zurück, wenn das erste Argument nicht 0 ergibt, sonst das dritte Argument.
<i>Speziell</i>	
<code>integrate</code>	Liefert das numerische Integral des ersten Argumentes vom zweiten Argument bis zum dritten Argument zurück.
<code>sum</code>	Ergibt die Summe des ersten Arguments, berechnet für jede Ganzzahl im Bereich vom zweiten bis zum dritten Argument.
<code>product</code>	Ergibt das Produkt des ersten Argumentes, berechnet für jede Ganzzahl im Bereich vom zweiten bis zum dritten Argument.
<code>fact</code>	Liefert die Fakultät des Arguments zurück.
<code>gamma</code>	Liefert the Eulersche Gammafunktion des Arguments zurück.
<code>beta</code>	Liefert die für die Argumente berechnete Betafunktion zurück.
<code>W</code>	Liefert die für die Argumente berechnete Lambertsche W-Funktion zurück.
<code>zeta</code>	Liefert die für die Argumente berechnete Riemannsches Zetafunktion zurück.
<code>mod</code>	Liefert den Rest der Division des ersten Arguments durch das zweite Argument.
<code>dnorm</code>	Liefert die Normalverteilung des ersten Arguments, optional mit Erwartungswert und Standardabweichung.

Beachten Sie folgende Abhängigkeiten:

- $\sin(x)^2 = (\sin(x))^2$
- $\sin 2x = \sin(2x)$
- $\sin 2+x = \sin(2)+x$
- $\sin x^2 = \sin(x^2)$
- $2(x+3)x = 2*(x+3)*x$
- $-x^2 = -(x^2)$
- $2x = 2*x$
- $1/2x = 1/(2*x)$
- $e^2x = e^(2*x)$
- $x^2^3 = x^(2^3)$

Konstanten

Die Konstante rand

Ergibt eine zufällig reelle Zahl zwischen 0 und 1.

Syntax

rand

Beschreibung:

rand wird wie eine Konstante benutzt, allerdings ergibt diese bei jeder Berechnung eine pseudo-zufällige reelle Zahl aus dem Intervall [0;1].

Bemerkungen

Verwendet man rand in einer Funktion, dann wird bei jeder Neuberechnung der Funktion der zugehörige Funktionsgraph anders aussehen, da rand bei jeder neuen Berechnung einen anderen Wert beinhaltet. Die Graphen von Funktionen, die rand beinhalten, werden auch dann ihr Aussehen verändern, wenn Graph die Funktionsgraphen neu berechnen muss, z.B. beim Verschieben des Koordinatensystems oder bei Anpassung der Fenstergröße.

Implementierung

rand verwendet zur Erzeugung von reellen Pseudozufallszahlen zwischen 0 und 1 einen multiplikativen Kongruenzgenerator mit einer Periodenlänge von 2^{32} .

Siehe auch

[Wikipedia](http://en.wikipedia.org/wiki/Random_number_generator#Computational_methods) [http://en.wikipedia.org/wiki/Random_number_generator#Computational_methods]

[MathWorld](http://mathworld.wolfram.com/RandomNumber.html) [http://mathworld.wolfram.com/RandomNumber.html]

Trigonometrisch

sin-Funktion

Liefert den Sinus des Arguments.

Syntax

sin(z)

Beschreibung:

Die sin-Funktion berechnet den Sinus eines Winkels z mit der Einheit *Bogenmaß* oder Grad, abhängig von der aktuellen Einstellung. z kann ein beliebiger *numerischer Ausdruck* sein, der eine *reelle Zahl* oder eine *komplexe Zahl* darstellt. Ist z eine reelle Zahl, so liegt das Ergebnis zwischen -1 und +1.

Bemerkungen

Bei sehr großen Argumenten wird die Funktion zunehmend ungenauer.

Siehe auch

[Wikipedia](http://en.wikipedia.org/wiki/Trigonometric_functions#Sine) [http://en.wikipedia.org/wiki/Trigonometric_functions#Sine]

[MathWorld](http://mathworld.wolfram.com/Sine.html) [http://mathworld.wolfram.com/Sine.html]

cos-Funktion

Liefert den Cosinus des Arguments.

Syntax

cos(z)

Beschreibung:

Die cos-Funktion berechnet den Cosinus eines Winkels z mit der Einheit *Bogenmaß* oder Grad, abhängig von der aktuellen Einstellung. z kann ein beliebiger *numerischer Ausdruck* sein, der eine *reelle Zahl* oder eine *komplexe Zahl* darstellt. Ist z eine reelle Zahl, so liegt das Ergebnis zwischen -1 und +1.

Bemerkungen

Bei sehr großen Argumenten wird die Funktion zunehmend ungenauer.

Siehe auch

[Wikipedia](http://en.wikipedia.org/wiki/Trigonometric_functions#Cosine) [http://en.wikipedia.org/wiki/Trigonometric_functions#Cosine]

[MathWorld](http://mathworld.wolfram.com/Cosine.html) [http://mathworld.wolfram.com/Cosine.html]

tan-Funktion

Liefert den Tangens des Arguments.

Syntax

$\tan(z)$

Beschreibung:

Die \tan -Funktion berechnet den Tangens eines Winkels z mit der Einheit *Bogenmaß* oder Grad, abhängig von der aktuellen Einstellung. z kann ein beliebiger *numerischer Ausdruck* sein, der eine *reelle Zahl* oder eine *komplexe Zahl* darstellt.

Bemerkungen

Bei sehr großen Argumenten wird die Funktion zunehmend ungenauer. \tan ist bei $z = p \cdot \pi/2$ undefiniert, wobei p eine *Ganzzahl* ist, jedoch liefert die Function eine sehr große Zahl zurück, wenn z in der Nähe des undefinierten Wertes ist.

Siehe auch

[Wikipedia](http://en.wikipedia.org/wiki/Trigonometric_functions#Tangent) [http://en.wikipedia.org/wiki/Trigonometric_functions#Tangent]

[MathWorld](http://mathworld.wolfram.com/Tangent.html) [http://mathworld.wolfram.com/Tangent.html]

asin-Funktion

Liefert den Arcussinus des Arguments.

Syntax

$\text{asin}(z)$

Beschreibung:

Die asin -Funktion berechnet den Arcussinus von z . Das Ergebnis hat die Einheit *Bogenmaß* oder Grad, abhängig von der aktuellen Einstellung. z kann ein beliebiger *numerischer Ausdruck* sein, der eine *reelle Zahl* darstellt. Dies ist die Umkehrung der sin -Funktion.

Siehe auch

[Wikipedia](http://en.wikipedia.org/wiki/Inverse_trigonometric_functions) [http://en.wikipedia.org/wiki/Inverse_trigonometric_functions]

[MathWorld](http://mathworld.wolfram.com/InverseSine.html) [http://mathworld.wolfram.com/InverseSine.html]

acos-Funktion

Liefert den Arcuscosinus des Arguments.

Syntax

$\text{acos}(z)$

Beschreibung:

Die acos -Funktion berechnet den Arcuscosinus von z . Das Ergebnis hat die Einheit *Bogenmaß* oder Grad, abhängig von der aktuellen Einstellung. z kann ein beliebiger *numerischer Ausdruck* sein, der eine *reelle Zahl* darstellt. Dies ist die Umkehrung der cos -Funktion.

Siehe auch

[Wikipedia](http://en.wikipedia.org/wiki/Inverse_trigonometric_functions) [http://en.wikipedia.org/wiki/Inverse_trigonometric_functions]

[MathWorld](http://mathworld.wolfram.com/InverseCosine.html) [http://mathworld.wolfram.com/InverseCosine.html]

atan-Funktion

Liefert den Arcustangens des Arguments zurück.

Syntax $\operatorname{atan}(z)$ **Beschreibung:**

Die atan -Funktion berechnet den Arcustangens von z . Das Ergebnis hat die Einheit *Bogenmaß* oder Grad, abhängig von der aktuellen Einstellung. z kann ein beliebiger *numerischer Ausdruck* sein, der eine *reelle Zahl* darstellt. Dies ist die Umkehrung der tan -Funktion.

Siehe auch

[Wikipedia](http://en.wikipedia.org/wiki/Inverse_trigonometric_functions) [http://en.wikipedia.org/wiki/Inverse_trigonometric_functions]

[MathWorld](http://mathworld.wolfram.com/InverseTangent.html) [http://mathworld.wolfram.com/InverseTangent.html]

sec-Funktion

Liefert den Sekans des Arguments zurück.

Syntax $\operatorname{sec}(z)$ **Beschreibung:**

Die sec -Funktion berechnet den Secans von z . Das Ergebnis hat die Einheit *Bogenmaß* oder Grad, abhängig von der aktuellen Einstellung. $\operatorname{sec}(z)$ ist dasselbe wie $1/\cos(z)$. z kann ein beliebiger *numerischer Ausdruck* sein, der eine *reelle Zahl* oder eine *komplexe Zahl* darstellt.

Bemerkungen

Bei sehr großen Argumenten wird die Funktion zunehmend ungenauer.

Siehe auch

[Wikipedia](http://en.wikipedia.org/wiki/Trigonometric_functions#Reciprocal_functions) [http://en.wikipedia.org/wiki/Trigonometric_functions#Reciprocal_functions]

[MathWorld](http://mathworld.wolfram.com/Secant.html) [http://mathworld.wolfram.com/Secant.html]

csc-Funktion

Liefert den Cosekans des Arguments zurück.

Syntax $\operatorname{csc}(z)$ **Beschreibung:**

Die csc -Funktion berechnet den Cosecans von z . Das Ergebnis hat die Einheit *Bogenmaß* oder Grad, abhängig von der aktuellen Einstellung. $\operatorname{csc}(z)$ ist dasselbe wie $1/\sin(z)$. z kann ein beliebiger *numerischer Ausdruck* sein, der eine *reelle Zahl* oder eine *komplexe Zahl* darstellt.

Bemerkungen

Bei sehr großen Argumenten wird die Funktion zunehmend ungenauer.

Siehe auch

[Wikipedia](http://en.wikipedia.org/wiki/Trigonometric_functions#Reciprocal_functions) [http://en.wikipedia.org/wiki/Trigonometric_functions#Reciprocal_functions]

[MathWorld](http://mathworld.wolfram.com/Cosecant.html) [http://mathworld.wolfram.com/Cosecant.html]

cot-Funktion

Liefert den Cotangens des Arguments zurück.

Syntax $\operatorname{cot}(z)$ **Beschreibung:**

Die cot -Funktion berechnet den Cotangens von z . Das Ergebnis hat die Einheit *Bogenmaß* oder Grad, abhängig von der aktuellen Einstellung. $\operatorname{cot}(z)$ ist dasselbe wie $1/\tan(z)$. z kann ein beliebiger *numerischer Ausdruck* sein, der eine *reelle Zahl* oder eine *komplexe Zahl* darstellt.

Bemerkungen

Bei sehr großen Argumenten wird die Funktion zunehmend ungenauer.

Siehe auch

[Wikipedia](http://en.wikipedia.org/wiki/Trigonometric_functions#Reciprocal_functions) [http://en.wikipedia.org/wiki/Trigonometric_functions#Reciprocal_functions]
[MathWorld](http://mathworld.wolfram.com/Cotangent.html) [http://mathworld.wolfram.com/Cotangent.html]

asec-Funktion

Liefert den Arcusekans des Arguments zurück.

Syntax

$\text{asec}(z)$

Beschreibung:

Die asec -Funktion berechnet den Arcusekans von z . Das Ergebnis hat die Einheit *Bogenmaß* oder Grad, abhängig von der aktuellen Einstellung. $\text{asec}(z)$ ist dasselbe wie $\text{acos}(1/z)$. z kann ein beliebiger *numerischer Ausdruck* sein, der eine *reelle Zahl* darstellt. Dies ist die Umkehrung der sec -Funktion.

Siehe auch

[Wikipedia](http://en.wikipedia.org/wiki/Inverse_trigonometric_functions) [http://en.wikipedia.org/wiki/Inverse_trigonometric_functions]
[MathWorld](http://mathworld.wolfram.com/InverseSecant.html) [http://mathworld.wolfram.com/InverseSecant.html]

acsc-Funktion

Liefert den Arcuscosekans des Arguments zurück.

Syntax

$\text{acsc}(z)$

Beschreibung:

Die acsc -Funktion berechnet den Arcuscosekans von z . Das Ergebnis hat die Einheit *Bogenmaß* oder Grad, abhängig von der aktuellen Einstellung. $\text{acsc}(z)$ ist dasselbe wie $\text{asin}(1/z)$. z kann ein beliebiger *numerischer Ausdruck* sein, der eine *reelle Zahl* darstellt. Dies ist die Umkehrung der csc -Funktion.

Siehe auch

[Wikipedia](http://en.wikipedia.org/wiki/Inverse_trigonometric_functions) [http://en.wikipedia.org/wiki/Inverse_trigonometric_functions]
[MathWorld](http://mathworld.wolfram.com/InverseCosecant.html) [http://mathworld.wolfram.com/InverseCosecant.html]

acot-Funktion

Liefert den Arcuscotangens des Arguments zurück.

Syntax

$\text{acot}(z)$

Beschreibung:

Die acot -Funktion berechnet den Arcuscotangens von z . Das Ergebnis hat die Einheit *Bogenmaß* oder Grad, abhängig von der aktuellen Einstellung. $\text{acot}(z)$ ist dasselbe wie $\text{atan}(1/z)$. z kann ein beliebiger *numerischer Ausdruck* sein, der eine *reelle Zahl* darstellt. Dies ist die Umkehrung der cot -Funktion.

Bemerkungen

Die acot -Funktion liefert einen Wert im Intervall $]-\pi/2;\pi/2[$ ($]-90;90[$) wenn in Grad gerechnet wird, das ist die allgemeine Definition, obwohl der Bereich manchmal auch mit $]0;\pi[$ definiert wird.

Siehe auch

[Wikipedia](http://en.wikipedia.org/wiki/Inverse_trigonometric_functions) [http://en.wikipedia.org/wiki/Inverse_trigonometric_functions]
[MathWorld](http://mathworld.wolfram.com/InverseCotangent.html) [http://mathworld.wolfram.com/InverseCotangent.html]

Hyperbolicus

sinh-Funktion

Liefert den Sinus Hyperbolicus des Arguments.

Syntax $\sinh(z)$ **Beschreibung:**

Die \sinh -Funktion berechnet den Sinus Hyperbolicus von z . z kann ein beliebiger *numerischer Ausdruck* sein, der eine *reelle Zahl* oder eine *komplexe Zahl* darstellt.

Sinus Hyperbolicus ist als $\sinh(z) = \frac{1}{2}(e^z - e^{-z})$ definiert.

Siehe auch

[Wikipedia](http://en.wikipedia.org/wiki/Hyperbolic_function) [http://en.wikipedia.org/wiki/Hyperbolic_function]

[MathWorld](http://mathworld.wolfram.com/HyperbolicSine.html) [http://mathworld.wolfram.com/HyperbolicSine.html]

cosh-Funktion

Liefert den Cosinus Hyperbolicus des Arguments.

Syntax $\cosh(z)$ **Beschreibung:**

Die \cosh -Funktion berechnet den Cosinus Hyperbolicus von z . z kann ein beliebiger *numerischer Ausdruck* sein, der eine *reelle Zahl* oder eine *komplexe Zahl* darstellt.

Cosinus Hyperbolicus ist als $\cosh(z) = \frac{1}{2}(e^z + e^{-z})$ definiert.

Siehe auch

[Wikipedia](http://en.wikipedia.org/wiki/Hyperbolic_function) [http://en.wikipedia.org/wiki/Hyperbolic_function]

[MathWorld](http://mathworld.wolfram.com/HyperbolicCosine.html) [http://mathworld.wolfram.com/HyperbolicCosine.html]

tanh-Funktion

Liefert den Tangens Hyperbolicus des Arguments.

Syntax $\tanh(z)$ **Beschreibung:**

Die \tanh -Funktion berechnet den Tangens Hyperbolicus von z . z kann ein beliebiger *numerischer Ausdruck* sein, der eine *reelle Zahl* oder eine *komplexe Zahl* darstellt.

Tangens Hyperbolicus ist als $\tanh(z) = \sinh(z)/\cosh(z)$ definiert.

Siehe auch

[Wikipedia](http://en.wikipedia.org/wiki/Hyperbolic_function) [http://en.wikipedia.org/wiki/Hyperbolic_function]

[MathWorld](http://mathworld.wolfram.com/HyperbolicTangent.html) [http://mathworld.wolfram.com/HyperbolicTangent.html]

asinh-Funktion

Liefert den Arkussinus Hyperbolicus des Arguments.

Syntax $\operatorname{asinh}(z)$ **Beschreibung:**

Die asinh -Funktion berechnet den Arcussinus Hyperbolicus von z . z kann ein beliebiger *numerischer Ausdruck* sein, der eine *reelle Zahl* oder eine *komplexe Zahl* darstellt. asinh stellt die Umkehrfunktion von \sinh dar, also $\operatorname{asinh}(\sinh(z)) = z$.

Siehe auch

[Wikipedia](http://en.wikipedia.org/wiki/Hyperbolic_function) [http://en.wikipedia.org/wiki/Hyperbolic_function]

[MathWorld](http://mathworld.wolfram.com/InverseHyperbolicSine.html) [http://mathworld.wolfram.com/InverseHyperbolicSine.html]

acosh-Funktion

Liefert den Arkuscosinus Hyperbolicus des Arguments.

Syntax

`acosh(z)`

Beschreibung:

Die `acosh`-Funktion berechnet den Arkuscosinus Hyperbolicus von z . z kann ein beliebiger *numerischer Ausdruck* sein, der eine *reelle Zahl* oder eine *komplexe Zahl* darstellt. `acosh` stellt die Umkehrfunktion von `cosh` dar, also $\text{acosh}(\cosh(z)) = z$.

Siehe auch

[Wikipedia](http://en.wikipedia.org/wiki/Hyperbolic_function) [http://en.wikipedia.org/wiki/Hyperbolic_function]

[MathWorld](http://mathworld.wolfram.com/InverseHyperbolicCosine.html) [http://mathworld.wolfram.com/InverseHyperbolicCosine.html]

atanh-Funktion

Liefert den Arkustangens Hyperbolicus des Arguments.

Syntax

`atanh(z)`

Beschreibung:

Die `atanh`-Funktion berechnet den Areatangens Hyperbolicus von z . z kann ein beliebiger *numerischer Ausdruck* sein, der eine *reelle Zahl* oder eine *komplexe Zahl* darstellt. `atanh` stellt die Umkehrfunktion zu `tanh` dar, also $\text{atanh}(\tanh(z)) = z$.

Siehe auch

[Wikipedia](http://en.wikipedia.org/wiki/Hyperbolic_function) [http://en.wikipedia.org/wiki/Hyperbolic_function]

[MathWorld](http://mathworld.wolfram.com/InverseHyperbolicTangent.html) [http://mathworld.wolfram.com/InverseHyperbolicTangent.html]

csch-Funktion

Liefert den Cosecans Hyperbolicus des Arguments.

Syntax

`csch(z)`

Beschreibung:

Die `csch`-Funktion berechnet den Cosecans Hyperbolicus von z . z kann ein beliebiger *numerischer Ausdruck* sein, der eine *reelle Zahl* oder eine *komplexe Zahl* darstellt.

Cosecans Hyperbolicus ist als $\text{csch}(z) = 1/\sinh(z) = 2/(e^z - e^{-z})$ definiert.

Siehe auch

[Wikipedia](http://en.wikipedia.org/wiki/Hyperbolic_function) [http://en.wikipedia.org/wiki/Hyperbolic_function]

[MathWorld](http://mathworld.wolfram.com/HyperbolicCosecant.html) [http://mathworld.wolfram.com/HyperbolicCosecant.html]

sech-Funktion

Liefert den Secanshyperbolicus des Arguments.

Syntax

`sech(z)`

Beschreibung:

Die `sech`-Funktion berechnet den Secans Hyperbolicus von z . z kann ein beliebiger *numerischer Ausdruck* sein, der eine *reelle Zahl* oder eine *komplexe Zahl* darstellt.

Hyperbolic Secans ist als $\text{sech}(z) = 1/\cosh(z) = 2/(e^z + e^{-z})$ definiert.

Siehe auch

[Wikipedia](http://en.wikipedia.org/wiki/Hyperbolic_function) [http://en.wikipedia.org/wiki/Hyperbolic_function]

[MathWorld](http://mathworld.wolfram.com/HyperbolicSecant.html) [http://mathworld.wolfram.com/HyperbolicSecant.html]

coth-Funktion

Liefert den Cotangens Hyperbolicus des Arguments.

Syntax

$\operatorname{coth}(z)$

Beschreibung:

Die coth -Funktion berechnet den Cotangens Hyperbolicus von z . z kann ein beliebiger *numerischer Ausdruck* sein, der eine *reelle Zahl* oder eine *komplexe Zahl* darstellt.

Hyperbolic Cotangens ist als $\operatorname{coth}(z) = 1/\tanh(z) = \cosh(z)/\sinh(z) = (e^z + e^{-z})/(e^z - e^{-z})$ definiert.

Siehe auch

[Wikipedia](http://en.wikipedia.org/wiki/Hyperbolic_function) [http://en.wikipedia.org/wiki/Hyperbolic_function]

[MathWorld](http://mathworld.wolfram.com/HyperbolicCotangent.html) [http://mathworld.wolfram.com/HyperbolicCotangent.html]

acsch-Funktion

Liefert den Arkuscosecans Hyperbolicus des Arguments.

Syntax

$\operatorname{acsch}(z)$

Beschreibung:

Die acsch -Funktion berechnet den Areacosecans Hyperbolicus von z . z kann ein beliebiger *numerischer Ausdruck* sein, der eine *reelle Zahl* oder eine *komplexe Zahl* darstellt. acsch stellt die Umkehrfunktion zu csch dar, also $\operatorname{acsch}(\operatorname{csch}(z)) = z$.

Siehe auch

[Wikipedia](http://en.wikipedia.org/wiki/Hyperbolic_function) [http://en.wikipedia.org/wiki/Hyperbolic_function]

[MathWorld](http://mathworld.wolfram.com/InverseHyperbolicCosecant.html) [http://mathworld.wolfram.com/InverseHyperbolicCosecant.html]

asech-Funktion

Liefert den Arkusecans Hyperbolicus des Arguments.

Syntax

$\operatorname{asech}(z)$

Beschreibung:

Die asech -Funktion berechnet den Areasecans Hyperbolicus von z . z kann ein beliebiger *numerischer Ausdruck* sein, der eine *reelle Zahl* oder eine *komplexe Zahl* darstellt. asech stellt die Umkehrfunktion zu sech dar, also $\operatorname{asech}(\operatorname{sech}(z)) = z$.

Siehe auch

[Wikipedia](http://en.wikipedia.org/wiki/Hyperbolic_function) [http://en.wikipedia.org/wiki/Hyperbolic_function]

[MathWorld](http://mathworld.wolfram.com/InverseHyperbolicSecant.html) [http://mathworld.wolfram.com/InverseHyperbolicSecant.html]

acoth-Funktion

Liefert den Arkuscotangens Hyperbolicus des Arguments.

Syntax

$\operatorname{acoth}(z)$

Beschreibung:

Die acoth -Funktion berechnet den Area Cotangens Hyperbolicus von z . z kann ein beliebiger *numerischer Ausdruck* sein, der eine *reelle Zahl* oder eine *komplexe Zahl* darstellt. acoth stellt die Umkehrfunktion von coth dar, also $\operatorname{acoth}(\operatorname{coth}(z)) = z$. Für reelle Zahlen ist acoth im Intervall $[-1;1]$ nicht definiert.

Siehe auch

[Wikipedia](http://en.wikipedia.org/wiki/Hyperbolic_function) [http://en.wikipedia.org/wiki/Hyperbolic_function]

[MathWorld](http://mathworld.wolfram.com/InverseHyperbolicCotangent.html) [http://mathworld.wolfram.com/InverseHyperbolicCotangent.html]

Potenz und Logarithmus

sqr-Funktion

Liefert das Quadrat des Arguments.

Syntax

sqr(z)

Beschreibung:

Die `sqr`-Funktion berechnet das Quadrat von z , also z hoch 2. z kann ein beliebiger *numerischer Ausdruck* sein, der eine *reelle Zahl* oder eine *komplexe Zahl* darstellt.

exp-Funktion

Liefert e potenziert mit dem Argument.

Syntax

exp(z)

Beschreibung:

Die `exp`-Funktion wird e , die Eulersche Zahl, mit z potenziert. Dies ist dasselbe wie e^z . z kann ein beliebiger *numerischer Ausdruck* sein, der eine *reelle Zahl* oder eine *komplexe Zahl* darstellt.

Siehe auch

[Wikipedia](http://en.wikipedia.org/wiki/Exponential_function) [http://en.wikipedia.org/wiki/Exponential_function]

[MathWorld](http://mathworld.wolfram.com/ExponentialFunction.html) [http://mathworld.wolfram.com/ExponentialFunction.html]

sqrt-Funktion

Liefert die Quadratwurzel des Arguments.

Syntax

sqrt(z)

Beschreibung:

Die `sqrt`-Funktion berechnet die Quadratwurzel von z , also z hoch $\frac{1}{2}$. z kann ein beliebiger *numerischer Ausdruck* sein, der eine *reelle Zahl* oder eine *komplexe Zahl* darstellt. Wird die Berechnung mit reellen Zahlen durchgeführt, so ist das Argument nur für $z \geq 0$ definiert.

Siehe auch

[Wikipedia](http://en.wikipedia.org/wiki/Square_root) [http://en.wikipedia.org/wiki/Square_root]

[MathWorld](http://mathworld.wolfram.com/SquareRoot.html) [http://mathworld.wolfram.com/SquareRoot.html]

root-Funktion

Liefert die n -te Wurzel des Arguments.

Syntax

root(n, z)

Beschreibung:

Die `root`-Funktion berechnet die n . Wurzel von z . n und z können beliebige *numerischer Ausdruck* sein, die *reelle Zahl* oder *komplexe Zahl* darstellen. Wird die Berechnung mit reellen Zahlen durchgeführt, so ist das Argument nur für $z \geq 0$ definiert.

Bemerkungen

Wird die Berechnung mit reellen Zahlen durchgeführt, so ist die Funktion nur für $z < 0$ definiert, wenn n eine ungerade *Ganzzahl* ist. Bei Berechnungen mit komplexen Zahlen ist `root` für die ganze komplexe

Ebene definiert (ohne den Pol $n=0$). Beachten Sie, daß bei Berechnungen mit komplexen Zahlen das Ergebnis immer einen imaginären Anteil hat, wenn $z < 0$ ist; obgleich das Ergebnis real ist, wenn mit reellen Zahlen und n als ungerader Ganzzahl gerechnet wird.

Beispiel

Anstatt $x^{(1/3)}$ können Sie $\text{root}(3, x)$ verwenden.

Siehe auch

[Wikipedia](http://en.wikipedia.org/wiki/Nth_root) [http://en.wikipedia.org/wiki/Nth_root]

[MathWorld](http://mathworld.wolfram.com/RadicalRoot.html) [http://mathworld.wolfram.com/RadicalRoot.html]

In-Funktion

Liefert den natürlichen Logarithmus des Arguments.

Syntax

$\ln(z)$

Beschreibung:

Die \ln -Funktion berechnet die Logarithmus von z mit der Basis e , der Eulerschen Zahl. $\ln(z)$ ist allgemein als der natürlichen Logarithmus bekannt. z kann ein beliebiger *numerischer Ausdruck* sein, der eine *reelle Zahl* oder eine *komplexe Zahl* darstellt. Wird die Berechnung mit reellen Zahlen durchgeführt, so ist das Argument nur für $z > 0$ definiert. Bei Berechnungen mit komplexen Zahlen ist z für alle Zahlen außer $z=0$ definiert.

Siehe auch

[Wikipedia](http://en.wikipedia.org/wiki/Natural_logarithm) [http://en.wikipedia.org/wiki/Natural_logarithm]

[MathWorld](http://mathworld.wolfram.com/NaturalLogarithm.html) [http://mathworld.wolfram.com/NaturalLogarithm.html]

log-Funktion

Liefert den 10er- Logarithmus des Arguments.

Syntax

$\log(z)$

Beschreibung:

Die \log -Funktion berechnet den Logarithmus von z mit der Basis 10. z kann ein beliebiger *numerischer Ausdruck* sein, der eine *reelle Zahl* oder eine *komplexe Zahl* darstellt. Wird die Berechnung mit reellen Zahlen durchgeführt, so ist das Argument nur für $z > 0$ definiert. Bei Berechnungen mit komplexen Zahlen ist z für alle Zahlen außer $z=0$ definiert.

Siehe auch

[Wikipedia](http://en.wikipedia.org/wiki/Common_logarithm) [http://en.wikipedia.org/wiki/Common_logarithm]

[MathWorld](http://mathworld.wolfram.com/CommonLogarithm.html) [http://mathworld.wolfram.com/CommonLogarithm.html]

logb-Funktion

Liefert den Logarithmus mit der Basis n des Arguments.

Syntax

$\log_b(z, n)$

Beschreibung:

Die \log_b -Funktion berechnet den Logarithmus von z mit der Basis n . z kann ein beliebiger *numerischer Ausdruck* sein, der eine *reelle Zahl* oder eine *komplexe Zahl* darstellt. Wird die Berechnung mit reellen Zahlen durchgeführt, so ist das Argument nur für $z > 0$ definiert. Bei Berechnungen mit komplexen Zahlen ist z für alle Zahlen außer $z=0$ definiert. n muß eine positive reelle Zahl sein.

Siehe auch

[Wikipedia](http://en.wikipedia.org/wiki/Logarithm) [http://en.wikipedia.org/wiki/Logarithm]

[MathWorld](http://mathworld.wolfram.com/Logarithm.html) [http://mathworld.wolfram.com/Logarithm.html]

Komplex

abs-Funktion

Liefert den Absolutwert des Arguments zurück.

Syntax

$\text{abs}(z)$

Beschreibung:

Die Funktion `abs` ergibt den absoluten Wert, auch Betrag genannt, einer Zahl z . Üblicherweise schreibt man dies als $|z|$. Bei z kann es sich um einen *numerischer Ausdruck* handeln, der eine *reelle Zahl* oder eine *komplexe Zahl* ergibt. Das Ergebnis von `abs(z)` ist in jedem Fall ein reeller Wert.

Siehe auch

[Wikipedia](http://en.wikipedia.org/wiki/Absolute_value) [http://en.wikipedia.org/wiki/Absolute_value]

[MathWorld](http://mathworld.wolfram.com/AbsoluteValue.html) [http://mathworld.wolfram.com/AbsoluteValue.html]

arg-Funktion

Ergibt das Argument des Parameters.

Syntax

$\text{arg}(z)$

Beschreibung:

Die `arg`-Funktion liefert das Argument oder den Winkel von z . z kann ein beliebiger *numerischer Ausdruck* sein, der eine

Siehe auch

[Wikipedia](http://en.wikipedia.org/wiki/Arg_(mathematics)) [http://en.wikipedia.org/wiki/Arg_(mathematics)]

[MathWorld](http://mathworld.wolfram.com/ComplexArgument.html) [http://mathworld.wolfram.com/ComplexArgument.html]

conj-Funktion

Ergibt die konjugiert komplexe Zahl des Arguments.

Syntax

$\text{conj}(z)$

Beschreibung:

Die `conj`-Funktion liefert die konjugierte Zahl von z . z kann ein beliebiger *numerischer Ausdruck* sein, der eine *reelle Zahl* oder eine *komplexe Zahl* darstellt. Die Funktion ist folgendermaßen definiert: $\text{conj}(z) = \text{re}(z) - i \cdot \text{im}(z)$.

Siehe auch

[Wikipedia](http://en.wikipedia.org/wiki/Complex_conjugation) [http://en.wikipedia.org/wiki/Complex_conjugation]

[MathWorld](http://mathworld.wolfram.com/ComplexConjugate.html) [http://mathworld.wolfram.com/ComplexConjugate.html]

re-Funktion

Liefert den Realteil des Arguments zurück.

Syntax

$\text{re}(z)$

Beschreibung:

Die `re`-Funktion liefert den Realteil von z . z kann ein beliebiger *numerischer Ausdruck* sein, der eine *reelle Zahl* oder eine *komplexe Zahl* darstellt.

Siehe auch

[Wikipedia](http://en.wikipedia.org/wiki/Real_part) [http://en.wikipedia.org/wiki/Real_part]

[MathWorld](http://mathworld.wolfram.com/RealPart.html) [http://mathworld.wolfram.com/RealPart.html]

im-Funktion

Liefert den Imaginärteil des Arguments zurück.

Syntax

$\text{im}(z)$

Beschreibung:

Die `im`-Funktion liefert den Imaginärteil von z . z kann ein beliebiger *numerischer Ausdruck* sein, der eine *reelle Zahl* oder eine *komplexe Zahl* darstellt.

Siehe auch

[Wikipedia](http://en.wikipedia.org/wiki/Imaginary_part) [http://en.wikipedia.org/wiki/Imaginary_part]

[MathWorld](http://mathworld.wolfram.com/ImaginaryPart.html) [http://mathworld.wolfram.com/ImaginaryPart.html]

Runden

trunc-Funktion

Entfernt den gebrochenen Anteil des Arguments.

Syntax

$\text{trunc}(z)$

Beschreibung:

Die `trunc`-Funktion liefert den *Ganzzahl* Anteil von z zurück. Die Funktion entfernt den dezimalen Anteil von z , d.h. rundet in Richtung Null. z kann ein beliebiger *numerischer Ausdruck* sein, der eine *reelle Zahl* oder eine *komplexe Zahl* darstellt. Ist z eine komplexe Zahl, liefert die Funktion $\text{trunc}(\text{re}(z))+\text{trunc}(\text{im}(z))\mathbf{i}$ zurück.

Siehe auch

[Wikipedia](http://en.wikipedia.org/wiki/Truncate) [http://en.wikipedia.org/wiki/Truncate]

[MathWorld](http://mathworld.wolfram.com/Truncate.html) [http://mathworld.wolfram.com/Truncate.html]

fract-Funktion

Liefert den gebrochenen Anteil des Argumentes.

Syntax

$\text{fract}(z)$

Beschreibung:

Die `fract`-Funktion liefert den gebrochenen Anteil von z zurück. Die Funktion entfernt den *Ganzzahl* Anteil von z , also $\text{fract}(z) = z - \text{trunc}(z)$. z kann ein beliebiger *numerischer Ausdruck* sein, der eine *reelle Zahl* oder eine *komplexe Zahl* darstellt. Wenn z eine komplexe Zahl ist, liefert die Funktion $\text{fract}(\text{re}(z))+\text{fract}(\text{im}(z))\mathbf{i}$.

Siehe auch

[Wikipedia](http://en.wikipedia.org/wiki/Floor_and_ceiling_functions#Fractional_part) [http://en.wikipedia.org/wiki/Floor_and_ceiling_functions#Fractional_part]

[MathWorld](http://mathworld.wolfram.com/FractionalPart.html) [http://mathworld.wolfram.com/FractionalPart.html]

ceil-Funktion

Rundet das Argument auf.

Syntax

$\text{ceil}(z)$

Beschreibung:

Die `ceil`-Funktion findet die kleinste *Ganzzahl*, die nicht kleiner als z ist. z kann ein beliebiger *numerischer Ausdruck* sein, der eine *reelle Zahl* oder eine *komplexe Zahl* darstellt. Wenn z eine komplexe Zahl ist, liefert die Funktion $\text{ceil}(\text{re}(z))+\text{ceil}(\text{im}(z))\mathbf{i}$ zurück.

Siehe auch

[Wikipedia](http://en.wikipedia.org/wiki/Floor_and_ceiling_functions) [http://en.wikipedia.org/wiki/Floor_and_ceiling_functions]

[MathWorld](http://mathworld.wolfram.com/CeilingFunction.html) [http://mathworld.wolfram.com/CeilingFunction.html]

floor-Funktion

Rundet das Argument ab.

Syntax

`floor(z)`

Beschreibung:

Die `floor`-Funktion, auch Abrundungsfunktion genannt, liefert die größte *Ganzzahl*, die nicht größer als z ist. z kann ein beliebiger *numerischer Ausdruck* sein, der eine *reelle Zahl* oder eine *komplexe Zahl* darstellt. Wenn z eine komplexe Zahl ist, liefert die Funktion `floor(re(z))+floor(im(z))i` zurück.

Siehe auch

[Wikipedia](http://en.wikipedia.org/wiki/Floor_and_ceiling_functions) [http://en.wikipedia.org/wiki/Floor_and_ceiling_functions]

[MathWorld](http://mathworld.wolfram.com/FloorFunction.html) [http://mathworld.wolfram.com/FloorFunction.html]

round-Funktion

Rundet eine Zahl auf die angegebene Anzahl Dezimalstellen.

Syntax

`round(z,n)`

Beschreibung:

Die `round`-Funktion rundet z auf die Anzahl Dezimalstellen in n . z kann ein beliebiger *numerischer Ausdruck* sein, der eine *reelle Zahl* oder eine *komplexe Zahl* darstellt. Wenn z eine komplexe Zahl ist, liefert die Funktion `round(re(z),n)+round(im(z),n)i`. n kann ein beliebiger numerischer Ausdruck sein, der eine *Ganzzahl* darstellt. Wenn $n < 0$ ist, wird z auf n Vorkommastellen gerundet.

Beispiele

`round(412.4572,3) = 412.457`

`round(412.4572,2) = 412.46`

`round(412.4572,1) = 412.5`

`round(412.4572,0) = 412`

`round(412.4572,-2) = 400`

Siehe auch

[Wikipedia](http://en.wikipedia.org/wiki/Rounding) [http://en.wikipedia.org/wiki/Rounding]

[MathWorld](http://mathworld.wolfram.com/NearestIntegerFunction.html) [http://mathworld.wolfram.com/NearestIntegerFunction.html]

Stückweise

sign-Funktion

Liefert das Vorzeichen des Arguments.

Syntax

`sign(z)`

Beschreibung:

Die `sign`-Funktion, auch Signum genannt, liefert das Vorzeichen von z . z kann ein beliebiger *numerischer Ausdruck* sein, der eine *reelle Zahl* oder eine *komplexe Zahl* darstellt. Ist z eine reelle Zahl, liefert `sign(z)` liefert 1 bei $z > 0$ und -1 bei $z < 0$. `sign(z)` liefert 0 bei $z = 0$. `sign(z)` liefert $z/|z|$, wenn z eine komplexe Zahl ergibt.

Siehe auch

[Wikipedia](http://en.wikipedia.org/wiki/Sign_function) [http://en.wikipedia.org/wiki/Sign_function]

[MathWorld](http://mathworld.wolfram.com/Sign.html) [http://mathworld.wolfram.com/Sign.html]

u-Funktion

Die Einheits-Sprungfunktion.

Syntax

$u(z)$

Beschreibung:

$u(z)$ ist allgemein unter dem Namen Einheits-Sprungfunktion bekannt. z kann ein beliebiger *numerischer Ausdruck* sein, der eine *reelle Zahl* darstellt. Die Funktion ist nicht definiert, wenn z einen imaginären Anteil hat. $u(z)$ liefert 1 für $z \geq 0$ und 0 für $z < 0$.

Siehe auch

[Wikipedia](http://en.wikipedia.org/wiki/Unit_step#Discrete_form) [http://en.wikipedia.org/wiki/Unit_step#Discrete_form]

[MathWorld](http://mathworld.wolfram.com/HeavisideStepFunction.html) [http://mathworld.wolfram.com/HeavisideStepFunction.html]

min-Funktion

Findet und liefert den Minimalwert der Argumente.

Syntax

$\min(A, B, \dots)$

Beschreibung:

Die \min -Funktion liefert den kleinsten Wert ihrer Argumente. \min muß mindestens 2 Argumente haben. Die Argumente können beliebige *numerische Ausdrücke* sein, die *reelle Zahlen* oder *Komplexe Zahlen* ergeben. Wenn die Argumente komplexe Zahlen sind, liefert die Funktion $\min(\operatorname{re}(A), \operatorname{re}(B), \dots) + \min(\operatorname{im}(A), \operatorname{im}(B), \dots)i$ zurück.

max-Funktion

Findet und liefert den Maximalwert der Argumente.

Syntax

$\max(A, B, \dots)$

Beschreibung:

Die \max -Funktion liefert den größten Wert ihrer Argumente. \max muß mindestens 2 Argumente haben. Die Argumente können beliebige *numerische Ausdrücke* sein, die *reelle Zahlen* oder *Komplexe Zahlen* ergeben. Wenn die Argumente komplexe Zahlen sind, liefert die Funktion $\max(\operatorname{re}(A), \operatorname{re}(B), \dots) + \max(\operatorname{im}(A), \operatorname{im}(B), \dots)i$ zurück.

range-Funktion

Liefert das zweite Argument, wenn es im Bereich zwischen dem ersten und dem dritten Argument ist.

Syntax

$\operatorname{range}(A, z, B)$

Beschreibung:

Die range -Funktion liefert z zurück, wenn z größer als A und kleiner als B ist. Ist $z < A$, wird A zurückgeliefert. Ist $z > B$, wird B zurückgeliefert. Die Argumente können beliebige *numerische Ausdrücke* sein, die *reelle Zahlen* oder *Komplexe Zahlen* ergeben. Die Funktion hat dieselbe Wirkung wie $\max(A, \min(z, B))$.

if-Funktion

Wertet eine oder mehrere Bedingungen aus und liefert ein davon abhängiges Ergebnis zurück.

Syntax

$\operatorname{if}(\operatorname{cond1}, f1, \operatorname{cond2}, f2, \dots, \operatorname{condn}, fn [, fz])$

Beschreibung:

Die `if`-Funktion berechnet `cond1`, und wenn das Ergebnis nicht Null ist, wird `f1` berechnet und zurückgegeben. Andernfalls wird `cond2` berechnet, und wenn dieses Ergebnis nicht Null ist, wird `f2` zurückgeliefert usw. Wenn keine der Bedingungen wahr ist, wird `fz` zurückgeliefert. `fz` ist optional; wenn nicht angegeben, liefert `if` einen Fehler, wenn keine der Bedingungen wahr ist. Die Argumente können beliebige *numerische Ausdrücke* sein, die *reelle Zahlen* oder *Komplexe Zahlen* ergeben.

Speziell

integrate-Funktion

Berechnet eine Näherung für das numerische Integral des gegebenen Ausdrucks im angegebenen Intervall.

Syntax

`integrate(f,var,a,b)`

Beschreibung:

Die Funktion `integrate` liefert eine Näherung für das numerische Integral von f mit der Variablen `var` von a bis b , mathematisch beschrieben mit:

$$\int_a^b f(x) \, dx$$

Dieses Integral entspricht der Fläche zwischen der Funktion f und der X-Achse von a bis b , wobei hier die Fläche unterhalb der Achse negativ gezählt wird. f kann jede beliebige Funktionen mit `var` als zweitem Argument sein. a und b können beliebige *numerische Ausdrücke* sein, die *reelle Zahlen* entsprechen; sie können auch `-INF` oder `INF` sein (negativ bzw. positiv unendlich). `integrate` berechnet das Integral nicht exakt, sondern approximiert es mit Hilfe der Gauß-Kronrod-Quadratur, wobei der relative Schätzfehler geringer als 10^{-3} ist.

Beispiele

`f(x)=integrate(t^2-7t+1, t, -3, 15)` integriert die Funktion $f(t)=t^2-7t+1$ zwischen -3 und 15 und ergibt 396 . Hilfreicher ist das Folgende: `f(x)=integrate(s*sin(s), s, 0, x)`. Dies zeichnet das Integral der Funktion $f(s)=s*\sin(s)$ zwischen 0 und x , was dem bestimmten Integral von $f(x)=x*\sin(x)$ entspricht.

Siehe auch

[Wikipedia](http://en.wikipedia.org/wiki/Integral) [http://en.wikipedia.org/wiki/Integral]

[MathWorld](http://mathworld.wolfram.com/Integral.html) [http://mathworld.wolfram.com/Integral.html]

sum-Funktion

Liefert die Summe aller Ergebniswerte eines Ausdrucks für einen bestimmten Bereich ganzer Zahlen.

Syntax

`sum(f,var,a,b)`

Beschreibung:

Die Funktion `sum` summiert f , wobei `var` ganzzahlig von a bis b gezählt wird. Formal wird dies beschrieben durch:

$$\sum_{x=a}^b f(x)$$

f können beliebige Funktionen sein: mit `var` als Variable, wenn eine Standardfunktion vorliegt. a und b können beliebige *numerische Ausdrücke* sein, die *Ganzzahl* entsprechen.

Siehe auch

[Wikipedia](http://en.wikipedia.org/wiki/Summation) [http://en.wikipedia.org/wiki/Summation]

[MathWorld](http://mathworld.wolfram.com/Sum.html) [http://mathworld.wolfram.com/Sum.html]

product-Funktion

Liefert das Produkt aller Ergebniswerte eines Ausdrucks für einen bestimmten Bereich ganzer Zahlen.

Syntax

product(f,var,a,b)

Beschreibung:

Die `product`-Funktion liefert das Produkt von f , wobei var ganzzahlig von a bis b gezählt wird. Formal wird dies beschrieben durch:

$$\prod_{x=a}^b f(x)$$

f können beliebige Funktionen sein: mit var als Variable, wenn eine Standardfunktion vorliegt. a und b können beliebige *numerische Ausdrücke* sein, die *Ganzzahl* entsprechen.

Siehe auch

[Wikipedia](http://en.wikipedia.org/wiki/Multiplication#Capital_pi_notation) [http://en.wikipedia.org/wiki/Multiplication#Capital_pi_notation]

[MathWorld](http://mathworld.wolfram.com/Product.html) [http://mathworld.wolfram.com/Product.html]

fact-Funktion

Liefert die Fakultät des Arguments zurück.

Syntax

fact(n)

Beschreibung:

Die `fact`-Funktion liefert die Fakultät von n zurück, allgemein beschrieben mit $n!$. n kann ein beliebiger *numerischer Ausdruck* sein, der eine positive *Ganzzahl* darstellt. Die Funktion ist als $\text{fact}(n)=n(n-1)(n-2)\dots 1$ definiert, und die Beziehung zur `gamma`-Funktion ist $\text{fact}(n)=\text{gamma}(n+1)$.

Siehe auch

[Wikipedia](http://en.wikipedia.org/wiki/Factorial) [http://en.wikipedia.org/wiki/Factorial]

[MathWorld](http://mathworld.wolfram.com/Factorial.html) [http://mathworld.wolfram.com/Factorial.html]

gamma-Funktion

Liefert den Wert der Euler'schen Gammafunktion des Arguments.

Syntax

gamma(z)

Beschreibung:

Die `gamma`-Funktion liefert das Ergebnis der Euler'schen Gammafunktion von z , allgemein beschrieben mit $\Gamma(z)$. z kann ein beliebiger *numerischer Ausdruck* sein, der eine *reelle Zahl* oder eine *komplexe Zahl* darstellt. Die Beziehung zwischen Gammafunktion und Fakultäts-Funktion ist $\text{fact}(n)=\text{gamma}(n+1)$. Die mathematische Definition der Gammafunktion lautet:

$$\Gamma(z) = \int_0^{\infty} t^{z-1} e^{-t} dt$$

. Sie lässt sich nicht präzise berechnen, daher benutzt Graph für die Annäherung das sogenannte Lanczos-Verfahren, um die `gamma`-Funktion zu berechnen.

Siehe auch

[Wikipedia](http://en.wikipedia.org/wiki/Gamma_function) [http://en.wikipedia.org/wiki/Gamma_function]

[MathWorld](http://mathworld.wolfram.com/GammaFunction.html) [http://mathworld.wolfram.com/GammaFunction.html]

beta-Funktion

Liefert den Wert der Euler'schen Beta Funktion zu den Argumenten.

Syntax

$\text{beta}(m, n)$

Beschreibung:

Die beta-Funktion liefert das Ergebnis der Euler'schen Betafunktion von m und n . m und n können beliebige *numerische Ausdrücke* sein, die *reelle Zahlen* oder *Komplexe Zahlen* darstellen. Die Beziehung zwischen der beta-Funktion und der gamma-Funktion ist $\text{beta}(m, n) = \text{gamma}(m) * \text{gamma}(n) / \text{gamma}(m+n)$.

Siehe auch

[Wikipedia](http://en.wikipedia.org/wiki/Beta_function) [http://en.wikipedia.org/wiki/Beta_function]

[MathWorld](http://mathworld.wolfram.com/BetaFunction.html) [http://mathworld.wolfram.com/BetaFunction.html]

W-Funktion

Liefert den Wert der Euler'schen Beta Funktion zum Argument.

Syntax

$W(z)$

Beschreibung:

Die w-Funktion liefert das Ergebnis der Lambert W-function (auch: Omega-Funktion) für z . z kann ein beliebiger *numerischer Ausdruck* sein, der eine *reelle Zahl* oder eine *komplexe Zahl* darstellt. Die Umkehrfunktion zu W ist $f(W)=W*e^W$.

Bemerkungen

Für reelle Werte von z - bei $z < -1/e$ - liefert die w-Function Werte mit einem Imaginärteil.

Siehe auch

[Wikipedia](http://en.wikipedia.org/wiki/Lambert_w_function) [http://en.wikipedia.org/wiki/Lambert_w_function]

[MathWorld](http://mathworld.wolfram.com/LambertW-Function.html) [http://mathworld.wolfram.com/LambertW-Function.html]

zeta-Funktion

Liefert den Wert der Riemanschen Zetafunktion zum Argument.

Syntax

$\text{zeta}(z)$

Beschreibung:

Die zeta-Funktion liefert das Ergebnis der Riemanschen Zetafunktion zurück, allgemein beschrieben mit $\zeta(s)$. z kann jeder beliebige *numerischer Ausdruck* sein, der eine *reelle Zahl* oder eine *komplexe Zahl* darstellt.

Bemerkungen

Die zeta-Funktion ist definiert für die ganze komplexe Ebene ohne die Pole bei $z=1$.

Siehe auch

[Wikipedia](http://en.wikipedia.org/wiki/Riemann_zeta_function) [http://en.wikipedia.org/wiki/Riemann_zeta_function]

[MathWorld](http://mathworld.wolfram.com/RiemannZetaFunction.html) [http://mathworld.wolfram.com/RiemannZetaFunction.html]

mod-Funktion

Liefert den Rest der Division des ersten Arguments durch das zweite Argument.

Syntax

$\text{mod}(m,n)$

Beschreibung:

Berechnet m Modulo n , den Teilungsrest von m/n . mod berechnet den Teilungsrest f , wobei $m = a*n + f$ für eine Ganzzahl a ist. Das Vorzeichen von f entspricht immer dem Vorzeichen von n . Wenn $n=0=0$ ist, liefert mod 0 zurück. m und n können beliebige *numerische Ausdrücke* sein, die *reelle Zahlen* darstellen.

Siehe auch

[Wikipedia](http://en.wikipedia.org/wiki/Modular_arithmetic) [http://en.wikipedia.org/wiki/Modular_arithmetic]

[MathWorld](http://mathworld.wolfram.com/Congruence.html) [http://mathworld.wolfram.com/Congruence.html]

dnorm-Funktion

Liefert die Normalverteilung des ersten Arguments, optional mit Erwartungswert und Standardabweichung.

Syntax

dnorm(x , [μ , σ])

Beschreibung:

Die dnorm-Funktion ist die Wahrscheinlichkeitsdichte der Normalverteilung, auch: Gauß-Verteilung. x ist die Zufallsvariable, μ ist der Erwartungswert und σ ist die Standardabweichung. μ und σ sind optional.

Fehlen sie, so wird die Standard-Normalverteilung benutzt, wobei $\mu=0=0$ und $\sigma=1$ sind. x , μ und σ können beliebige *numerische Ausdrücke* sein, die *reelle Zahlen* darstellen, wobei $\sigma > 0$ ist. Die Normalverteilung ist folgendermaßen definiert:

$$\text{dnorm}(x, \mu, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{-\frac{(x-\mu)^2}{2\sigma^2}}$$

Siehe auch

[Wikipedia](http://en.wikipedia.org/wiki/Normal_distribution) [http://en.wikipedia.org/wiki/Normal_distribution]

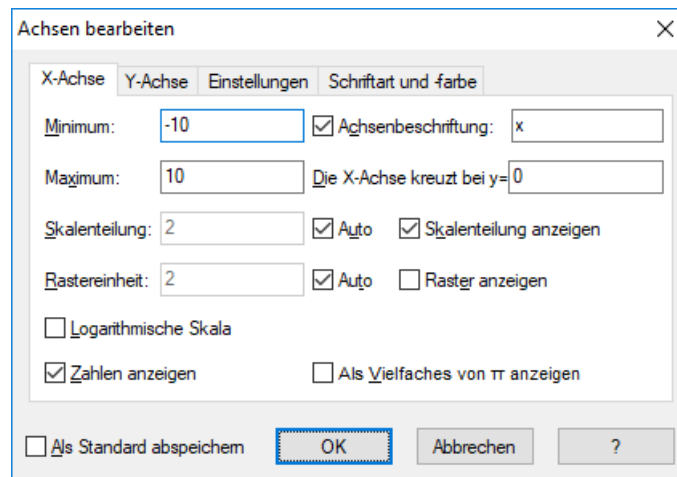
[MathWorld](http://mathworld.wolfram.com/NormalDistribution.html) [http://mathworld.wolfram.com/NormalDistribution.html]

Dialoge

Achsen bearbeiten

Wenn Sie den Menüpunkt **Bearbeiten** → **Achsen...** aufrufen, öffnet sich der unten abgebildete Dialog. Darin können Sie alle Einstellungen tätigen, die sich auf das Aussehen der Achsen des Koordinatensystems beziehen. Der Dialog besteht aus vier Registern. Im ersten Register (siehe unten) sind die Einstellungsmöglichkeiten für die X-Achse dargestellt, denen analog die Einstellungsmöglichkeiten der Y-Achse (zweites Register) entsprechen.

X-Achse/Y-Achse



Minimum

Dies bezeichnet den niedrigsten Wert der Achse. Standardwert: -10.

Maximum

Dies bezeichnet den höchsten Wert der Achse. Standardwert: 10.

Skalenteilung

Dies bezeichnet die Größe der Skalenteilung auf der jeweiligen Achse (Abstand zwischen den Hilfsstrichen). Diese kleinen Markierungen sind senkrecht zur Achse angebracht. Mit *Skalenteilung* werden sowohl Skalenteilung als auch die Beschriftung der Achse mit Zahlen geregelt. Wenn von der (standardmäßig verwendeten) linearen auf eine logarithmische Skala übergewechselt wird, dann wird mit Hilfe von *Skalenteilung* der Faktor zwischen den von zwei Hilfsstrichen abgeteilten Werten festgelegt. Beispielsweise sorgt das Setzen von *Skalenteilung* auf den Wert 4 dazu, dass bei einer linearen Achse die Werte 0, 4, 8, 12 usw. angezeigt werden, während bei einer logarithmischen Achse die Werte 1, 4, 16, 64 zu sehen wären.

Rastereinheit

Dies bezeichnet den Abstand der Gitternetzlinien des Rasters, die senkrecht zur Achse verlaufen. Es ist natürlich nur dann sinnvoll, diesen Wert zu bearbeiten, wenn das Raster auch angezeigt wird.

Logarithmische Skala

Aktivieren Sie diese Option, wenn die Achse eine logarithmische Skalierung erhalten soll.

Zahlen anzeigen

Aktivieren Sie diese Option, wenn an den Achsen entlang der (unter *Skalenteilung* einzustellenden) Skalenteilung die zugehörigen Achsenwerte angezeigt werden sollen.

Textfeld

Wenn diese Option aktiviert wird, beschriftet Graph die X-Achse auf der rechten Seite des Koordinatensystems mit dem angegebenen Text. Analog erfolgt die Beschriftung der Y-Achse, wobei

der Beschriftungstext am oberen Rand des Koordinatensystems rechts von der Y-Achse platziert wird. Auf diese Weise können Sie die Achsen sinnvoll beschriften, z.B. angeben, in welcher Maßeinheit die jeweilige Größe abgetragen ist.

Die X-Achse schneidet bei / Die Y-Achse schneidet bei:

Wenn *Achsenstil* auf *Gekreuzt* gesetzt wurde, gibt dies die Koordinate an, an der diese Achse die andere Achse schneidet. Standardwert: 0.

Auto

Wenn diese Option aktiviert wird, wählt Graph selbst einen passenden Wert für *Skalenteilung*, um das gesamte Koordinatensystem größtmäßig dem Fenster anzupassen.

Auto

Wird diese Option aktiviert, enthält *Rastereinheit* denselben Wert wie *Skalenteilung*.

Skalenteilung anzeigen

Wird diese Option aktiviert, erscheinen die Hilfsstriche der Skalenteilung auf der jeweiligen Achse mit dem unter *Skalenteilung* eingestellten Abstand.

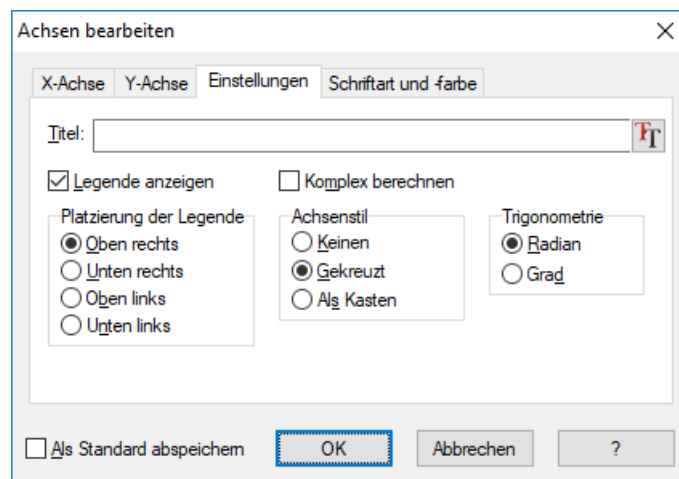
Raster anzeigen

Wird diese Option aktiviert, erscheint ein gepunktetes, senkrecht zu den Achsen stehendes Raster, dessen Farbe man unter *Schriftart und -farbe* wählen und dessen Breite man unter *Rastereinheit* festlegen kann.

Als Vielfaches von π anzeigen

Wird diese Option aktiviert, stellt Graph die Skalenteile als Vielfache von π dar, z.B. $3\pi/2$. Um diese Option nutzen zu können, müssen Sie zuvor *Zahlen anzeigen* einschalten.

Einstellungen



Titel

Hier können Sie eine Überschrift für Ihr Koordinatensystem festlegen. Die zu verwendende Schriftart können Sie über die rechts daneben stehende Schaltfläche einstellen.

Legende anzeigen

Wird diese Option aktiviert, zeigt Graph in der oberen rechten Ecke des Koordinatensystems die *Legende* an, die eine Liste aller Funktionen und Punktserien enthält. Die zu verwendende Schriftart können Sie unter *Schriftart und -farbe* festlegen.

Platzierung der Legende

Hier können Sie festlegen, in welcher der vier Ecken des Koordinatensystems Graph die *Legende* platzieren soll. Die Position der Legende können Sie auch durch einen Rechtsklick im Koordinatensystem verändern.

Komplex berechnen

Wird diese Option aktiviert, verwendet Graph *Komplexe Zahlen* zur Berechnung von Funktionsgraphen. Diese Option ist zu aktivieren, wenn bei der Berechnung der reellen Funktionswerte komplexe Zahlen als Zwischenergebnisse auftreten. Beachten Sie, dass sich dadurch die erforderliche Zeit erhöht, die Graph benötigt, um die Funktionsgraphen zu zeichnen, so dass diese Option meist nicht benötigt wird.

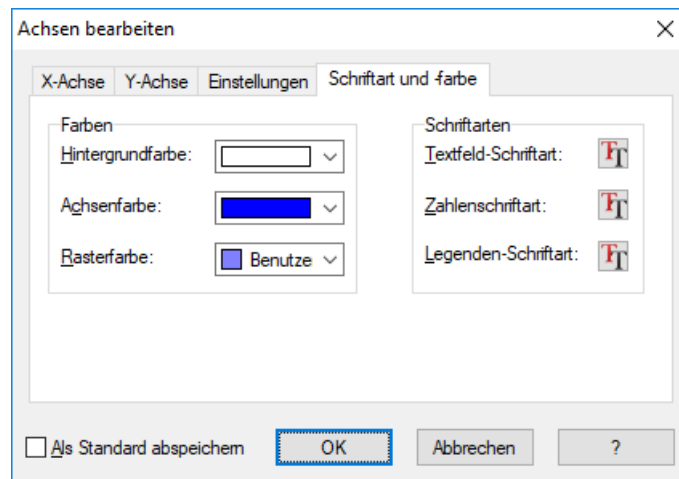
Achsenstil

Wenn die Option *Keinen* gewählt wird, blendet Graph die Achsen des Koordinatensystems aus. Entscheiden Sie sich für *Gekreuzt*, wenn Sie ein normales Koordinatensystem wünschen. Die Anordnung der Achsen kann mit Hilfe von *Die Y-Achse schneidet bei* und *Die X-Achse schneidet bei* beeinflusst werden. Wenn Sie *Als Kasten* auswählen, wird die X-Achse am unteren, die Y-Achse am linken Fensterrand platziert und die Einstellungen für *Die Y-Achse schneidet bei* und *Die X-Achse schneidet bei* überschrieben.

Trigonometrie

Legen Sie fest, ob die trigonometrischen Funktionen die Berechnungen in *Rad* oder in *Grad* ausführen sollen. Hiermit legen Sie auch fest, dass *Komplexe Zahlen* in polarer Form anzugeben sind.

Schriftart und -farbe



Farben

Sie können die Hintergrundfarbe des Koordinatensystems sowie die Farben für Achsen und Raster ändern.

Schriftarten

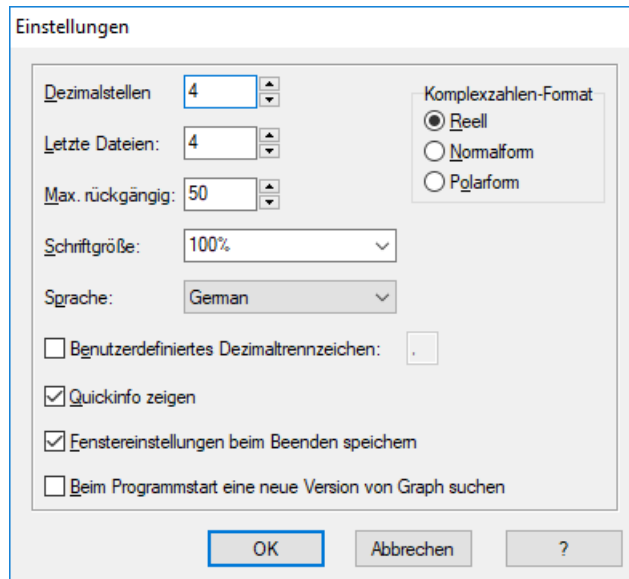
Sie können die Schriftarten ändern, die für Achsenbeschriftungen, Skalenteilung und *Legende* verwendet werden.

Als Standard abspeichern

Aktivieren Sie diese Option, um die von Ihnen getätigten Einstellungen als Standardeinstellung zu behalten. Wenn Sie beim nächsten Mal ein neues Koordinatensystem anlegen, werden diese Einstellungen diesem zugrunde gelegt. Die Einstellungen werden übrigens in ihrem Windows-Benutzerprofil gespeichert, so dass jeder Benutzer seine eigenen Standardeinstellungen einrichten kann.

Einstellungen

Mit dem Menü-Punkt *Bearbeiten* → *Einstellungen...* wird der folgende Dialog angezeigt. In diesem Dialog lassen sich allgemeine Programmeinstellungen ändern.



Dezimalstelle

Alle Ergebnisse werden mit dieser Anzahl Dezimalstellen angezeigt. Die Zahl hat keinen Einfluß auf Berechnungen oder die dargestellte Graphen.

Letzte Dateien

Dies ist die höchste Anzahl zuletzt benutzer Dateien, die im **Datei**-Menü angezeigt werden. Die Zahl muß zwischen 0 und 9 sein. 0 bedeutet, daß die zuletzt benutzten Dateien nicht angezeigt werden.

Max. rückgängig

Bei jeder von Ihnen gemachten Änderung, behält sich das Programm Informationen, um die Änderung rückgängig zu machen. Voreingestellt sind 50 *Max. rückgängig*; Sie können also die letzten 50 im Programm gemachten Änderungen zurücknehmen. Die Rücknahmeschritte verbrauchen Speicher. Falls Ihr System wenig RAM hat, können Sie durch Verringern von *Max. rückgängig* RAM-Platz freigeben.

Schriftgröße

Hiermit können Sie die Schriftgröße und die meisten Einstellungen der Benutzerschnittstelle ändern; besonders nützlich bei sehr hohen Bildschirmauflösungen oder schlechter Lesbarkeit.

Sprache

Dies zeigt eine Liste der für das Programm verfügbaren Sprachen. Das Programm wird in Zukunft die ausgewählte Sprache verwenden. Für jeden Benutzer kann eine andere Sprache gewählt werden.

Benutzerdefiniertes Dezimaltrennzeichen

Dezimaltrennzeichen für den Datenexport in Dateien und über die Zwischenablage. Wenn dieser Punkt nicht aktiviert ist, benutzt Graph die Standardeinstellungen von Windows. Diese Einstellung gilt nicht für Ausdrücke, die in Graph eingegeben werden. Hier wird immer der Punkt als Dezimaltrennzeichen verwendet.

Quickinfo zeigen

Bei markiertem Feld sehen Sie für einige Sekunden ein kleines Fenster mit einer Erläuterung, wenn der Maus-Zeiger über dem Objekt steht, (z. B. ein Edit-Feld, eine Selektion usw.). Die Beschreibung wird auch in der Statusleiste am unteren Rand des Hauptfensters angezeigt.

Fenstereinstellungen beim Beenden speichern

Bei markiertem Feld merkt sich Graph vor dem Beenden die Größe des Hauptfensters und übernimmt beim nächsten Start die gespeicherte Größe. Zusätzlich wird auch die Breite der *Funktionsliste* gespeichert. Ohne eine Markierung werden die zuletzt gespeicherten Einstellungen benutzt.

Komplexzahlen-Format

Wählen Sie hier, wie komplexe Zahlen im **Evaluieren**-Fenster dargestellt werden sollen. *Reell* bedeutet, daß nur *reelle Zahlen* angezeigt werden. Hat eine Zahl einen imaginären Anteil, erhalten Sie statt der Zahl

einen Fehler. *Normalform* bedeutet, daß *Komplexe Zahlen* als $a+bi$ dargestellt werden, mit a als Realteil und b als Imaginärteil. *Polarform* bedeutet, daß Zahlen als $a\angle\theta$ dargestellt werden, mit a als Absolutwert und θ als Winkel. θ ist abhängig von der Einstellung *Rad* oder *Grad* unter *Trigonometrie* im *Achsen bearbeiten*-Dialog.

Beachten Sie, daß Sie in der Eingabemaske *Evaluieren* in einigen Fällen - je nach *Komplexzahlen-Format*-Einstellung - unterschiedliche Resultate erhalten können: Bei *Reell* versucht Graph, ein reelles Ergebnis zu finden, während *Normalform* und *Polarform* ein nicht-reelles Ergebnis für dieselbe Berechnung liefern können.

Beim Programmstart eine neue Version von Graph suchen

Wenn aktiviert, wird bei jedem Programmstart im Internet nach einer neuen Version von Graph gesucht.

Wenn eine neue Version gefunden wurde, werden Sie gefragt, ob Sie die Webseite für ein Update

besuchen wollen; ansonsten erscheint keine Meldung. Wenn nicht aktiviert, können Sie mit ? → Internet

→ Nach Aktualisierung suchen nach einer neuen Version sehen.

Funktion einfügen

Mit Funktion → Funktion einfügen... fügen Sie eine neue Funktion ein; es erscheint der untenstehende

Dialog. Um eine Funktion zu ändern, wählen Sie sie in der *Funktionsliste* aus und verwenden dann Funktion

→ Bearbeiten....

Funktionstyp

Sie können zwischen drei verschiedenen Funktionstypen wählen: *Standardfunktion*, *Parameterfunktion* und *Polarfunktion*. Eine Standardfunktion wird mit $y=f(x)$ definiert; d.h. zu jeder X-Koordinate gibt es genau eine Y-Koordinate, obwohl sie für einige X-Werte undefiniert sein kann. Für eine Parameterfunktion werden die X- und Y-Werte aus einer unabhängigen Variablen t - Parameter genannt - berechnet, d.h. die Parameterfunktion wird als zwei Funktionen definiert: $x(t)$ und $y(t)$.

Eine Polarfunktion $r(t)$ bezeichnet eine Gleichung zur Berechnung der Entfernung vom Ursprungspunkt zu einem Punkt der Funktion mit dem Winkel t . t ist der direkte Winkel zwischen dem ursprünglichen Strahl und dem Punkt der Funktion. D. h., daß für die X- und Y-Koordinaten $x(t)=r(t)\cdot\cos(t)$, $y(t)=r(t)\cdot\sin(t)$ gilt.

Funktionsgleichung

Hier geben Sie die Funktionsgleichung ein. Sie kann $f(x)$, $x(t)$, $y(t)$ oder $r(t)$ sein, abhängig vom Funktionstyp. Unter [Funktionsliste](#) können Sie alle verfügbaren Variablen, Konstanten und Funktionen sehen, mit denen sich den Graphen zeichnen lassen.

Definitionsmenge

Sie können einen Intervall für die unabhängige Variable wählen. *Von* und *Bis* geben Start und Ende des Intervalls an. Bei einer Standardfunktion können Sie ein oder beide Felder freilassen, um z. B. den Graphen von minus unendlich bis plus unendlich zu zeichnen. Bei einer Parameter- oder Polarfunktionen muss immer ein Intervall angegeben werden, außerdem die Anzahl der Schritte, in denen die Funktionen ausgewertet werden soll. Je mehr Schritte, desto glatter wird die Kurve, aber desto länger dauert das Zeichnen. Es wird empfohlen, das Feld *Schritte* bei Standardfunktionen frei zu lassen, damit Graph die optimale Anzahl der Schritte wählt. Sie können jedoch selbst die Anzahl der Schritte eingeben, wenn der Graph nicht genug Einzelheiten darstellt; zum Beispiel wenn eine Asymptote nicht korrekt dargestellt wird. Beachten Sie, daß *Schritte* nur die Mindestanzahl von Berechnungen angibt. Graph fügt eventuell Schritte an kritischen Punkten hinzu, wenn *Darstellungsart* auf *Automatisch* steht.

Endpunkte

Hier können Sie wählen, ob am Anfang und/oder Ende des Intervalls Markierungen angezeigt werden. Ist kein Intervall angegeben, werden die Endpunkte dort angezeigt, wo der Funktionsgraph in die Zeichenfläche eintritt und sie verläßt. In der Voreinstellung werden keine Markierungen angezeigt.

Legende

Geben Sie eine Beschreibung für die *Legende* ein. Wenn leer, wird in der Legende die Funktionsgleichung angezeigt.

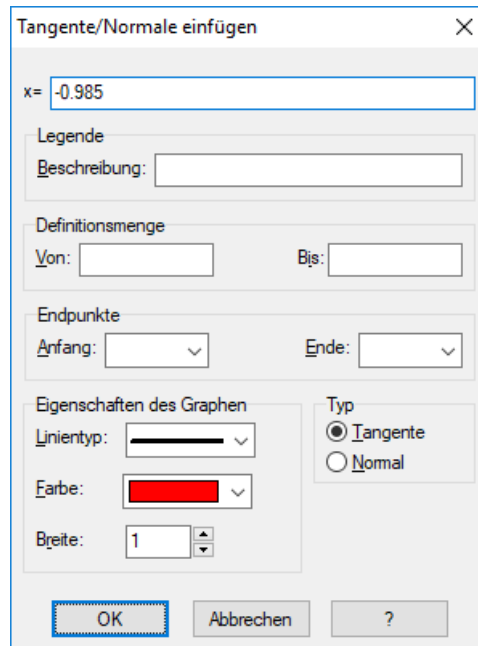
Eigenschaften des Graphen

Zur Zeichnung der Funktionsgraphen können Sie zwischen verschiedenen Linienarten wählen. Zur Wahl stehen durchgezogene, gestrichelte, gepunktete oder abwechselnd gestrichelte und gepunktete Linien. *Linientyp* stehen nur dann zur Verfügung, wenn unter *Darstellungsart Verlauf* oder *Automatisch* eingestellt wird. Wird unter *Darstellungsart Punkte* gewählt, wird anstelle einer Kurve nur ein Punkt für jeden berechneten Datenpunkt dargestellt. Ebenso führt die Einstellung von *Verlauf* unter *Darstellungsart* dazu, dass die berechneten Punkte durch Linien verbunden werden. Die Option *Automatisch* führt ebenfalls dazu, dass Linien gezeichnet werden, jedoch zeichnet Graph mit einer höheren Präzision. Sollte Graph einen asymptotischen Verlauf feststellen, wird die Verbindungslinie unterbrochen. Sie können auch die Breite des Graphen (in Bildpunkten / Pixeln) festlegen und aus mehreren Farben wählen. Graph speichert die Einstellungen und startet beim nächsten Mal mit diesen Einstellungen.

Tangente/Normale einfügen

Mit dem unten stehenden Dialog können Sie eine Tangente oder Normale zu einer Funktion hinzufügen oder bearbeiten. Mit Funktion → Tangente/Normale einfügen... fügen Sie eine neue Tangente oder Normale hinzu. Um eine Tangente zu ändern, wählen Sie sie zuerst in der *Funktionsliste* aus und verwenden dann Funktion → Bearbeiten....

Eine Tangente ist eine gerade Linie, die den Funktionsgraphen in einem definierten Punkt berührt, ohne ihn dort zu kreuzen. Die Tangente darf jedoch den Graphen an anderen Stellen kreuzen. Eine Normale ist eine gerade Linie, die in einem definierten Punkt rechtwinklig zur Funktion ist. Bei Standard-Funktionen wird der Punkt durch die X-Koordinate identifiziert, und bei Parameter- und Polarfunktionen durch den unabhängigen t-Parameter.



Definitionsmenge

Sie können einen Intervall für die Tangente/Normale wählen. *Von* und *Bis* geben den Start und das Ende des Intervalls an. Sie können eins oder beide Felder freilassen, um den Graphen von minus unendlich bis plus unendlich zu zeichnen.

Endpunkte

Hier können Sie wählen, ob am Anfang und/oder Ende des Intervalls Markierungen angezeigt werden. Ist kein Intervall angegeben, werden die Markierungen am Rand der Zeichenfläche angezeigt. In der Voreinstellung werden keine Markierungen angezeigt.

Legende

Geben Sie eine Beschreibung für die *Legende* ein. Wenn leer, wird die Funktionsgleichung angezeigt.

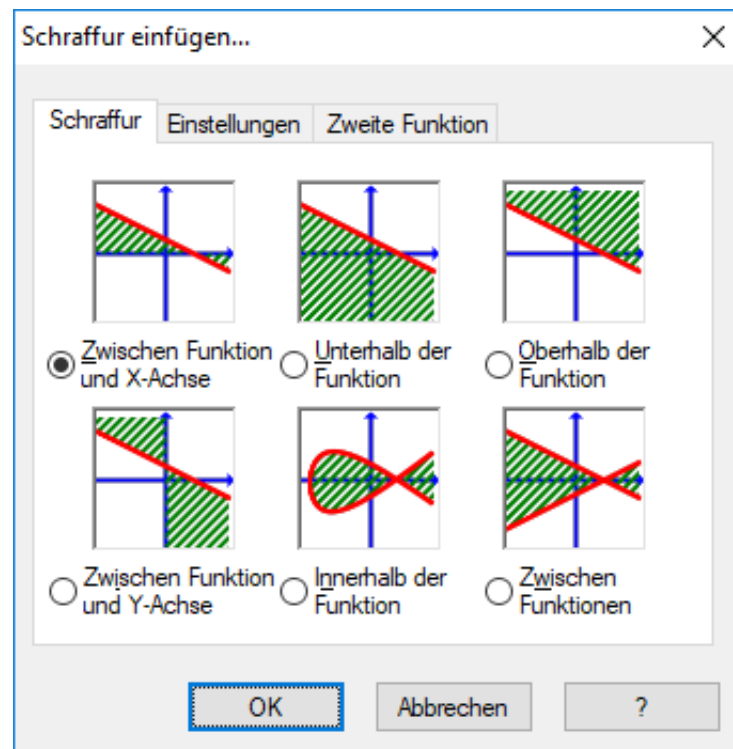
Eigenschaften des Graphen

Sie können verschiedene Linientypen für die Tangente/Normale wählen. Sie können ausgefüllte, gestrichelte, getüpfelte und deren Kombinationen aussuchen, ebenso wie die Breite der Graph-Linie, die in Pixel angegeben wird. Außerdem können Sie zwischen vielen verschiedenen Farben wählen.

Schraffur einfügen...

Mit dem untenstehenden Dialog können Sie einer Funktion eine Schraffur hinzufügen. Mit **Funktion → Schraffur einfügen...** fügen Sie eine neue Schraffur hinzu. Um eine Schraffur zu ändern, wählen Sie sie zuerst in der *Funktionsliste* aus und verwenden dann **Funktion → Bearbeiten....** Die Schraffur füllt die Fläche zwischen dem Funktionsgraphen und einem anderen Element.

Schraffur



Im Reiter *Schraffur* können Sie zwischen den folgenden Schraffurtypen auswählen:

Zwischen Funktion und X-Achse

Dies ist der meistbenutzte Schraffurtyp. Er schraffiert den Bereich zwischen dem Funktionsgraphen und der X-Achse im gewählten Intervall. Mit *Verringern bis Schnittpunkt* oder *Erhöhen zum Schnittpunkt* wird der Intervall verkleinert oder vergrößert, bis der Graph die X-Achse kreuzt.

Zwischen Funktion und Y-Achse

Es wird der Bereich zwischen dem Funktionsgraphen und der Y-Achse im gewählten Intervall schraffiert. Dieser Typ wird selten benutzt und ist wohl für Parameterfunktionen am sinnvollsten. Beachten Sie, daß Sie für die Intervallgrenzen die X-Koordinaten benutzen. Mit *Verringern bis Schnittpunkt* oder *Erhöhen zum Schnittpunkt* wird der Intervall verkleinert oder vergrößert, bis der Graph die Y-Achse kreuzt.

Unterhalb der Funktion

Es wird der Bereich zwischen dem Funktionsgraphen und dem unteren Rand des Zeichenbereichs im gewählten Intervall schraffiert. Mit *Verringern bis Schnittpunkt* oder *Erhöhen zum Schnittpunkt* wird der Intervall verkleinert oder vergrößert, bis der Graph den unteren Rand des Zeichenbereichs kreuzt.

Oberhalb der Funktion

Es wird der Bereich zwischen dem Funktionsgraphen und dem oberen Rand des Zeichenbereichs im gewählten Intervall schraffiert. Mit *Verringern bis Schnittpunkt* oder *Erhöhen zum Schnittpunkt* wird der Intervall verkleinert oder vergrößert, bis der Graph den oberen Rand des Zeichenbereichs kreuzt.

Innerhalb der Funktion

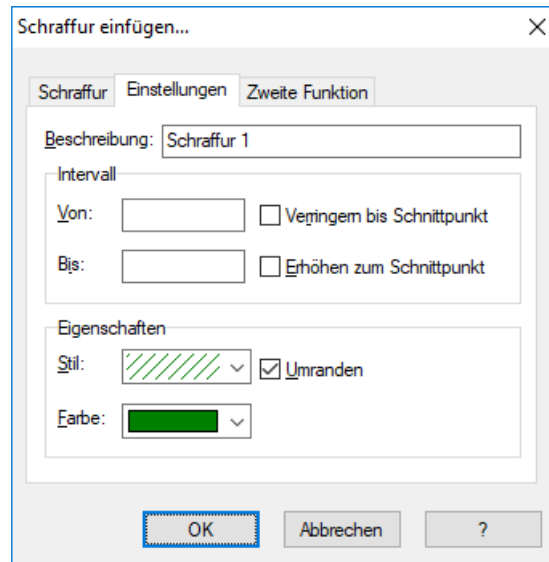
Es wird der Bereich innerhalb des Funktionsgraphen im gewählten Intervall schraffiert. Mit *Verringern bis Schnittpunkt* oder *Erhöhen zum Schnittpunkt* wird der Intervall verkleinert oder vergrößert, bis der Graph sich selbst kreuzt. Dies ist besonders nützlich, um einen umschlossenen Teil einer Parameter- oder Polarfunktion zu schraffieren, läßt sich aber auch für die Schraffur von Standardfunktionen verwenden.

Zwischen Funktionen

Es wird der Bereich zwischen den Graphen zweier Funktionen schraffiert. Die erste Funktion wurde in der *Funktionsliste* im Hauptfenster vor dem Aufruf des Dialogs gewählt. Die zweite Funktion wählen Sie in der Liste im Reiter *Zweite Funktion*. Bei Standardfunktionen gilt für beide Funktionen derselbe Intervall. Bei Parameterfunktionen können Sie unterschiedliche Intervalle für die beiden Funktionen wählen. Wenn Sie keinen Intervall für die zweite Funktion wählen, gilt für sie derselbe Intervall wie für die erste.

Einstellungen

Im untenstehenden Reiter *Einstellungen* können Sie die Schraffureinstellungen ändern.



Von

Hier können Sie den Wert eingeben, an dem die Schraffur beginnen soll. Bei einer Standardfunktion geben Sie die X-Koordinate ein, bei einer Parameter- oder Polarfunktion den t-Parameter. Wenn Sie keinen Wert eingeben, startet die Schraffur bei minus unendlich. Wenn Sie *Verringern bis Schnittpunkt* aktivieren, wird die Startkoordinate der Schraffur vom eingegebenen Wert bis zu der Koordinate erhöht, an der der Graph die Achse, den Rand der Zeichenfläche, sich selbst oder einen anderen Graphen kreuzt, abhängig vom gewählten Schraffurtyp.

Bis

Hier können Sie den Wert eingeben, an dem die Schraffur enden soll. Bei einer Standardfunktion geben Sie die X-Koordinate ein, bei einer Parameter- oder Polarfunktion den t-Parameter. Wenn Sie keinen Wert eingeben, setzt sich die Schraffur bis plus unendlich fort. Wenn Sie *Erhöhen zum Schnittpunkt* aktivieren, wird die Endkoordinate der Schraffur vom eingegebenen Wert bis zu der Koordinate erhöht, an der der Graph die Achse, den Rand der Zeichenfläche, sich selbst oder einen anderen Graphen kreuzt, abhängig vom gewählten Schraffurtyp.

Stil

Hier können Sie zwischen verschiedenen Schraffurtypen wählen.

Farbe

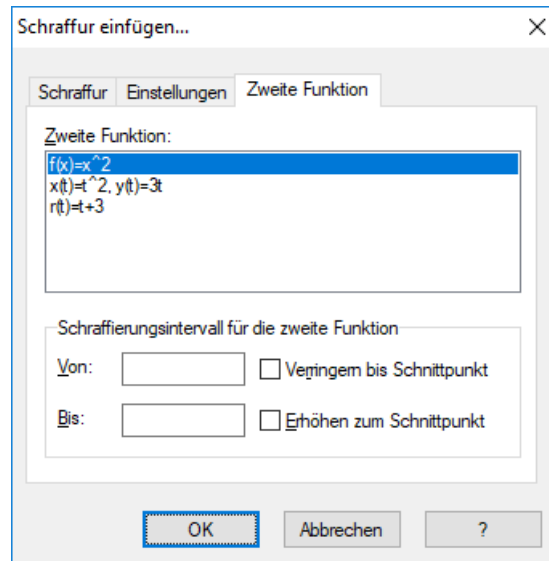
Hier können Sie die Farbe der Schraffur auswählen.

Umranden

Aktivieren, um die Schraffur mit einer Umrandung zu versehen. Deaktivieren, um zwei separate Schraffuren wie eine einzige zusammenhängende Schraffur aussehen zu lassen.

Zweite Funktion

Wenn Sie im Reiter *Schraffur* die *Zwischen Funktionen* gewählt haben, können Sie im Reiter *Zweite Funktion* die zweite Funktion aussuchen. Der Dialog des Reiters *Zweite Funktion* wird unten gezeigt.



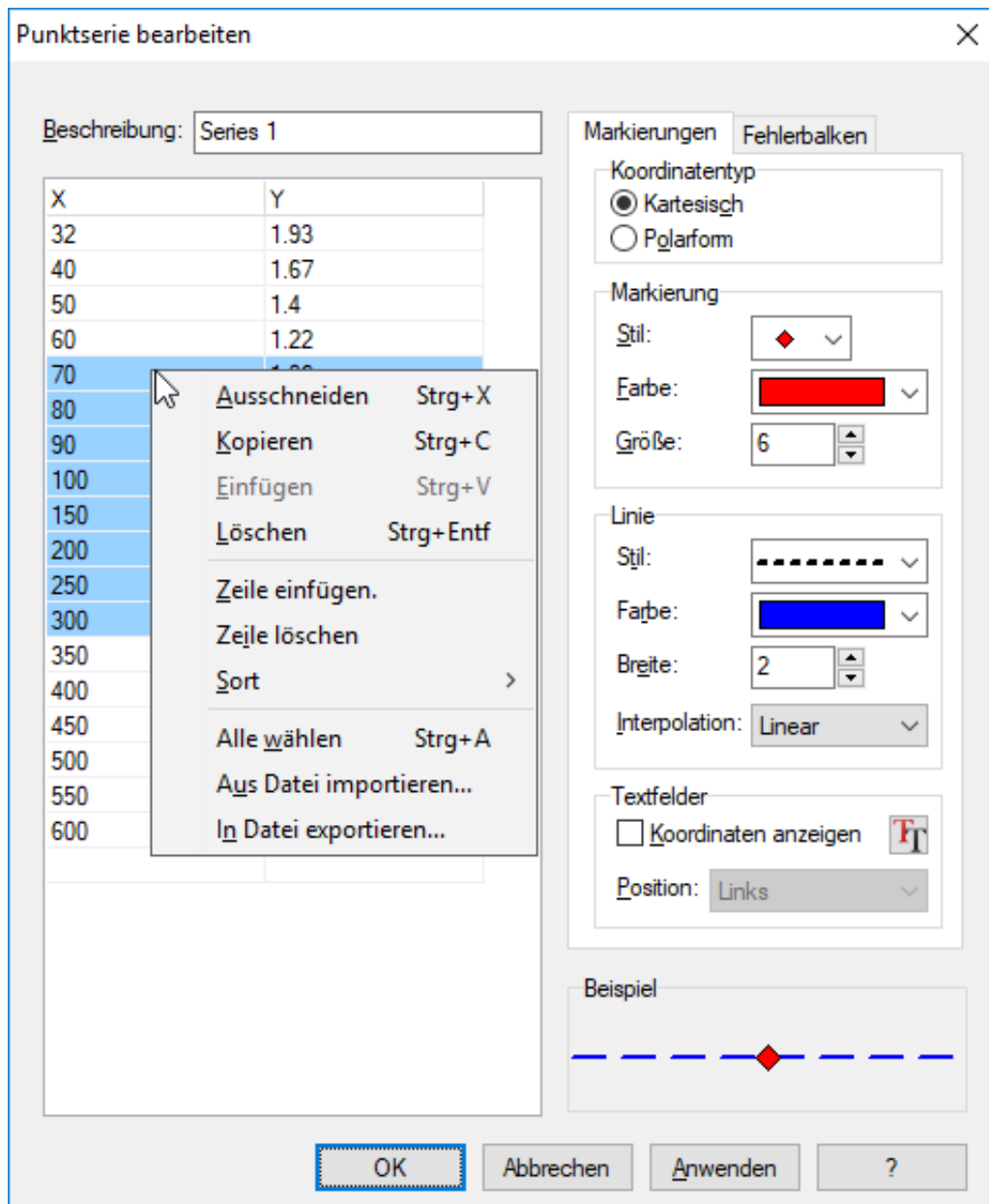
Schraffierungsintervall für die zweite Funktion

Hiermit wählt man den Intervall für die zweite Funktion, genauso wie Sie im Reiter *Einstellungen* den Intervall für die erste Funktion eingestellt haben. Wenn Sie bei einer Parameterfunktion weder Intervall-Beginn noch -Ende angeben, so werden für die zweite Funktion dieselben Werte wie für die erste verwendet.

Schraffuren eignen sich gut zum Markieren einer Fläche. Falls Sie eigenartige Ergebnisse bekommen, ob Funktion und Intervall richtig gewählt wurden. Wenn die Schraffur eine Asymptote kreuzt oder die Schraffur einer seltsamen Parameterfunktion zugeordnet ist, könnten sonderbare Resultat vorkommen. Aber mal ehrlich, was haben Sie denn sonst erwartet?

Punktserie einfügen

Mit dem untenstehenden Dialog können Sie dem Koordinatensystem eine Serie von Punkten hinzufügen. Die Punkte werden im Koordinatensystem in der Zeichenfläche als eine Serie von Markierungen dargestellt. Mit Funktion → Punktserie einfügen... fügen Sie eine neue Punktserie ein. Um eine Punktserie zu ändern, wählen Sie sie zuerst in der *Funktionsliste* aus und verwenden dann Funktion → Bearbeiten....



Sie können einer Punktserie eine **Trendlinie** hinzufügen; das ist die Kurve, die am besten mit den Punkten übereinstimmt.

In die Tabellenfelder geben Sie die X- und Y-Koordinaten der Punkte ein. Sie können eine beliebige Anzahl eingeben, aber jeder Punkt muß sowohl eine X- als auch eine Y-Koordinate haben.

Selektierte Koordinaten lassen sich in ein anderes Programm zu kopieren. Ähnlich lassen sich auch Daten aus anderen Programmen - wie MS Word oder MS Excel - kopieren und in die Tabelle einfügen.

Das Kontext-Menü (Rechtsklick) Rechtsklick-Menü erlaubt auch, Daten aus einer Datei einzufügen. Graph unterstützt zwei Formate: tabulatorgetrennte und semikolongetrennte Dateien. Daten werden an der Cursorposition plaziert. Damit ist es möglich, Daten von mehreren Dateien einzufügen, oder die X- und Y-Koordinaten jeweils aus unterschiedlichen Dateien. Im Normalfall, wenn sich alle Daten in derselben Datei befinden, sollten Sie sich deshalb vergewissern, daß der Cursor vor dem Einfügen im obersten linken Feld steht.

Beschreibung:

Im Feld oben im Dialog können Sie einen Namen für die Punktserie angeben, der in der *Legende* angezeigt werden soll.

Koordinatentyp

Für die Punkte müssen sie den Koordinatentyp festlegen. *Kartesisch* bedeutet, dass (x,y)-Koordinaten verwendet werden sollen, und *Polarform* legt (θ,r)-Koordinaten fest, wobei θ Winkel und r Abstand vom Koordinatenursprung bezeichnen. Der Winkel θ wird in *Bogenmaß* oder Grad angegeben, je nach aktueller Einstellung.

Markierung

Rechts können Sie zwischen verschiedenen Punkt-Typen wählen, z. B. Kreis, Quadrat, ein Dreieck usw. . Farbe und Größe der Punkte lassen sich ebenfalls bearbeiten. Bei der Größe 0 werden keine Punkte oder Fehlerbalken angezeigt..

Wenn die Markierung ein Pfeil ist, zeigt dieser am betreffenden Punkt tangential auf die Linie. Die tatsächliche Richtung hängt von der Einstellung *Interpolation* ab. Der erste Punkt der Serie wird bei einem Pfeil als Markierung nie dargestellt.

Linie

Man kann zwischen die Markierungen Linien zeichnen. Die Reihenfolge der Linien ist dieselbe wie in der Tabelle. Für die Linien lassen sich Stil, Farbe und Breite wählen. Man kann auch einstellen, gar keine Linie zu zeichnen.

Vier Typen der Interpolation stehen zur Wahl: *Linear* zeichnet eine gerade Linie zwischen die Markierungen. *Kubische Spline 1D* zeichnet eine [Natürlicher Kubischer Spline](http://en.wikipedia.org/wiki/Cubic_splines) [http://en.wikipedia.org/wiki/Cubic_splines], eine Kurve auf Basis eines Polynoms dritten Grades, die alle Punkte entlang der x-Koordinate verbindet. *Kubische Spline 2D* zeichnet einen kubischen Spline nacheinander durch alle Punkte. *Trigonometrisch* zeichnet Halb-Cosinus-Kurve zwischen die Punkte, die evtl. nicht so geschwungen ist wie kubische Splines, aber auch nicht stark ausschlagen wie es bei diesen möglich ist.

Textfelder

Mit der Option *Koordinaten anzeigen* werden die X- und Y-Koordinaten an jedem Punkt angezeigt.

Mit **T** lässt sich die Schriftart ändern, ebenso die Auswahl, ob die Textfelder oberhalb, unterhalb, links oder rechts der Punkte angezeigt werden.

Fehlerbalken

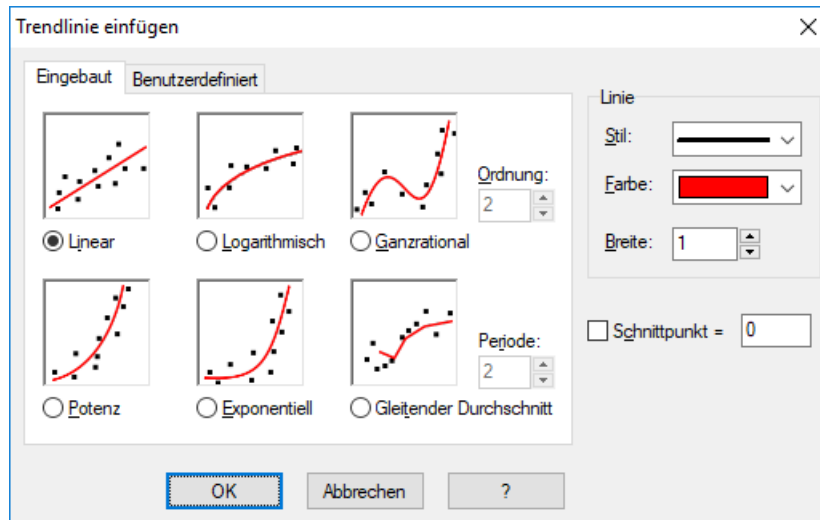
Hier können Sie sich waagerechte oder senkrechte Fehlerbalken - auch bekannt als Toleranzbalken - anzeigen lassen. Als dünne Balken an jedem Punkt der Punktserie zeigen Sie Ungenauigkeiten dieses (Mess-)Punkts an. Es gibt drei Arten, die Größe der Fehlerbalken darzustellen: Mit *Fest* gibt man an, dass alle Punkte dieselbe Abweichung haben. Mit *Relativ* gibt man für jeden Punkt einen Prozentsatz der X- oder Y-Koordinate als Abweichung an. *Benutzerdefiniert* erweitert die Tabelle um eine gesonderte Spalte, in der man für jeden Punkt eine andere Abweichung festlegen kann. Alle Abweichungen sind vorzeichenbehaftet. Benutzerdefinierte Y-Abweichungen werden auch bei der Erzeugung einer Trendlinie zur Gewichtung der Punkte verwendet.

Trendlinie einfügen

Verwenden Sie den unten gezeigten Dialog, um eine Trendlinie einzufügen; das ist die Funktion, die mit einer [Punktserie](#) am besten übereinstimmt. Eine Trendlinie ist eine Funktion, die den Trend bzw. einer Anzahl von Punkten zeigt, d. h. es ist die für diese Punkte am besten passende Kurve eines bestimmten Typs. Die Trendlinie wird als gewöhnliche Funktion hinzugefügt. Um eine Trendlinie zu erzeugen, selektieren Sie die gewünschte Punktserie und wählen dann Funktion → Trendlinie einfügen....

Enthält die Punktserie benutzerdefinierte Y-Fehler, so werden diese zur Wichtung der Punkte verwendet. Die Wichtung für jeden Punkt ist $1/\sigma^2$ mit σ als seinem Y-Fehler. X-Fehler werden nicht benutzt.

Eingebaut



Sie können zwischen den folgenden eingebauten Funktionen wählen. Diese Funktionen liefern ein genaues Ergebnis. Für *Linear*-, *Ganzrational*- und *Exponentiell*-Trendlinien können Sie das *Schnittpunkt*-Feld aktivieren und den Punkt angeben, an dem die Trendlinie die Y-Achse treffen soll.

Linear

Dies ist eine Gerade mit der Gleichung $f(x) = a \cdot x + b$, wobei die Konstanten a und b so ermittelt wurden, daß die Linie sich möglichst gut den Punkten annähert.

Die Trendlinie wird so berechnet, daß die Summe der Quadrate $\sum (y_i - f(x_i))^2$ möglichst klein wird. Wenn möglich, durchkreuzt die Funktion die Punkte; andernfalls ist die Funktion so nah an der Punktserie, das die Summe nicht kleiner werden kann.

Logarithmisch

Eine logarithmische Trendlinie folgt der Gleichung $f(x) = a \cdot \ln(x) + b$, wobei a und b Konstanten sind und \ln der natürliche Logarithmus. Sie läßt sich nur hinzufügen, wenn kein Punkt der Serie eine X-Koordinate hat, die negativ oder 0 ist.

Eine logarithmische Funktion ist eine Gerade in einem halblogarithmischen Koordinatensystem.

Die Punktserie wird deshalb für ein halblogarithmisches Koordinatensystem umgerechnet, und die logarithmische Funktion mit der Summe der kleinsten Quadrate wird ermittelt.

Ganzrational

Eine Polynom ist eine Funktion, die mit $f(x) = a_n \cdot x^n + \dots + a_3 \cdot x^3 + a_2 \cdot x^2 + a_1 \cdot x + a_0$ dargestellt wird. $a_0 \dots a_n$ sind Konstanten. n ist der Grad des Polynoms. Die Anzahl der Punkte muß höher als der Grad sein.

Potenz

Eine Potenzfunktion wird mit $f(x) = a \cdot x^b$ angegeben, wobei die Konstanten a und b so ermittelt wurden, daß die Funktion sich möglichst gut den Punkten annähert. Sie läßt sich nur hinzufügen, wenn kein Punkt der Serie eine X- oder Y-Koordinate hat, die negativ oder 0 ist.

Eine Potenzfunktion ist eine Gerade in einem doppellogarithmischen Koordinatensystem. Die Punktserie wird deshalb in ein doppellogarithmisches Koordinatensystem übertragen, und die Potenzfunktion mit der kleinsten Summe der Quadrate wird ermittelt.

Exponentiell

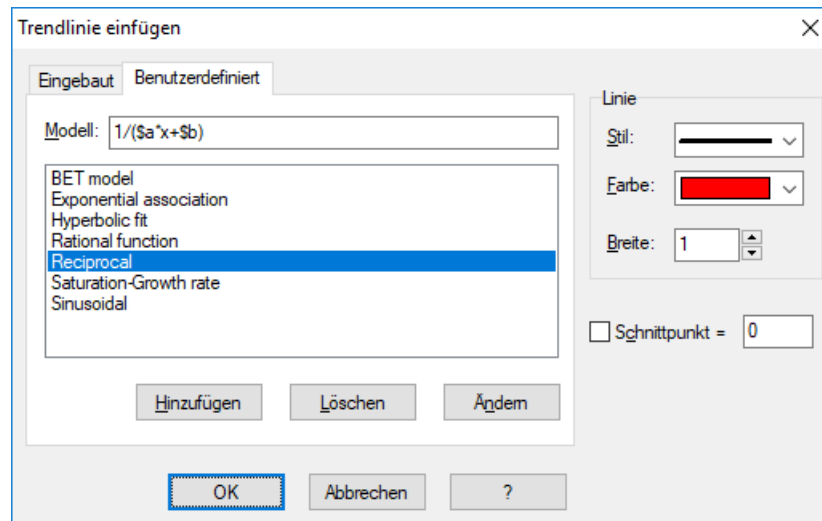
Eine Exponentialfunktion wird mit $f(x) = a \cdot b^x$ angegeben, wobei die Konstanten a und b so ermittelt wurden, daß die Funktion sich möglichst gut den Punkten annähert. Sie läßt sich nur hinzufügen, wenn kein Punkt der Serie eine Y-Koordinate hat, die negativ oder 0 ist.

Eine Exponentialfunktion ist eine gerade Linie in einem halblogarithmischen Koordinatensystem mit der Y-Achse als logarithmischer Achse. Die Punktserien werden deshalb in ein halblogarithmisches Koordinatensystem übertragen, und die Exponentialfunktion mit möglichst kleiner Summe der Quadrate wird ermittelt.

Gleitender Durchschnitt

Der Gleitende Mittelwert ist eine Reihe von geraden Linien basierend auf dem Durchschnitt der vorhergehenden Punkte. Die *Periode* gibt an, aus wievielen Punkten der Mittelwert errechnet wird. Ist *Periode* gleich 1, wird nur ein Punkt benutzt (das ist eigentlich kein Mittelwert). Es wird eine Linie direkt zwischen den Punkten gezogen. Ist *Periode* größer als 1, so berührt die Linie nicht zwangsläufig alle dazwischenliegenden Punkte.

Benutzerdefiniert



In diesem Reiter können Sie ihre eigenen Trendlinienmodelle eingeben. Ein Modell wird wie eine normale Funktion eingegeben, in der alle gesuchten Konstanten wie folgt benannt werden: Ein \$ gefolgt von einem oder mehreren Buchstaben (a-z) und Zahlen (0-9). Gültige Namen sind z. B. \$a, \$y0 und \$const.

Ein Beispiel eines Modells wäre $f(x) = \$a \cdot x^{\$b} + \$c$. Das Programm versucht, die Konstanten $\$a$, $\$b$ und $\$c$ so festzulegen, daß $f(x)$ sich möglichst gut den Punkten annähert. Mit der Schaltfläche **Hinzufügen** läßt sich das Modell unter einem frei wählbaren Namen abspeichern.

Das Programm benötigt eine Schätzung für den Start der Berechnung der optimalen Funktion. Die Standardschätzung für alle Konstanten ist 1, aber diese kann für die Modelle in der Liste geändert werden. Je besser die Schätzung ist, desto größer ist die Chance, ein Optimum zu finden.

Graph versucht, die Konstanten des Modells $f(x)$ so zu bestimmen, daß die Summe der Quadrate $\sum (y_i - f(x_i))^2$ möglichst klein wird. Das Programm setzt zunächst die Schätzwerte ein und optimiert sie für ein Minimum der Summe der Quadrate. Wurde nach 100 Iterationen keine Lösung gefunden oder ist die gegebene Schätzung ungültig, so beendet das Programm die Berechnung.

In seltenen Fällen existiert mehr als ein Minimum. In diesem Fall wird das Minimum gefunden, das am nächsten an der Schätzung liegt, obwohl es vielleicht nicht das beste ist.

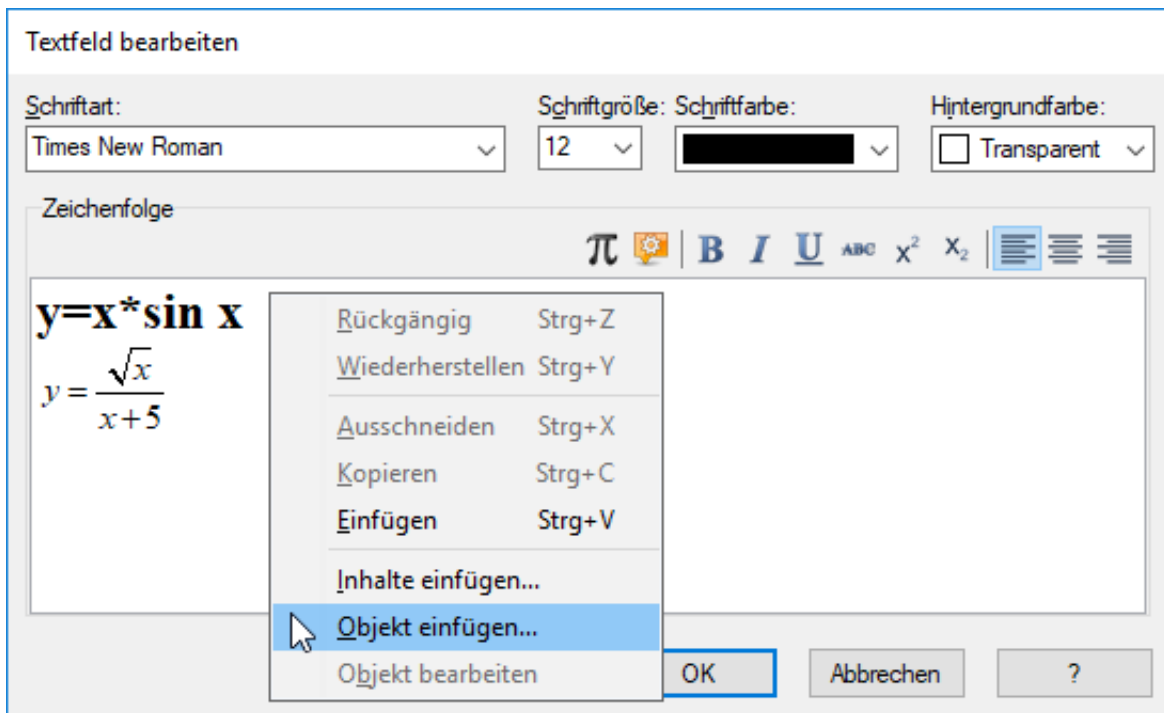
Beachten Sie, dass Sie redundante Konstanten vermeiden sollten, weil sie das Programm "verwirren" könnten. Im folgenden Modell ist beispielsweise eine redundante Konstante enthalten: $f(x) = \$c + \$d / (\$a \cdot x + \$b)$. Beachten Sie die Beziehung zwischen den Konstanten $\$a$, $\$b$ und $\$d$. Multipliziert man $\$a$, $\$b$ und $\$d$ mit demselben Wert, ändert sich die resultierende Funktion nicht. Damit gibt es unendlich viele Konstanten-Kombinationen mit derselben resultierenden Funktion und folglich unendlich viele beste Lösungen. Daher sollte entweder $\$a$, $\$b$ oder $\$d$ entfernt werden.

Nach dem Hinzufügen der Trendlinie wird der Korrelationskoeffizient R^2 im Kommentar angezeigt. Je näher R^2 an 1 liegt, desto besser stimmt die Trendlinie mit den Punkten überein.

Textfeld einfügen

Mit diesem Dialog können Sie Textfelder hinzufügen oder bearbeiten. Mit Funktion \rightarrow **Textfeld einfügen...** fügen Sie ein Textfeld hinzu. Das Textfeld wird in der Mitte der Zeichenfläche eingefügt, kann aber an eine

andere Position verschoben werden. Ein Textfeld ändern Sie entweder mit einem double click darauf, oder Sie wählen es in der *Funktionsliste* aus und verwenden dann Funktion → Bearbeiten....



Der Text wird ins Editierfeld eingegeben. Sie können den Textstil für verschiedene Textteile ändern. Die Hintergrundfarbe, die undurchsichtig oder transparent sein kann, gilt grundsätzlich für das ganze Textfeld. Mit der Schaltfläche π lassen sich Sonderzeichen wie mathematische Symbole oder griechische Buchstaben einfügen.

Ein Textfeld kann auch ein **OLE-Objekt** enthalten, z. B. ein Bild oder MS Gleichung. Sie können ein OLE-Objekt genauso wie Text auf die Bearbeitungsfläche bringen. Durch **Objekt einfügen** im Kontext-Menü kann an der Cursor-Position ein neues Objekt erzeugt werden. Bei mehr als einem OLE-Objekt in der Zwischenablage läßt sich mit **Inhalte einfügen** im Kontext-Menü das einzufügende Objekt aussuchen.

Mit der Schaltfläche **OK** wird das Textfeld in der Zeichenfläche angezeigt. Das Textfeld läßt sich mit der Maus verschieben. Durch Rechtsklick auf das Textfeld und nachfolgende Platzierung im Kontext-Menü kann es an einer der Achsen einrasten. Aus dem Kontext-Menü heraus läßt sich das Textfeld auch rotieren, z. B. um den Text senkrecht anzuzeigen.

Ein Textfeld kann einen *numerischer Ausdruck* enthalten und berechnen. Dies eignet sich sehr gut für die Anzeige von **benutzerdefinierten Konstanten** in einem Textfeld. Graph versuchen, alle Ausdrücke in einem Textfeld zu berechnen, die in Klammern hinter einem Prozentzeichen stehen (%). Wenn Sie 3 benutzerdefinierte Konstanten $a=2.5$, $b=-3$, und $c=8.75$ haben, können Sie ein Textfeld mit dem Text $f(x) = \%(a)x^2 + \%(b)x + \%(c)$ erstellen. Dieses Textfeld wird auf der Zeichenfläche als $f(x) = 2.5x^2 - 3x + 8.75$ dargestellt. Sobald Sie die Konstanten ändern, aktualisiert sich das Textfeld, um die neuen Werte anzuzeigen. Im obigen Fall wird das + vor dem $\%(b)$ entfernt, weil b ein negatives Vorzeichen hat.

Relation einfügen

In diesem Dialog lässt sich eine Relation ins Koordinatensystem einfügen. „Relation“ ist ein allgemeiner Name für Ungleichungen und Gleichungen (implizite Funktion). Mit Funktion → Relation einfügen... fügen Sie eine Relation ein. Bestehende Relationen können bearbeitet werden, indem Funktion → Bearbeiten... in der *Funktionsliste* ausgewählt wird.

Relation

Hier geben Sie die Relation ein, die Sie zeichnen wollen - entweder eine Gleichung oder eine Ungleichung. x und y sind unabhängige Variablen. Eine Gleichung ist eine Aussage, daß zwei Werte gleich sind; die Werte müssen durch ein $=$ getrennt sein. Die Gleichung $x^2 + y^2 = 25$ wird z. B. einen Kreis mit dem Radius 5 zeichnen.

Eine Ungleichung ist eine Aussage, daß zwei Werte ungleich sind; die Werte müssen durch einen der vier folgenden Operatoren getrennt sein: $<$, $>$, $<=$, $>=$. Eine Ungleichung ist beispielsweise $\text{abs}(x) + \text{abs}(y) < 1$. Zwei Operatoren können einen Bereich festlegen, z. B. $y < \sin(x) < 0.5$.

Sie können dieselben Operatoren und [eingebauten Funktionen](#) benutzen wie für Funktionsgraphen. Außerdem können Sie auch [benutzerdefinierte Funktionen](#) erstellen.

Restriktionen

Hier geben Sie die Nebenbedingungen für die Relation ein. Eine Nebenbedingung ist ein beliebiger *numerischer Ausdruck*. Geplottet werden nur die Bereiche der Relation, für die alle Nebenbedingungen erfüllt sind (d.h. einen Wert ungleich Null ergeben). Nebenbedingungen bestehen normalerweise aus einer Reihe von Ungleichungen, die durch Logikoperatoren (`and`, `orxor`) verbunden werden. Für Relationen stellen x und y die unabhängigen Variablen dar.

Z. B. Bei der Relation $x^2 + y^2 < 25$ (ein schraffierter Kreis), wird durch die Nebenbedingungen $x > 0$ and $y < 0$ nur der im 4. Quadrant befindliche Teil des Kreises dargestellt.

Beschreibung:

Hier können Sie einen beschreibenden Text für die *Legende* eingeben. Wird dieses Feld leergelassen, so werden in der Legende die Relation und die Nebenbedingungen angezeigt.

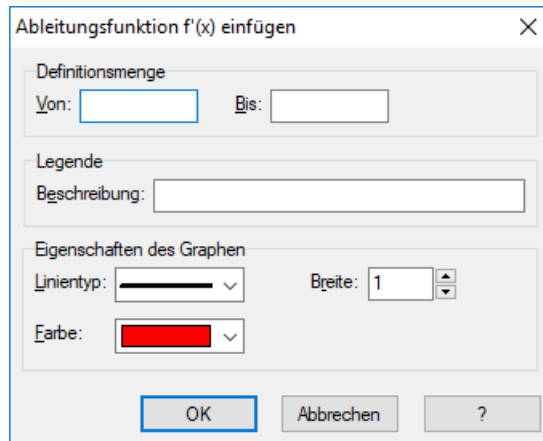
Eigenschaften

Hier können Sie Schraffurstile für Ungleichungen, Farbe und Breite für Gleichungen wählen. Die Schraffur *Stil* wird für Ungleichungen benutzt und für Gleichungen ignoriert. Man kann nur überlappende Ungleichungen sehen, wenn sie unterschiedliche Schraffurstile haben. Die *Linientyp* zeigt an die Größe der Linie für Gleichungen und die Breite der Grenzlinie für Ungleichungen. Bei Ungleichungen kann die Breite auf 0 gesetzt werden, um das Zeichnen einer Grenzlinie zu vermeiden.

Ableitungsfunktion $f'(x)$ einfügen

Der unten abgebildete Dialog dient zur Bildung der ersten Ableitung einer Funktion. Wählen Sie dazu die Funktion aus, die abgeleitet werden soll und benutzen Sie Funktion \rightarrow Ableitungsfunktion $f'(x)$ einfügen....

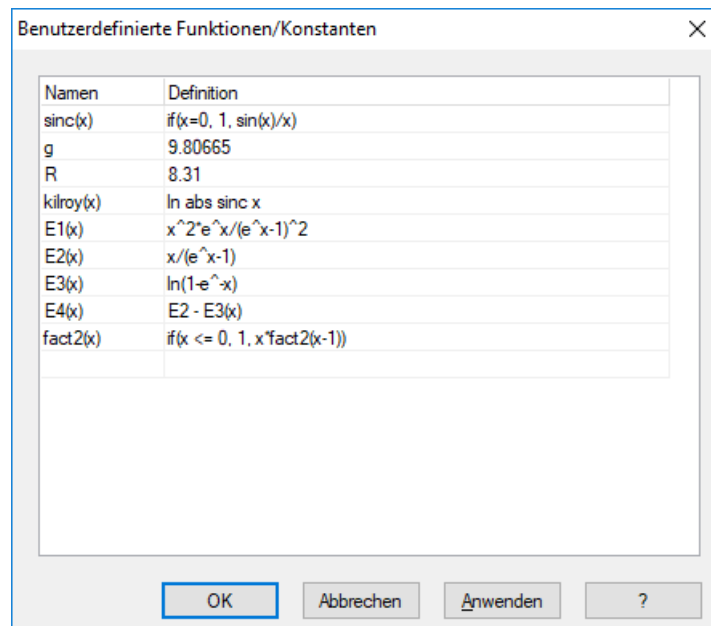
Handelt es sich um eine Standardfunktion, gibt die erste Ableitungsfunktion die Steigung der abgeleiteten Funktion an. Die Ableitungsfunktion erhält man in diesem Fall durch Differenzierung der gegebenen Funktion nach x : $f'(x) = df(x)/dx$



Sie können für die Ableitungsfunktion den Definitionsbereich sowie die Linienart, die Breite und die Farbe des Funktionsgraphen festlegen. Die Ableitungsfunktion wird der Funktionsliste hinzugefügt und kann wie gewohnt bearbeitet werden. Beachten Sie, dass die Ableitungsfunktion nicht angepasst wird, wenn die zugrunde gelegte Funktion modifiziert wird.

Benutzerdefinierte Funktionen/Konstanten

In Graph ist es möglich, benutzerdefinierte Funktionen und Konstanten zu erstellen, die dann beliebig verwendet werden können. Auf diese Weise können Konstanten oder Terme, die Sie häufig verwenden, auslagern, um Ihre Terme zu vereinfachen und leichter verändern zu können. Rufen Sie dazu den Menüpunkt Funktion → Benutzerdefinierte Funktionen... auf.



Eingabe von Funktionen

Bezeichnungen für Funktionen und Konstanten werden in der ersten Spalte eingegeben. Diese kann aus einer beliebigen Kombination von Buchstaben, Ziffern und Unterstrichen bestehen, sofern sie mit einem Buchstaben beginnt. Natürlich können Sie keine Bezeichnung vergeben, die bereits einer fest eingebauten Funktion oder Variable zugeordnet ist.

Die Argumente einer Funktion werden nach der Funktionsbezeichnung in Klammern angegeben und durch Kommata getrennt, beispielsweise bezeichnet $f(x, y, z)$ eine Funktion f , die drei Argumente namens x , y und z erwartet. Die Bezeichnungen für die Argumente müssen wie der Funktionsname mit einem Buchstaben beginnen und dürfen nur aus Buchstaben und Ziffern bestehen.

Den Ausdruck, den Sie definieren wollen, geben Sie in der zweiten Spalte ein. Der Ausdruck kann die Argumente, die in der ersten Spalte definiert wurden, sowie alle eingebauten Funktionen und übrigen

benutzerdefinierten Funktionen und Konstanten beinhalten und sich sogar selbst rekursiv aufrufen. Durch das Symbol # kann am Ende eines Ausdrucks auch ein Kommentar hinzugefügt werden.

Bearbeiten und Löschen von Funktionen

Eine Funktion oder Konstante wird entfernt, in dem ihre Bezeichnung und Definition gelöscht wird oder indem aus dem Kontextmenü **Zeile löschen** ausgewählt wird. Bedenken Sie, dass alle Elemente, die diese Funktion aufrufen oder diese Konstante verwenden, dann nicht mehr funktionieren.

Wenn Sie im abgebildeten Dialog auf **OK** oder **Anwenden** klicken, werden alle Elemente aktualisiert, indem die Änderungen an Funktionen und Konstanten übernommen werden.

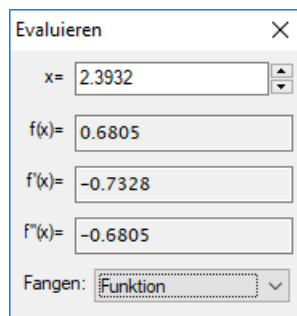
Evaluieren

Dieser Dialog dient interaktiven Berechnungen von Funktionswerten. Standardmäßig ist er unterhalb der Funktionsliste angedockt, Sie können ihn jedoch auch dort ausklinken und frei verschieben.

Evaluieren

Wenn Sie **Berechnen** → **Evaluieren** aktivieren, dann dient dieser Dialog dazu, verschiedene Funktionswerte der ausgewählten Funktion zu berechnen oder die X- und Y-Werte des Funktionsgraphen direkt mit der Maus abzufragen.

Unten ist der Dialog abgebildet, der im Fall einer Standardfunktion zu sehen ist. Wenn es sich um eine Parameterfunktion, eine Polarfunktion oder eine Tangente handelt, sieht dieses Fenster nur leicht anders aus.



Sie können einen Wert eingeben, für den der Funktionswert berechnet werden soll. Graph berechnet dann für den eingegebenen Wert den Funktionswert des in der *Funktionsliste* gerade markierten Elements. Wenn der Eingabewert und der berechnete Wert zu einem Punkt führen, der innerhalb des sichtbaren Koordinatensystems liegt, markiert Graph diesen Punkt durch ein gestricheltes Fadenkreuz. Mit der Maus kann man dieses Fadenkreuz über die dargestellte Kurve bewegen und die zugehörigen Punktwerte ablesen.

Es kann der Fall eintreten, dass eine Berechnung eine komplexe Zahl mit Imaginärteil ergibt. Ein solches Ergebnis würde entweder in der Form $a+bi$, $a\angle$ oder gar nicht ausgegeben werden, je nach unter vorgenommener Einstellung.

Wenn Sie mit der Maus auf das Koordinatensystem klicken, können Sie festlegen, woran der Mauszeiger andocken soll:

Funktion

Der Mauszeiger springt zum nächsten Punkt der ausgewählten Funktion.

Schnittpunkt

Der Mauszeiger springt zum nächsten Schnittpunkt der ausgewählten Funktion mit irgendeiner dargestellten Funktion (incl. der Funktion selber).

X-Achse

Der Mauszeiger springt zum nächsten Schnittpunkt der ausgewählten Funktion mit der X-Achse.

Y-Achse

Der Mauszeiger springt zum nächsten Schnittpunkt der ausgewählten Funktion mit der Y-Achse. Nicht für Standard-Funktionen verfügbar.

X-Extremwert

Der Mauszeiger springt zum nächsten lokalen Extremwert der X-Koordinate. Nicht für Standard-Funktionen verfügbar.

Y-Extremwert

Der Mauszeiger springt zum nächsten lokalen Extremwert der Y-Koordinate.

Fläche

Bei Auswahl von **Berechnen** → **Fläche** wird dieser Dialog dazu benutzt, um bestimmte Integrale für die gewählte Funktion über ein bestimmtes Intervall berechnen zu lassen. Bei Standardfunktionen, parametrischen Funktionen und Tangenten entspricht das bestimmte Integral der Fläche zwischen Graph und X-Achse. Dabei werden Verläufe unterhalb der X-Achse durch negatives Vorzeichen berücksichtigt. Beachten Sie, dass Graph hierfür annimmt, dass die X-Achse durch den Koordinatenursprung verläuft. Wenn Sie eingestellt haben, dass die X-Achse die Y-Achse nicht im Nullpunkt schneidet, wirkt sich diese Einstellung nicht auf die hier gemeinten Berechnungen aus.

Bei Polarfunktionen wird die Fläche zwischen Kurve und Koordinatenursprung über das angegebene Intervall berechnet. Dabei erhalten Flächenstücke ein negatives Vorzeichen, wenn die Winkel kleiner werden (Drehung im Uhrzeigersinn).

Das Intervall können Sie entweder direkt eingeben oder mit der Maus festlegen. Die berechnete Fläche wird unterhalb des Intervalls angegeben und die zugehörige Fläche im Koordinatensystem markiert. Zur internen Berechnung nutzt Graph die Gauß-Kronrod-Quadratur, um eine höchstmögliche Genauigkeit zu erreichen. Sollte dennoch ein Schätzfehler von 10^{-4} überschritten wird, wird kein Ergebnis angezeigt.

Länge eines Kurvenstücks

Berechnen → **Länge eines Kurvenstücks** wird benutzt, um die Länge eines Kurvenstücks zwischen zwei Punkten zu berechnen. Die Punkte können entweder direkt eingegeben oder mit der Maus markiert werden. In beiden Fällen markiert Graph dieses Intervall im Koordinatensystem. Die Berechnung liegt die Simpson'sche Regel (1.000 Iterationen) zugrunde.

Tabelle

Der untenstehende Dialog wird benutzt, um die gewählte Funktion für einen Bereich zu berechnen. Wählen Sie eine Funktion aus der *Funktionsliste* und lassen Sie sich den Dialog mit dem Menüpunkt **Berechnen** → **Tabelle** anzeigen. Legen Sie für die unabhängige Variable den ersten Wert (*Von*), den letzten Wert (*Bis*) und die Schrittweite (Δx bzw. Δt) für jede neue Berechnung fest.

Wenn Sie die Schaltfläche **Berechnen** drücken, wird die Tabelle mit der unabhängigen Variablen in der ersten Spalte gefüllt. Die restlichen Spalten hängen vom Funktionstyp ab. Bei einer Standardfunktion zeigt die Tabelle $f(x)$, $f'(x)$ und $f''(x)$, bei einer Parameterfunktion $x(t)$, $y(t)$, dx/dt , dy/dt und dy/dx und bei einer Polarfunktion $r(t)$, $x(t)$, $y(t)$, dr/dt und dy/dx . Nicht benötigte Spalten lassen sich im Kontextmenü ausblenden. Bei länger dauernden Berechnungen wird ein Fortschrittsbalken angezeigt.

Wertetabelle

Von: Bis:

$\Delta x =$

x	f(x)	f'(x)	f''(x)
-10.0	-7.2799	-3.1954	-2.7201
-9.9	-7.6123	-3.446	-2.2877
-9.8	-7.9676	-3.6521	-1.8324
-9.7	-8.3412	-3.8118	-1.3588
-9.6	-8.7284	-3.9234	-0.8716
-9.5	-9.1242	-3.9859	-0.3758
-9.4	-9.5239	-3.9985	0.1239
-9.3	-9.9223	-3.9611	0.6223
-9.2	-10.3144	-3.8742	1.1144
-9.1	-10.6955	-3.7386	1.5955
-9.0	-11.0606	-3.5557	2.0606
-8.9	-11.4051	-3.3272	2.5051
-8.8	-11.7246	-3.0555	2.9246
-8.7	-12.0148	-2.7432	3.3148
-8.6	-12.272	-2.3936	3.672
-8.5	-12.4924	-2.0101	3.9924
-8.4	-12.673	-1.5964	4.273
-8.3	-12.8109	-1.1569	4.5109

Sie können einige Zellen mit der Maus selektieren, machen einen Rechtsklick und kopieren die Zellen mit Kopieren im Kontextmenü in die Zwischenablage. Die Daten können aus der Zwischenablage in ein anderes Programm eingefügt werden, z. B. Microsoft Excel.

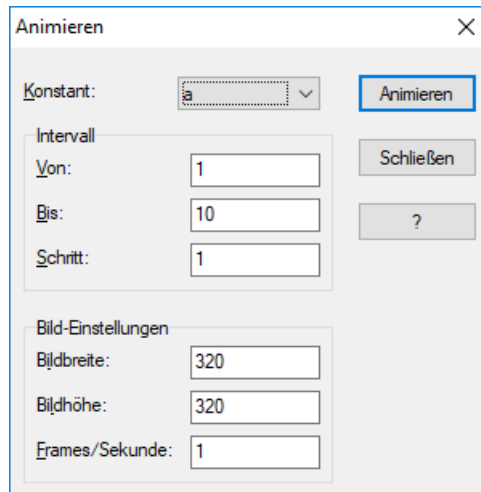
Auf der linken Seite der Tabellen verwandelt sich der Mauszeiger in einen Rechts-Pfeil. Jetzt lassen sich mit der Maus ganze Reihen auswählen. Im oberen Tabellenbereich verwandelt sich der Mauszeiger in einen Nach-Unten-Pfeil. Jetzt lassen sich mit der Maus ganze Spalten auswählen. Die komplette Tabelle läßt sich mit Rechts-Klick und **Alle wählen** auswählen. Einzelne Zellen lassen sich ebenfalls auswählen, indem bei gedrückter **Shift**-Taste die Pfeiltasten benutzt werden.

Im Kontextmenü lassen sich die ausgewählten Daten auch als komma- oder tabulator-separierter Text in eine Datei exportieren.

Beachten Sie, daß bei vielen Daten in der Tabelle die Berechnung einige Zeit dauern kann. Viele Werte können auch viel Systempeicher in Anspruch nehmen.

Animieren

Dieser Dialog dient dazu, aus einer Schar von Funktionsgraphen eine Animation zu erstellen. Diese kann dann in Graph abgespielt, als Datei exportiert oder in ein Dokument eingefügt werden. Eine Animation kann alle Elemente beinhalten, die in Graph zur Verfügung stehen, d.h. beispielsweise Funktionen, Relationen, Punktserien, Beschriftungen usw.



Konstante

Hier legen Sie fest, mit Hilfe welches Parameters die Animation gesteuert werden soll. Dieser Parameter muss allerdings zuerst im Dialog [Benutzerdefinierte Funktionen/Konstanten](#) erstellt werden. Für jeden Animationsschritt verändert Graph den Wert dieses Parameters.


Intervall

Mit Hilfe von *Von* und *Bis* begrenzen Sie das Intervall an Werten, das dem Parameter für die Animation übergeben wird. Der Wert *Schritt* legt fest, in welchen Schritten der Parameterwert von Animationsschritt zu Animationsschritt verändert wird. Die Anzahl an Animationsschritten ergibt sich also durch den Quotienten $(Bis - Von) / Schritt$. Läuft die Animation mit mehr Zwischenschritten ab, ergeben sich weichere Übergänge, allerdings benötigt Graph auch länger für die Berechnung und die Animationsdatei benötigt beim Speichern mehr Speicherplatz.

Bildinformationen

Sie können die Bildgröße für die Animation festlegen. Beachten Sie, dass diese Einstellung sowohl auf die Dateigröße sowie auf die zur Erstellung der Animation notwendige Zeit Einfluss hat. *Frames/Sekunde* gibt die Geschwindigkeit an, in der die Animation standardmäßig abgespielt wird. Die meisten Media Player sollten in der Lage sein, die Animationsgeschwindigkeit regeln zu können.

Durch Anklicken der Schaltfläche **Animieren** wird auf Basis der gewählten Einstellungen eine Animation erstellt. Dieser Prozess dauert einen Moment, wobei die notwendige Zeit davon abhängt, was sich im Koordinatensystem befindet und wie viele Bilder als Animationsschritte berechnet werden müssen.

Nach der Animation wird ein einfacher Player geöffnet, mit dem Sie die Animation abspielen können. Die Schaltfläche  bietet Ihnen einige zusätzliche Optionen.

Geschwindigkeit

Hier läßt sich die Wiedergabegeschwindigkeit einstellen. Die gespeicherte Datei wird dabei nicht verändert.

Wiederholen

Aktiviert die Darstellung der Animation als Endlosschleife.

Auto-Reverse

Hiermit läuft die Animation rückwärts, wenn zu Ende ist. Dies ist sehr nützlich in Kombination mit der Option **Wiederholen**, mit der man die Animation zwischen beiden Enden oszillieren lassen kann.

Speichern unter...

Die Animation wird als AVI-Datei (Audio Video Interleave) gespeichert, die jeder Media-Player abspielen kann.

Bild speichern unter...

Dies speichert das aktuelle Einzelbild in eine Datei, entweder als Windows Bitmap (bmp), Portable Network Graphics (png) oder Joint Photographic Experts Group (jpeg).

Alle Bilder speichern...

Dies speichert alle Einzelbilder als separate Bilddateien. Dasselbe wird erreicht durch wiederholtes Bild speichern unter... für jedes Einzelbild der Animation.

Als Bilddatei abspeichern

Mit dem Menüpunkt *Datei* → *Als Bilddatei abspeichern...* können Sie das dargestellte Koordinatensystem als Bild-Datei speichern. Nachdem ein Menüpunkt gewählt wurde, erscheint ein Standard- *Speichern als* Dialog, in dem Sie einen Dateinamen mit Ordner auswählen und einen der folgenden Dateitypen für das Bild festlegen:

Windows Enhanced Metafile (auch: Erweiterte Metadatei, emf)

Metadateien haben die Vorteile, klein zu sein und auch in verschiedenen Bildgrößen optimal auszusehen. Allerdings sind EMF-Dateien nur unter Windows meist problemlos einsetzbar.

Scalable Vector Graphics (svg)

Dieses Dateiformat ist für portable Metadateien gedacht und sollte für die Veröffentlichung von Dateien im Internet benutzt werden. Dennoch unterstützen noch immer nicht alle Browser dieses Format.

Portable Network Graphics (png)

Bilder im png-Format (Portable Network Graphics) sind besser komprimiert als bmp-Dateien. Außerdem ist dieses Format für Web-Seiten ideal, nicht nur weil die Dateien klein sind, sondern auch weil es alle Browser unterstützen.

Windows Bitmap (bmp)

Windows Bitmap (bmp) ist ein Standardformat für Pixelgrafiken, das nahezu alle Windows-Programme beherrschen, die Grafiken importieren können.

Joint Photographic Experts Group (jpeg)

Joint Photographic Experts Group (jpeg) ist ein verlustbehaftetes Bitmap-Format. Es wird unterstützt, aber nicht empfohlen, weil die Grafiken üblicherweise unscharf werden.

Portable Document Format (pdf)

Portable Document Format (pdf) ist genau genommen kein Bild-Format. Es ist ein Weg, Dokumente portabel als Postscript zu speichern. Graph speichert das Bild als Portable Network Grafik innerhalb der PDF-Datei.

Über die Schaltfläche *Einstellungen...* im Dialog "Als Bilddatei abspeichern" können Bildgröße und weitere Parameter des jeweiligen Dateiformats festgelegt werden.

Punktserie importieren

Die Daten für Punktserien kann Graph aus Textdateien importieren. Die Daten müssen entweder durch Komma, Semikolon, Leerzeichen oder Tabulator getrennt sein. Benutzen Sie *Datei* → *Importieren* → *Punktserie(n)...* und wählen die zu importierende Datei. Ist der genaue Dateityp nicht angegeben, so versucht Graph, ihn automatisch zu bestimmen.

Die einfachste Datei mit Komma als Trennzeichen sieht so aus:

```
1,1.7
2,4.3
3,9.5
4,16.2
```

. Wird diese Datei importiert, erhält man eine Punktserie mit den folgenden Koordinaten: (1,1.7), (2,4.3), (3,9.5), (4,16.2).

Enthält eine Datei mehr als zwei Spalten, importiert Graph diese als mehrere Punktserien, wobei in der ersten Spalte jeweils als x-Koordinate interpretiert wird. Ein Beispiel mit Semikolon als Trennzeichen:

```
50;71,8113997;78,23883162;79,17509098;78,58979676;78,62449077;78,2374541;77,7637696
```

```
55;71,81941659;78,17869416;79,2155277;78,48195769;78,6005272;78,22113423;77,7874084
60;71,82943769;78,12714777;79,24787707;78,55661551;78,51266076;78,2129743;77,8425656
65;71,82866684;78,08419244;79,20744036;78,54002489;78,56857576;78,22929417;77,8189268
70;71,81448312;78,06701031;79,18317833;78,55661551;78,55260005;78,24561404;77,75589
```

. Wird diese Datei importiert, erhält man 7 Punktserien mit denselben x-Koordinaten.

Sie können Punktserien benennen, indem sie die erste Zeile der Datei mit einer Raute (#) beginnen und dahinter die Bezeichnung schreiben. Mehrere Bezeichnungen müssen dasselbe Trennzeichen verwenden wie die Datenzeilen. Auf diese Weise können auch mehrere Punktserien erstellt werden, die unterschiedliche X-Koordinaten haben. Diese werden in der Kopfzeile durch eine weitere Raute (#) eingeleitet, hinter der eine Bezeichnung stehen kann. Im Folgenden ist ein Beispiel zu sehen, in dem Leerzeichen als Trennzeichen verwendet werden:

```
#a b
0 0.2 0.5
1 1.4 -1.7
2 2.1 -2.6
3 3.8 -3.3
#c d
4 4.3 -4.1
5 5.8 -5.5
6 6.1 -6.1
7 7.9 -7.6
```

. Im Beispiel ergeben sich vier Punktserien a, b, c und d mit jeweils vier Datenpunkten. Punktserie a und b verwenden dieselben x-Koordinatenwerte, ebenso Punktserie c und d.

Importiert werden können nicht nur Zahlenwerte, sondern beliebige gültige Ausdrücke, wie das folgende Beispiel anhand von Semikolon-getrennten Werten zeigt:

```
1;sin(0.1)
2;sin(0.2)
3;1+sin(0.3)
pi;2*sin(pi/2)
```

Plugins

Graph verwendet Python 3.7, der mit Graph installiert wird, als Scripting-Engine. Wenn Sie jedoch Python 3.7 installiert haben, verwendet Graph diese Installation anstelle des Bundles, das mit Graph geliefert wird. Eine Dokumentation der Python-Sprache finden Sie online unter <http://docs.python.org/3.7/>.

Plugins

Plugins sind Python-Skripts, die normalerweise als .py-Dateien direkt im Quellcode als Textdatei vorliegen. Vorkompilierte Python-Dateien (.pyc-Dateien) sind jedoch auch möglich. Die Plugin-Dateien werden entweder im Verzeichnis `Plugins` abgelegt, in dem auch Graph installiert ist, oder im lokalen Datenverzeichnis. Beim Programmstart findet und lädt Graph sie automatisch. Die Dokumentation der API, welche Plugins zur Interaktion mit Graph verwenden können, finden Sie in der Dokumentation der Scripting-Engine.



Warnung

Plugins sind Skripte, d. h. kleine Programme, die von Graph selbst aufgerufen werden und die mit Graph interagieren. Plugins können alle Aktionen ausführen, die beliebige andere Programme ebenfalls ausführen könnten. Plugins können damit ein schweres Sicherheitsrisiko darstellen, wenn Sie Graph mit Administrator-Rechten ausführen. So wäre es bspw. möglich, ein Plugin mit Schadroutinen auszustatten und damit etwa die Festplatte zu löschen! Verwenden Sie also die Plugin-Schnittstelle mit Vorsicht und installieren Sie nur Plugins von Personen bzw. Webseiten, denen Sie absolut vertrauen oder überprüfen Sie den Quellcode selbst auf unerwünschten Code.

Plugin-Manager

Graph wird mit einem Plugin Manager geliefert, der eigentlich selbst nur ein Plugin ist. Dies macht es einfach, Plugins von Drittanbietern zu installieren und zu entfernen. Der Plugin-Manager kann .py- und .pyc-Dateien direkt und .zip-Dateien (die Python-Pakete sind), installieren, d.h. er muss eine `__init__.py`- oder `__init__.pyc`-Datei im Rootverzeichnis der .zip-Datei enthalten.

Python Interpreter

```
Python Interpreter
>>> F=Graph.StdFunc("sin(x)+x")
Traceback (most recent call last):
  File "<console>", line 1, in <module>
AttributeError: module 'Graph' has no attribute 'StdFunc'
>>> F=Graph.TStdFunc("sin(x)+x")
>>> Graph.FunctionList.append(F)
>>> |
```

Die Plugin-Schnittstelle stellt auch einen direkten Zugang zum Python-Interpreter zur Verfügung, wenn **F11** gedrückt wird. Diesem Interpreter können Sie direkt Python-Ausdrücke zur Verarbeitung geben, was natürlich sehr fortgeschrittene Arbeit mit Graph gestattet. Auf diese Weise können Sie sehr leicht Code testen, bevor Sie daraus ein Plugin bauen.

Danksagungen

Bibliotheken

dxgettext

Übersetzungsbibliothek.

Copyright © Lars B. Dybdahl et al.

<http://dybdahl.dk/dxgettext/>

PDFlib-Lite

Wird verwendet, um PDF-Dateien zu erzeugen.

Copyright © 1997-2005 Thomas Merz & PDFlib GmbH

<http://www.pdflib.com>

Python

Für die Plugin-Schnittstelle erforderlich.

Copyright © 2001-2006 Python Software Foundation

<http://www.python.org>

GNU Scientific Library

Numerische Bibliothek.

Copyright © 2009 Free Software Foundation, Inc.

<http://www.gnu.org/software/gsl/>

Boost

Begutachtete (peer reviewed) C++ Bibliothek

<http://www.boost.org>

Übersetzungen

Sprache	Programm	Hilfedatei	Übersetzer
Arabisch	Ja	Nein	Abdellah Chelli راطيبلا رساي
Baskisch	Ja	Nein	Xabier Maiza
Chinesisch (vereinfacht)	Ja	Nein	Loquat Lala Sha
Chinesisch (traditionell)	Ja	Nein	Jian-Jie Dong
Kroatisch	Ja	Ja	Hasan Osmanagić
Tschechisch	Ja	Ja	Pavel Simerka Martin Stružský Pavčina Krausová
Dänisch	Ja	Ja	Michael Bach Ipsen Erik Lyngholt Nielsen
Niederländisch	Ja	Ja	Etienne Goemaere
Englisch	Ja	Ja	Ivan Johansen
Finnisch	Ja	Nein	Pekka Lerssi
Französisch	Ja	Ja	Jean-Pierre Fontaine
Deutsch	Ja	Ja	Frank Hüttemeister Sebastian Stütz Michael Bach Ipsen
Griechisch	Ja	Ja	Dimitris Spingos (Δημήτρης Σπίγγος) Theodoros Kannas
Ungarisch	Ja	Nein	Róbert Gáti Gabor Magyar
Italienisch	Ja	Ja	Alessandro Serena Attilio Ridomi
Koreanisch	Ja	Nein	Choe Hyeon-gyu
Mongolisch	Ja	Nein	Batnasan Davaa
Norwegisch	Ja	Nein	Tore Ottinsen
Persisch	Ja	Nein	Shayan Abyari Yashar PourMohammad
Polnisch	Ja	Nein	Paweł Baczyński Dominik Barwacz
Portugiesisch (Brasilien)	Ja	Ja	Jorge Costa Mara Fernanda Deivid e Monalisa Andre Fduarte Haroldo Luiz Bertoldo Janete Flor de Maio Fonseca Wladimir A. Silva Aldemar C. Filho

Sprache	Programm	Hilfdatei	Übersetzer
Portugiesisch (Portugal)	Ja	Nein	Jorge Geraledes
Russisch	Ja	Nein	Ilya A. Ivans Leonovs
Serbisch	Ja	Nein	Jasmina Malinovic Branimir Krstic
Slovenisch	Ja	Ja	Jernej Baša Rok Štokelj Barbara Pušnar Sergej Pušnar
Spanisch	Ja	Ja	Francisco Oliver Alejandro Arce
Schwedisch	Ja	Nein	Pär Smårs Michael Bach Ipsen
Türkisch	Ja	Nein	Mumtaz Murat Arik
Ukrainian	Ja	Nein	Mumtaz Murat Arik
Vietnamesisch	Ja	Nein	Dmytro Amelchev

Spender

Sebastian Albrecht, specializing in the [Vancouver Special](http://yourvancouverrealestate.ca/vancouver-special/) [http://yourvancouverrealestate.ca/vancouver-special/]

[Sothebys real estate Toronto](http://partnershiptoronto.com/) [http://partnershiptoronto.com/]

Chantal Marr, a [Canadian life insurance](http://lsminsurance.ca/canadian-life-insurance-companies) [http://lsminsurance.ca/canadian-life-insurance-companies] expert

Elli Davis, a [Toronto MLS Listings](http://ellidavis.com/mls-listings) [http://ellidavis.com/mls-listings] expert

Verschiedene

Das Graph-Symbol wurde von Jonathan Holvey erstellt.

Glossar

Bogenmaß

Das Bogenmaß ist ein dimensionloses Winkelmaß, d.h. eine Größenangabe für einen Winkel, die ohne Gradzahl auskommt. Ein 360° umfassender Vollwinkel hat im Bogenmaß die Größe 2π , d.h. zwischen Bogenmaß und Grad kann eine Umrechnung vorgenommen werden, indem der im Bogenmaß angegebene Winkel mit $180^\circ/\pi$ bzw. der in Grad angegebene Winkel mit $\pi/180^\circ$ multipliziert wird. Im Register [Einstellungen](#) des Dialogs *Achsen bearbeiten* können Sie einstellen, ob Graph mit Winkelangaben in Grad oder im Bogenmaß arbeiten soll.

Funktionsliste

Die Funktionsliste befindet sich auf der linken Seite des Hauptfensters und beinhaltet eine Liste aller Funktionen, Tangenten, Punktserien, Schraffuren und Relationen. Wenn Sie ein Element aus dieser Liste bearbeiten wollen, müssen Sie es zuerst auswählen, d.h. anklicken. Das Element wird dann blau hinterlegt dargestellt, es sei denn, die Funktionsliste befindet sich gerade nicht im Vordergrund. In einem solchen Fall wird es grau hinterlegt. Ein Element kann mit Hilfe des Menüs *Funktion* oder des Kontextmenüs, das Sie durch einen Rechtsklick auf dieses Element erhalten, bearbeitet werden.

Ganzzahl

Die Zahlen $\dots, -3, -2, -1, 0, 1, 2, 3, \dots$ bilden die Menge der ganzen Zahlen (auch: Integerzahlen). Dabei handelt es sich um eine Untermenge der reellen Zahlen. Eine Integerzahl n kann also null oder ganzzahlig negativ oder positiv sein.

Graph-Element

Unter "Element" ist ein Objekt zu verstehen, das im Koordinatensystem von Graph platziert werden kann. Dabei kann es sich um eine Funktion, eine Punktserie, eine Beschriftung, Relation, etc. handeln. Solche Elemente werden ebenfalls in der Funktionsliste aufgeführt und können mit Hilfe des Menüs *Funktion* oder des Kontextmenüs bearbeitet werden.

komplexe Zahl

Die Menge der komplexen Zahlen stellt eine Obermenge der Menge der reellen Zahlen dar. Bei komplexen Zahlen handelt es sich um "zweidimensionale" reelle Zahlen, die meist mit Hilfe der gaußschen Zahlenebene in der Form $a+bi$ mit a als Real- und b als Imaginärteil veranschaulicht werden. Die imaginäre Einheit i ist definiert als $i^2=-1$. Neben dieser Darstellung können komplexe Zahlen auch mit Hilfe einer Polarkoordinatendarstellung wie folgt beschrieben werden: $a\angle\theta$ mit a als Betrag der darzustellenden Zahl und θ als Winkel (in Grad oder im Bogenmaß), den die komplexe Zahl mit der reellen Zahlenachse einschließt.

Komplexe Zahlen können im Dialog *Evaluiere* zum Zeichnen von Standardfunktionen Verwendung finden, sofern im Dialog *Achsen bearbeiten* im Register [Einstellungen](#) die Option *Komplex berechnen* aktiviert ist.

Legende

Die Legende ist ein Kasten, der sich standardmäßig im rechten oberen Bereich des Koordinatensystems befindet und eine Liste aller darin dargestellten Funktionsgraphen, Tangenten, Schraffuren und Punktserien beinhaltet. Um die Legende einzublenden, wählen Sie im Dialog *Achsen bearbeiten* unter [Einstellungen](#) die Option *Legende anzeigen*. Wenn ein Element der Funktionsliste nicht in der Legende aufgeführt werden soll, dann führen Sie einen Rechtsklick auf dem entsprechenden Eintrag der Funktionsliste auf und deaktivieren Sie den Punkt *In der Legende anzeigen*. Bei der Bearbeitung eines Elements der Funktionsliste können Sie festlegen, welcher Text dieses in der Legende beschreiben soll. Bei Funktionen und Tangenten wird standardmäßig die Funktionsvorschrift eingetragen, falls kein gesonderter Legendentext angegeben wird.

numerischer Ausdruck

Ein Ausdruck, dessen Wert berechenbar ist, wird als numerischer Ausdruck bezeichnet. Ein solcher Term kann aus einer beliebigen Verknüpfung von Zahlen, Konstanten, Variablen, Operatoren und Funktionen zusammengesetzt werden.

reelle Zahl

Eine reelle Zahl hat allgemein die Form $nnn.fffEeee$, wobei nnn den ganzzahligen Teil der Zahl darstellt, welcher auch negativ werden darf. fff stellt die Nachkommastellen dar, die - entgegen der in Deutschland üblichen Konvention - vom ganzzahligen Teil mit einem Punkt ('.') getrennt werden. Der Nachkommateil kann auch weggelassen werden, wenn es sich um eine ganze Zahl handelt. Umgekehrt kann der Teil vor dem Dezimaltrenner weggelassen werden, wenn es sich um eine Dezimalzahl zwischen null und eins handelt. Das 'E' ist ein Trennzeichen, das gesetzt werden kann, falls Dezimalzahlen in Gleitkommaform dargestellt werden sollen. In diesem Fall gibt die Dezimalzahl vor dem 'E' alle Ziffern der darzustellenden Zahl an (Mantisse), die mit der eee . Potenz von 10 multipliziert eben diese Zahl ergibt. Natürlich kann eee auch ein negatives Vorzeichen haben. Beispielsweise lässt sich $5 \cdot 10^8$ also verkürzt als $5E8$ darstellen.