Graph

Version 4.4

Translator:
Michael Bach Ipsen (rosenthall@hotmail.com)
Sebastian Stütz (der.stirz@gmx.net)
Frank Hüttemeister (frank.huettemeister@padowan.dk)

Inhaltsverzeichnis

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 is free software; you can redistribute it and/or modify it under the terms of the *GNU General Public License* [http://www.gnu.org/licenses/gpl.html]. Newest version of the program as well as the source code may be downloaded from http://www.padowan.dk.

Graph has been tested under Microsoft Windows 2000, Windows XP, Windows Vista, und Windows 7, but there may still be bugs left. If you need help using Graph or have suggestions for future improvements, please use the Graph support forum [http://www.padowan.dk/forum].

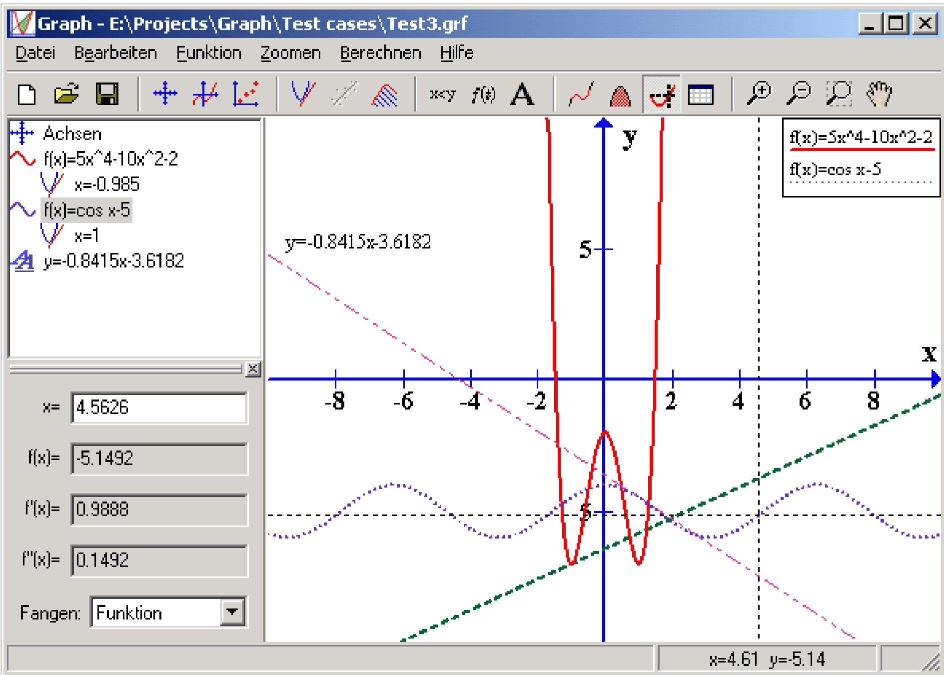
When Sie eine Fehlermeldung senden, beachten Sie bitte die folgenden Aspekte:

- 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 bespielsweise 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

Graph is usually distributed as an installation program named SetupGraph-x.y.exe, where x.y is the version number. To install, just execute the file and follow the instructions. The installation will install the following files in the selected directory and subdirectories:

Datei(en)	Beschreibung:
`Graph.exe`	Die Programm-Datei.
`PDFlib.dll`	Library used to create PDF files.
`Thumbnails.dll`	Shell extension für die Vorschaubild-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`	Library files used by plugins.
`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 angegebenenen Parameter übernehmen. Diese sind besonders nützlich, wenn Sie die Installation automatisieren wollen.

Parameter	Beschreibung:
/SILENT	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 *Jetzt Neustart?* angezeigt.
/VERYSILENT	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.
/NORESTART	Das Setup wird nicht neu starten, auch wenn es notwendig ist.
/LANG=language	Legt die Sprache fest. *language* gibt das englische Wort für die Sprache an. Bei einem gültigen Parameter /LANG wird der *Sprache wählen*-Dialog unterdrückt.
/DIR=x:\dirname	Überschreibt den voreingestellten Verzeichnisnamen, der im Assistenten unter *Wählen Sie den Zielort* angezeigt wird. Hier muß ein geeigneter Pfadname angegeben werden.

Deinstallation

Die Deinstallation wird von *Programme ändern oder entfernen* in der *Systemsteuerung* durchgeführt. Wählen Sie Graph und klicken auf *Entfernen*. Damit werden alle Spuren des Programms entfernt. Falls im Installationsverzeichnis nach der Installation Dateien hinzugefügt wurden, werden Sie gefragt, ob Sie diese löschen möchten. Vergewissern Sie sich, 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:
/SI=file	Wird benutzt, um eine göffnete .grf-Datei als Bild zu speichern. Der Dateityp kann eines der von Graph unterstützten Bildformate sein.
/WIDTH=width	Wird in Kombination mit /SI benutzt, um die Breite des zu speichernden Bildes in Pixeln anzugeben.
/HEIGHT=height	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 requires Microsoft Windows 2000 or newer. It has been tested under Windows 2000, Windows XP, Windows Vista, und Windows 7.

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: Analoge Antwort wie zuvor: Graph läuft nicht direkt auf Ihrem Mac, wenn Sie allerdings einen Windows-Emulator benutzen oder Windows in einer virtuellen Maschine ausführen, sollte auch Graph darin laufen.

F: Wann kommt die nächste Version?

A: Wenn es feritig 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 Mausrad 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: A vertical line can be drawn as a parametric function. Select *Parameterfunktion* as *Funktionstyp* when adding the function. You can then add the vertical line at x=5 as **`x(t)=5, y(t)=t`**. Alternatively you can add **`x=5`** as a relation.

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 *Funktionstyp* unter *Parameterfunktion*, wenn Sie die Funktion hinzufügen. Wollen Sie den Graphen der Funktion x=sin(y) zeichnen lassen, wäre die entsprechende parametrische Form **`x(t)=sin(t), y(t)=t`**. Eine andere Möglichkeit besteht darin die Relation **`x=sin(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 *Funktionstyp Parameterfunktion*. Um Kreis mit dem Punkt (2,3) als Mittelpunkt und mit dem Radius 5 zeichnen zu lassen, könnten Sie beispielsweise diese Parameterfunktion eingeben: **`x(t)=5cos(t)+2, y(t)=5sin(t)+3`**. Damit der Kreis auch wie ein Kreis aussieht (und nicht wie eine Ellipse), müssen Sie möglicheweise 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 **`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: **`(x-2)^2+(y-3)^2=5^2`**.

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 f1(x)=3x und f2(x)=x^2, dann fügen Sie zuerst die Differenzfunktion f(x)=f1(x)-f2(x)=3x-x^2 der Funktionsliste hinzu. Anschließend können Sie mit Berechnen → Fläche die Fläche zwischen den beiden Funktionsgraphen für ein bestimmtes Intervall berechnen lassen.

F: Kann ich eine Funktion oder einer Punktserie von einem laufenden Graph-Fenster zu einem anderen Graph-Fenster kopieren?

A: Ja, das ist möglich. Wählen Sie eine Funktion oder Punktserie und wählen Sie den Menüpunkt Bearbeiten → Kopieren, um das ausgewähltte Objekt in die Zwischenablage zu kopieren. Anschließend können Sie es in ein anderes Koordinatensystem einfügen.

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

A: Bei Funktionen wird für jedes x des Definitionsbereichs höchstens genau ein Wert f(x) berechnet, so dass bei f(x)=sqrt(x+2) die nicht-negativen Funktionswerte zu einem Funktionsgraphen im positiven Quadranten führen. Wenn dieser Graph in den negativen Bereich gespiegelt werden soll, müssen Sie eine weitere Funktion f(x)=-sqrt(x+2) anlegen. Denselben Graph erhält man durch die Eingabe der Relation y^2=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 x(t)=re(e^(i*t)), y(t)=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 Bild speichern ist es möglich, das Koordinatensystem als PDF-Datei abzuspeichern.

F: Funktioniert Graph auch unter Windows 95?

A: Aktuelle Versionen von Graph laufen unter Windows 95 nicht mehr. Die letzte Windows 95-kompatible Version war Graph 4.2.

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 Bearbeiten → 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 Bearbeiten → Spezielles Einfügen... verwenden.

Sie können eine neues Graph-Objekt in Word erzeugen, indem Sie bei Einfügen → Objekt... als *Objekttyp Graph-System* 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 ist 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).Speichert das Koordinatensystem als Bilddatei (*.emf, *.bmp oder *.png).Speichert das Koordinatensystem als Bilddatei (*.emf, *.bmp oder *.png).Speichert das Koordinatensystem als Bilddatei (*.emf, *.bmp oder *.png).Speichert das Koordinatensystem als Bilddatei (*.emf, *.bmp oder *.png).Speichert das Koordinatensystem als Bilddatei (*.emf, *.bmp oder *.png).Speichert das Koordinatensystem als Bilddatei (*.emf, *.bmp oder *.png).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 soviele 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**)
: Beendent 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, Plazierung der Legende usw.

Bearbeiten → Einstellungen...

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

Funktion → Funktion einfügen... (**Einfg**)

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

Funktion → Tangente einfügen... (**F2**)

Erzeugt in einer bestehenden Funktion eine Tangente an einem vom Benutzer angegebenen Punkt. Die Tangente wird zur in *Funktionsliste* gewählten Funktion hinzugefügt.

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 ¼ der vorigen Zeichenfläche sehen.

Zoomen → Verkleinern (**Strg+-**)

Hiermit zommen Sie heraus, so daß die nun sichtbare Zeichenfläche 4 mal so groß wie vorher ist.

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-Achses. 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 umherschieben. 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 umherschieben.

Zoomen → Passend

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

Zoomen → Alles

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 → Fläche

Berechnet die vorzeichenbehaftete Fläche zwischen dem Graph und der X-Achse. Dies entspricht dem numerischen Integral.

Berechnen → Evaluieren (**Strg+E**)

Die ausgewählte Funktion wird an einem gegebenen Punkt untersucht. Bei Standard-Funktionen werden f(x), f'(x) und f''(x) ausgwertet. 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.

? → 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.

? → Tipp des Tages

Hier werden einige Tips über die optimale Nutzung und bestimmte Leistungsmerkmale von Graph angezeigt, die Sie vielleicht noch nicht kennen.

? → Internet → Graph Web-Seite

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 Verion besuchen wollen.

? → About Graph (**Alt+F1**)

Zeigt Versionsnummer, Copyright und Lizenzinformationen für Graph an.

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° + p180° ist nicht definiert.

Der Tangens tan(x) ist für x= π/2+πp = 90°+p180°, 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#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 asin(x) und Arcuscosinus acos(x) sind nur auf dem Intervall [-1;1] definiert. Ferner sind sie für beliebige Zahlen mit Imaginärteil nicht definiert. Die Funktion Arcustangens atan(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: sign(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 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 acosh(x) ist für *reelle Zahlen* nur für x≥1 definiert. 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: Zuviele rekursive Aufrufe

Es wurden zuviele rekursive Aufrufe ausgeführt. Der Fehler wird meist von einer Funktion erzeugt, die sich selbst rekursiv unendlich oft aufruft; z. B. (x)=2*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 18: A plugin function failed.

A custom function in a Python plugin did not return a result. The plugin console may show more detailed information.

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 Endeklammern 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 Konstant 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: Funktion 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), conj(x), re(x) und 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.

Error 88: No result found.
No valid result exist. This may for example happen when trying to create a trendline from a point series where it is not possible to calculate a trendline. One reason can be that one of the calculated constants needs to be infinite.

Error 89: An accurate result cannot be found.
Graph could not calculate an accurate result. This may happen when calculating the numeric integral produced a result with a too high estimated error.

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 zusammengesellt, 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
`pi`	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 $\mathbf{i}^2 = -1$ und wird nur dann benötigt, wenn mit komplexen Zahlen gearbeitet wird.
`inf`	The constant for infinity. Only useful as arguments to the `integrate` function.
rand	Erzeugt eine Zufallszahl zwischen 0 und 1.

Operator	Beschreibung:
Exponentation (^)	Exponentialfunktion. Beispiel: f(x)=2^x
Negation (-)	Negativer Wert eines Faktors. Beispiel: f(x)=-x
Logisch NOT (not)	Evaluates to 1 if an expression evaluates to 0, and evaluates to 0 otherwise.
Multiplikation (*)	Multiplies two factors. Example: f(x)=2*x
Division (/)	Divides two factors. Example: f(x)=2/x
Addition (+)	Adds two terms. Example: f(x)=2+x
Subtraktion (-)	Subtracts two terms. Example 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)	Evaluates to 1 if two expressions both evaluate to a value different from 0, and evaluates to 0 otherwise.
Logisched OR (oder)	Evaluates to 1 if at least one of two expressions evaluates to a value different from 0, and evaluates to 0 otherwise.
Logisches XOR (exclusiv oder)	Evaluates to 1 if exactly one of two expressions evaluates to a value different from 0, and evaluates to 0 otherwise.

Funktion	Beschreibung:
	Trigonometrisch
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.
	Hyperbolicus
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.
	Potenz und Logarithmus
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.
	Komplex
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:
Runden	
trunc	Liefert den ganzzahligen Anteil des Arguments zurück.
fract	Liefert den gebrochenen Anteil des Argumentes.
ceil	Rundet das Argument zur nächsten Ganzzahl auf.
floor	Rundet das Argument zur nächsten Ganzzahl ab.
round	Rundet das erste Argument auf die Anzahl der Dezimalstellen aus dem zweiten Argument.
Stückweise	
sign	Liefert das Vorzeichen des Arguments zurück: 1, wenn das Argument gößer als 0 ist und -1, wenn es kleiner als 0 ist.
u	Einheitssprung: Liefert 1 zurück, wenn das Argument gößer gleich als 0 ist, sonst 0.
min	Liefert das kleinste Argument zurück.
max	Liefert das grösste Argument zurück.
range	Liefert das zweite Argument zurück, wenn es im Bereich zwischen dem ersten und dritten Argument ist.
if	Liefert das zweite Argument zurück, wenn das erste Argument nicht 0 ergibt, sonst das dritte Argument.
ifseq	Dasselbe wie mehrere if-Funktionen hintereinander.
Speziell	
integrate	Liefert das numerische Integral des ersten Argumentes vom zweiten Argument bis zum dritten Argument zurück.
sum	Ergibt die Summe des ersten Arguments, berechnet für jede Ganzzahl im Bereich vom zweiten bis zum dritten Argument.
product	Ergibt das Produkt des ersten Argumentes, berechnet für jede Ganzzahl im Bereich vom zweiten bis zum dritten Argument.
fact	Liefert die Fakultät des Arguments zurück.
gamma	Liefert the Eulersche Gammafunktion des Arguments zurück.
beta	Liefert die für die Argumente berechnete Betafunktion zurück.
W	Liefert die für die Argumente berechnete Lambertsche W-Funktion zurück.
zeta	Liefert die für die Argumente berechnete Riemannsche Zetafunktion zurück.
mod	Liefert den Rest der Division des ersten Arguments durch das zweite Argument.
dnorm	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
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}.

See also

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

See also

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

See also

Wikipedia [http://en.wikipedia.org/wiki/Trigonometric_functions#Cosine]
MathWorld [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*π/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.

See also

Wikipedia [http://en.wikipedia.org/wiki/Trigonometric_functions#Tangent]
MathWorld [http://mathworld.wolfram.com/Tangent.html]

asin-Funktion

Liefert den Arcussinus des Arguments.

Syntax

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.

See also

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

acos-Funktion

Liefert den Arcuscosinus des Arguments.

Syntax

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.

See also

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

atan-Funktion

Liefert den Arcustangens des Arguments zurück.

Syntax
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.

See also
Wikipedia [http://en.wikipedia.org/wiki/Inverse_trigonometric_functions]
MathWorld [http://mathworld.wolfram.com/InverseTangent.html]

sec-Funktion
Liefert den Sekans des Arguments zurück.

Syntax
sec(z)

Beschreibung:
Die `sec`-Funktion berechnet den Secans von `z`. Das Ergebnis hat die Einheit *Bogenmaß* oder Grad, abhängig von der aktuellen Einstellung. 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.

See also
Wikipedia [http://en.wikipedia.org/wiki/Trigonometric_functions#Reciprocal_functions]
MathWorld [http://mathworld.wolfram.com/Secant.html]

csc-Funktion
Liefert den Cosekans des Arguments zurück.

Syntax
csc(z)

Beschreibung:
Die `csc`-Funktion berechnet den Cosecans von `z`. Das Ergebnis hat die Einheit *Bogenmaß* oder Grad, abhängig von der aktuellen Einstellung. 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.

See also
Wikipedia [http://en.wikipedia.org/wiki/Trigonometric_functions#Reciprocal_functions]
MathWorld [http://mathworld.wolfram.com/Cosecant.html]

cot-Funktion
Liefert den Cotangens des Arguments zurück.

Syntax
cot(z)

Beschreibung:
Die `cot`-Funktion berechnet den Cotangens von `z`. Das Ergebnis hat die Einheit *Bogenmaß* oder Grad, abhängig von der aktuellen Einstellung. 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.

See also

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

asec-Funktion

Liefert den Arcusekans des Arguments zurück.

Syntax

asec(z)

Beschreibung:

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

See also

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

acsc-Funktion

Liefert den Arcuscosekans des Arguments zurück.

Syntax

acsc(z)

Beschreibung:

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

See also

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

acot-Funktion

Liefert den Arcuscotangens des Arguments zurück.

Syntax

acot(z)

Beschreibung:

Die `acot`-Funktion berechnet den Arcuscotangens von `z`. Das Ergebnis hat die Einheit *Bogenmaß* oder Grad, abhängig von der aktuellen Einstellung. acot(z) ist dasselbe wie 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`-Function liefert einen Wert im Intervall]-π/2;π/2] (]-90;90] wenn in Grad gerechnet wird), das ist die allgemeine Definition, obwohl der Bereich manchmal auch mit]0;π[definiert wird.

See also

Wikipedia [http://en.wikipedia.org/wiki/Inverse_trigonometric_functions]
MathWorld [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) = ½(e^z-e^{-z}) definiert.

See also

Wikipedia [http://en.wikipedia.org/wiki/Hyperbolic_function]
MathWorld [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) = ½(e^z+e^{-z}) definiert.

See also

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

See also

Wikipedia [http://en.wikipedia.org/wiki/Hyperbolic_function]
MathWorld [http://mathworld.wolfram.com/HyperbolicTangent.html]

asinh-Funktion

Liefert den Arkussinus Hyperbolicus des Arguments.

Syntax

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 asinh(sinh(z)) = z.

See also

Wikipedia [http://en.wikipedia.org/wiki/Hyperbolic_function]
MathWorld [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 acosh(cosh(z)) = z.

See also

Wikipedia [http://en.wikipedia.org/wiki/Hyperbolic_function]
MathWorld [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 atanh(tanh(z)) = z.

See also

Wikipedia [http://en.wikipedia.org/wiki/Hyperbolic_function]
MathWorld [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 $\mathrm{csch}(z) = 1/\sinh(z) = 2/(e^{z}-e^{-z})$ definiert.

See also

Wikipedia [http://en.wikipedia.org/wiki/Hyperbolic_function]
MathWorld [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 $\mathrm{sech}(z) = 1/\cosh(z) = 2/(e^{z}+e^{-z})$ definiert.

See also

Wikipedia [http://en.wikipedia.org/wiki/Hyperbolic_function]
MathWorld [http://mathworld.wolfram.com/HyperbolicSecant.html]

coth-Funktion

Liefert den Cotangens Hyperbolicus des Arguments.

Syntax

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 coth(z) = 1/tanh(z) = cosh(z)/sinh(z) = $(e^{z} + e^{-z})/(e^{z} - e^{-z})$ definiert.

See also

Wikipedia [http://en.wikipedia.org/wiki/Hyperbolic_function]
MathWorld [http://mathworld.wolfram.com/HyperbolicCotangent.html]

acsch-Funktion

Liefert den Arkuscosecans Hyperbolicus des Arguments.

Syntax

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 acsch(csch(z)) = z.

See also

Wikipedia [http://en.wikipedia.org/wiki/Hyperbolic_function]
MathWorld [http://mathworld.wolfram.com/InverseHyperbolicCosecant.html]

asech-Funktion

Liefert den Arkussecans Hyperbolicus des Arguments.

Syntax

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 asech(sech(z)) = z.

See also

Wikipedia [http://en.wikipedia.org/wiki/Hyperbolic_function]
MathWorld [http://mathworld.wolfram.com/InverseHyperbolicSecant.html]

acoth-Funktion

Liefert den Arkuscotangens Hyperbolicus des Arguments.

Syntax

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 acoth(coth(z)) = z. Für reelle Zahlen ist `acoth` im Interval [-1;1] nicht definiert.

See also

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

See also

Wikipedia [http://en.wikipedia.org/wiki/Exponential_function]
MathWorld [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 ½. 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.

See also

Wikipedia [http://en.wikipedia.org/wiki/Square_root]
MathWorld [http://mathworld.wolfram.com/SquareRoot.html]

root-Funktion

Liefert die n^{th} 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 root(3, x) verwenden.

See also

Wikipedia [http://en.wikipedia.org/wiki/Nth_root]
MathWorld [http://mathworld.wolfram.com/RadicalRoot.html]

ln-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.

See also

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

See also

Wikipedia [http://en.wikipedia.org/wiki/Common_logarithm]
MathWorld [http://mathworld.wolfram.com/CommonLogarithm.html]

logb-Funktion

Liefert den Logarithmus mit der Basis *n* des Arguments.

Syntax

logb(z, n)

Beschreibung:

Die `logb`-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.

See also

Wikipedia [http://en.wikipedia.org/wiki/Logarithm]
MathWorld [http://mathworld.wolfram.com/Logarithm.html]

Komplex

abs-Funktion

Liefert den Absolutwert des Arguments zurück.

Syntax
abs(z)

Beschreibung:
Die Funktion `abs` ergibt den absoluten Wert, auch Betrag genannt, einer Zahl `z`. Üblicherwerweise 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.

See also
Wikipedia [http://en.wikipedia.org/wiki/Absolute_value]
MathWorld [http://mathworld.wolfram.com/AbsoluteValue.html]

arg-Funktion
Ergibt das Argument des Parameters.

Syntax
arg(z)

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

See also
Wikipedia [http://en.wikipedia.org/wiki/Arg_(mathematics)]
MathWorld [http://mathworld.wolfram.com/ComplexArgument.html]

conj-Funktion
Ergibt die konjugiert komplexe Zahl des Arguments.

Syntax
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: conj(z) = re(z) - **i***im(z).

See also
Wikipedia [http://en.wikipedia.org/wiki/Complex_conjugation]
MathWorld [http://mathworld.wolfram.com/ComplexConjugate.html]

re-Funktion
Liefert den Realteil des Arguments zurück.

Syntax
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.

See also
Wikipedia [http://en.wikipedia.org/wiki/Real_part]
MathWorld [http://mathworld.wolfram.com/RealPart.html]

im-Funktion
Liefert den Imaginärteil des Arguments zurück.

Syntax
im(z)

Beschreibung:

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

See also

Wikipedia [http://en.wikipedia.org/wiki/Imaginary_part]
MathWorld [http://mathworld.wolfram.com/ImaginaryPart.html]

Runden

trunc-Funktion

Entfernt den gebrochenen Anteil des Arguments.

Syntax

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 trunc(re(z))+trunc(im(z))**i** zurück.

See also

Wikipedia [http://en.wikipedia.org/wiki/Truncate]
MathWorld [http://mathworld.wolfram.com/Truncate.html]

fract-Funktion

Liefert den gebrochenen Anteil des Argumentes.

Syntax

fract(z)

Beschreibung:

Die `fract`-Funktion liefert den gebrochenen Anteil von *z* zurück. Die Funktion entfernt den *Ganzzahl* Anteil von *z*, also fract(z) = z - 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 fract(re(z))+fract(im(z))**i**.

See also

Wikipedia [http://en.wikipedia.org/wiki/Floor_and_ceiling_functions#Fractional_part]
MathWorld [http://mathworld.wolfram.com/FractionalPart.html]

ceil-Funktion

Rundet das Argument auf.

Syntax

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 ceil(re(z))+ceil(im(z))**i** zurück.

See also

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

See also

Wikipedia [http://en.wikipedia.org/wiki/Floor_and_ceiling_functions]
MathWorld [http://mathworld.wolfram.com/FloorFunction.html]

round-Funktion

Rundet eine Zahl auf die angegebene Anzahle 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

See also

Wikipedia [http://en.wikipedia.org/wiki/Rounding]
MathWorld [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/abs(z), wenn *z* eine komplexe Zahl ergibt.

See also

Wikipedia [http://en.wikipedia.org/wiki/Sign_function]
MathWorld [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 imaginaren Anteil hat. u(z) liefert 1 für $z \geq 0$ und 0 für $z < 0$.

See also

Wikipedia [http://en.wikipedia.org/wiki/Unit_step#Discrete_form]
MathWorld [http://mathworld.wolfram.com/HeavisideStepFunction.html]

min-Funktion

Findet und liefert den Minimalwert der Argumente.

Syntax

min(A,B,...)

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(re(A), re(B), ...) + min(im(A), im(B), ...)**i** zurück.

max-Funktion

Findet und liefert den Maximalwert der Argumente.

Syntax

max(A,B,...)

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(re(A), re(B), ...) + max(im(A), im(B), ...)**i** zurück.

range-Funktion

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

Syntax

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

Liefert das zweite Argument, wenn das erste Argument einen von Null unterschiedlichen Wert ergibt, sonst wird das dritte Argument zurückgeliefert.

Syntax

if(cond,a,b)

Beschreibung:

Die `if`-Funktion liefert *a*, wenn *cond* einen von Null unterschiedlichen Wert ergibt. Ergibt *cond* 0, wird *b* zurückgegeben. Die Argumente können beliebige *numerische Ausdrücke* sein, die *reelle Zahlen* oder *Komplexe Zahlen* ergeben.

ifseq-Funktion

Wertet eine Reihe von if-Abfragen aus und liefert das erste Ergebnis, dessen Bedingung ungleich 0 ist.

Syntax

ifseq(cond1, f1, cond2, f2, ... , condn, fn [,fz])

Beschreibung:

Die `ifseq`-Funktion berechnet *cond1*, und wenn das Ergebnis von Null unterschiedlich ist, wird *f1* berechnet und zurückgegeben. Andernfalls wird *cond2* berechnet, und wenn das Ergebnis von Null unterschiedlich ist, wird *f2* zurückgeliefert usw. Wenn keine der Bedingungen wahr ist, wird *fz* zurückgeliefert. *fz* ist optional; wenn nicht angegeben, liefert `ifseq` einen Fehler, wenn keine Bedingung zutrifft. Hat *ifseq* 3 Argumente, verhält sie sich so wie die `if`-Funktion. Die Argumente können beliebige *numerische Ausdrücke* sein, die *reelle Zahlen* oder *Komplexe Zahlen* ergeben.

Speziell

integrate-Funktion

Returns the numeric integral of the given expression over the given range.

Syntax

integrate(f,var,a,b)

Beschreibung:

The `integrate` function returns the numeric integral of *f* with the variable *var* from *a* to *b*. This is mathematically written as:

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

This integral is the same as the area between the function *f* and the x-axis from *a* to *b* where the area under the axis is counted negative. *f* may be any function with the variable indicated as the second argument *var*. *a* and *b* may be any *numerische Ausdrücke* that evaluate to *reelle Zahlen* or they can be `-INF` or `INF` to indicate negative or positive infinity. `integrate` does not calculate the integral exactly. Instead the calculation is done using the Gauss-Kronrod 21-point integration rule adaptively to a estimated relative error less than 10^{-3}.

Beispiele

f(x)=integrate(t^2-7t+1, t, -3, 15) will integrate f(t)=5t^3+t^2-7t+1 from -3 to 15 and evaluate to 396. More useful is f(x)=integrate(s*sin(s), s, 0, x). This will plot the integral of f(s)=s*sin(s) from 0 to x, which is the same as the definite integral of f(x)=x*sin(x).

See also

Wikipedia [http://en.wikipedia.org/wiki/Integral]
MathWorld [http://mathworld.wolfram.com/Integral.html]

sum-Funktion

Returns the summation of an expression evaluated over a range of integers.

Syntax

sum(f,var,a,b)

Beschreibung:

The `sum` function returns the summation of *f* where *var* is evaluated for all integers from *a* to *b*. This is mathematically written as:

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

f may be any function with the variable indicated as the second argument *var*. *a* and *b* may be any *numerische Ausdrücke* that evaluate to *integers*.

See also

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

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

product-Funktion

Returns the product of an expression evaluated over a range of integers.

Syntax

product(f,var,a,b)

Beschreibung:

The `product` function returns the product of *f* where *var* is evaluated for all integers from *a* to *b*. This is mathematically written as:

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

f may be any function with the variable indicated as the second argument *var*. *a* and *b* may be any *numerische Ausdrücke* that evaluate to *integers*.

See also

Wikipedia [http://en.wikipedia.org/wiki/Multiplication#Capital_pi_notation]
MathWorld [http://mathworld.wolfram.com/Product.html]

fact-Funktion

Liefert die Fakultät des Arguments zurück.

Syntax

fact(z)

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 fact(n)=n(n-1)(n-2)...1 definiert, und die Beziehung zur `gamma`-Funktion ist fact(n)=gamma(n+1).

See also

Wikipedia [http://en.wikipedia.org/wiki/Factorial]
MathWorld [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 Γ(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 fact(n)=gamma(n+1). Die mathematische Definition der Gammafunktion ist:

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

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

See also

Wikipedia [http://en.wikipedia.org/wiki/Gamma_function]
MathWorld [http://mathworld.wolfram.com/GammaFunction.html]

beta-Funktion

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

Syntax

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 beta(m, n) = gamma(m) * gamma(n) / gamma(m+n).

See also

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

See also

Wikipedia [http://en.wikipedia.org/wiki/Lambert_w_function]
MathWorld [http://mathworld.wolfram.com/LambertW-Function.html]

zeta-Funktion

Liefert den Wert der Riemannschen Zetafunktion zum Argument.

Syntax

zeta(z)

Beschreibung:

Die `zeta`-Funktion liefert das Ergebnis der Riemannschen 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.

See also

Wikipedia [http://en.wikipedia.org/wiki/Riemann_zeta_function]
MathWorld [http://mathworld.wolfram.com/RiemannZetaFunction.html]

mod-Funktion

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

Syntax

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.

See also

Wikipedia [http://en.wikipedia.org/wiki/Modular_arithmetic]
MathWorld [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 μ=0=0 und σ=1 sind. x, μ und σ können beliebige *numerische Ausdrücke* sein, die *reelle Zahlen* darstellen, wobei σ > 0 ist. Die Normalverteilung ist folgendermaßen definiert:

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

See also

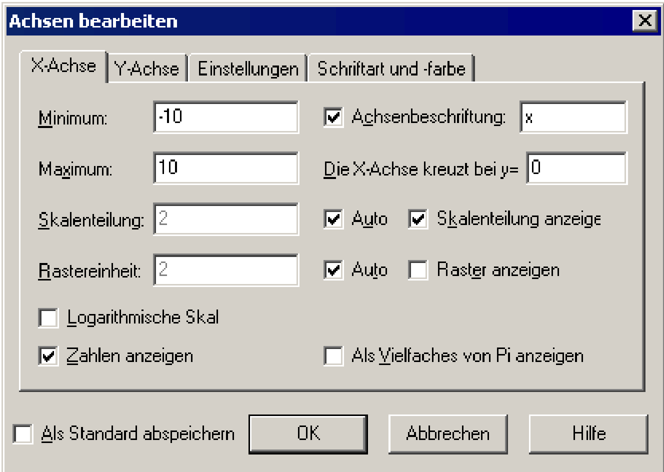
Wikipedia [http://en.wikipedia.org/wiki/Normal_distribution]
MathWorld [http://mathworld.wolfram.com/NormalDistribution.html]

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 größenmäßig dem Fenster anzupassen.

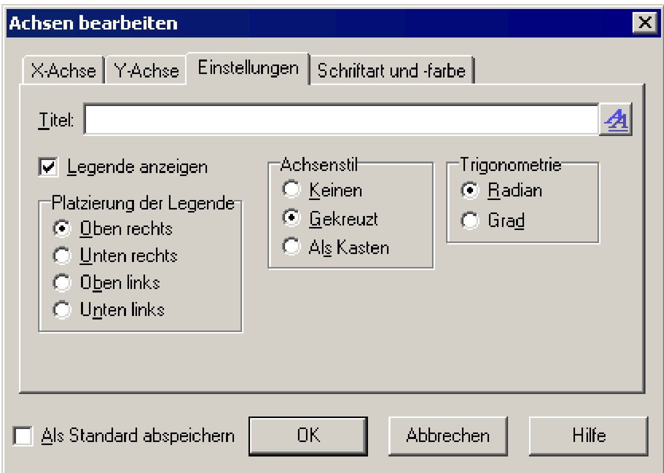
Auto
: Wird diese Option aktiviert, enthält *Rastereinheit* denselben Wert wie *Skalenteilung*.

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 Pi anzeigen
: Wird diese Option aktiviert, stellt Graph die Skalenteile als Vielfache von π dar, z.B. 3π/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.

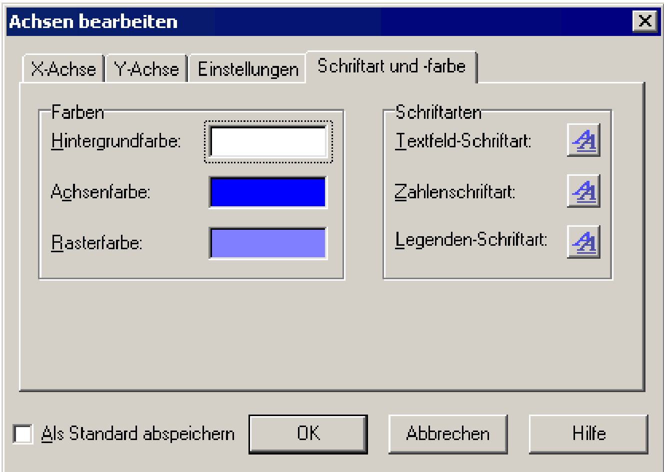
Achsenstil

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

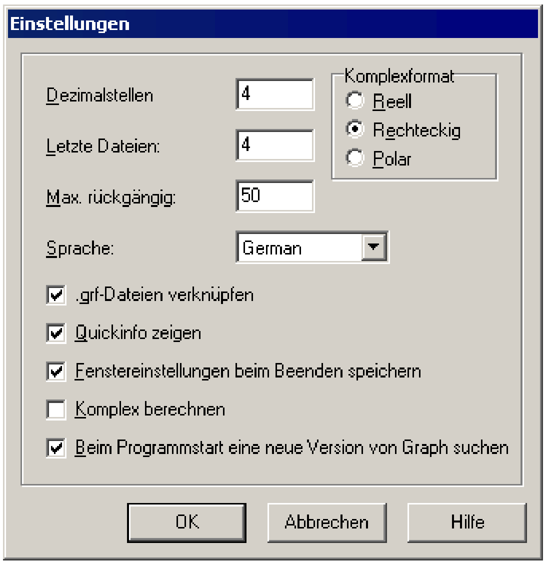
Sie können die Schriftarten ändern, die für Achsenbeschriftungen, Skalenteilung und *Legende* verwendet werden.

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.

Custom decimal separator

Decimal separator used when data are exported to files and the clipboard. When disabled the decimal separator from the Windows Regional settings is used. This is not used for expressions entered into Graph, which always use a dot as decimal separator.

.grf-Dateien verknüpfen

Ein Häkchen in diesem Feld zeigt an, daß der Dateityp .grf mit diesem Programm verknüpft ist. Das Programm startet automatisch und lädt die Datei, wenn Sie darauf im Explorer doppelklicken.

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 derZahl einen Fehler. *Normalform* bedeutet, daß *Komplexe Zahlen* als a+b**i** dargestellt werden, mit a als Realteil und b als Imaginärteil. *Polarform* bedeutet, daß Zahlen als a∠θ 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.

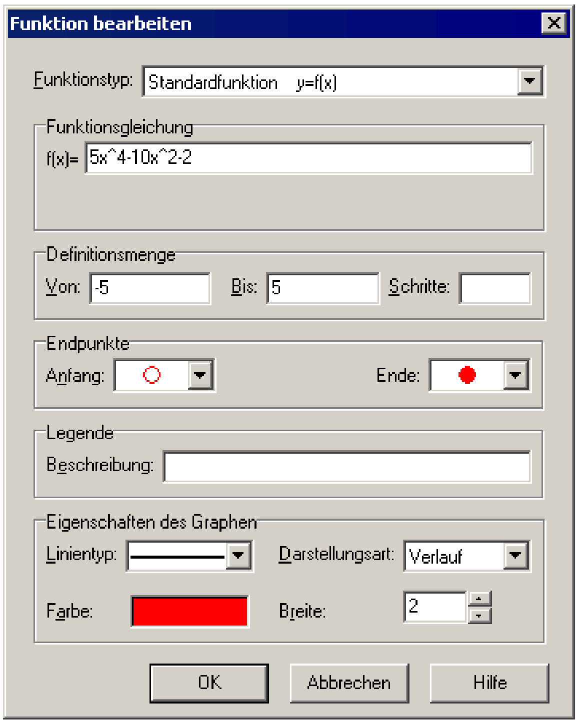
Beachten Sie, daß Sie in der Eingabemaske Evaluieren in einigen Fällen - je nach *Komplexzahlen-Format*-Einstellung - unterschiedliche Resultate erhalten können: Bei *Reell* versucht Graph, ein reelles Ergebnis zu finden, während *Normalform* und *Polarform* ein nicht-reelles Ergebnis für dieselbe Berechnung 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 Polarunktion 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)*cos(t), y(t)=r(t)*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 denen Graphen zeichnen lassen.

Definitionsmenge

Sie können einen Intervall für die unabhängige Variable wählen. *Von* and *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.

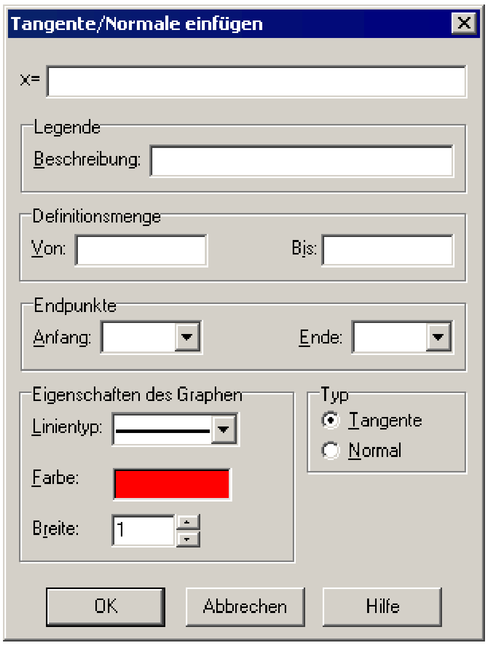
Eigenschaften des Graphen

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.

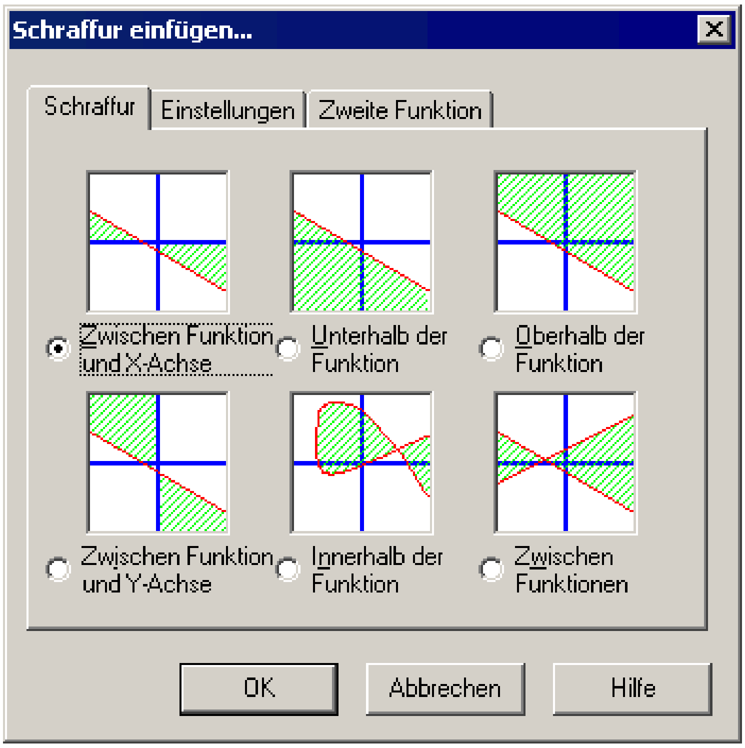
Eigenschaften des Graphen

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.

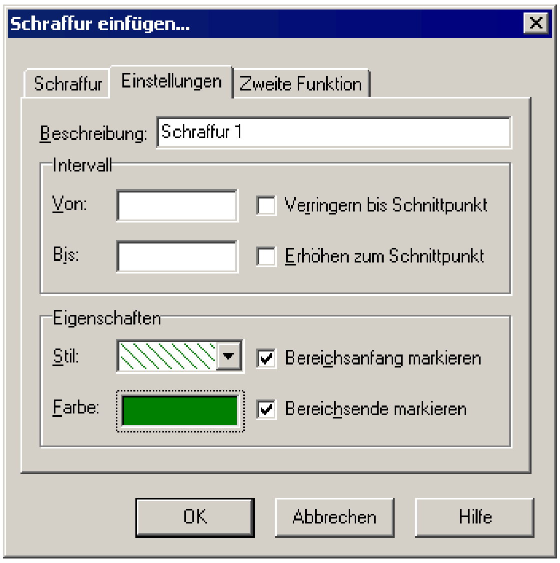
Innerhalb der Funktion
: Es wird der Bereich innerhalb des Funktionsgraphen im gewählten Intervall schraffiert. Mit *Verringern bis Schnittpunkt* oder *Erhöhen zum Schnittpunkt* wird der Intervall verkleinert oder vergrößert, bis der Graph sich selbst kreuzt. Dies ist besonders nützlich, um einen umschlossenen 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.

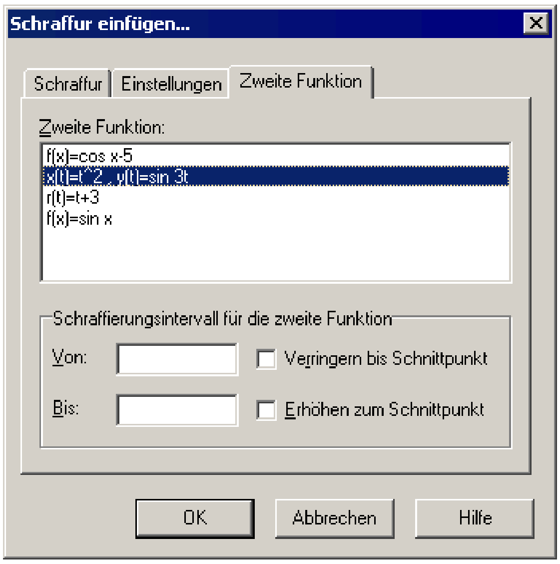
Stil
: Hier können Sie zwischen verschiedenen Schraffurtypen wählen.

Farbe
: Hier können Sie die Farbe der Schraffur auswählen.

Mark border
: Check this to draw a line around the border of the shading. Uncheck it to leave the shading without a border, which is useful if you want two shadings to look as one.

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.



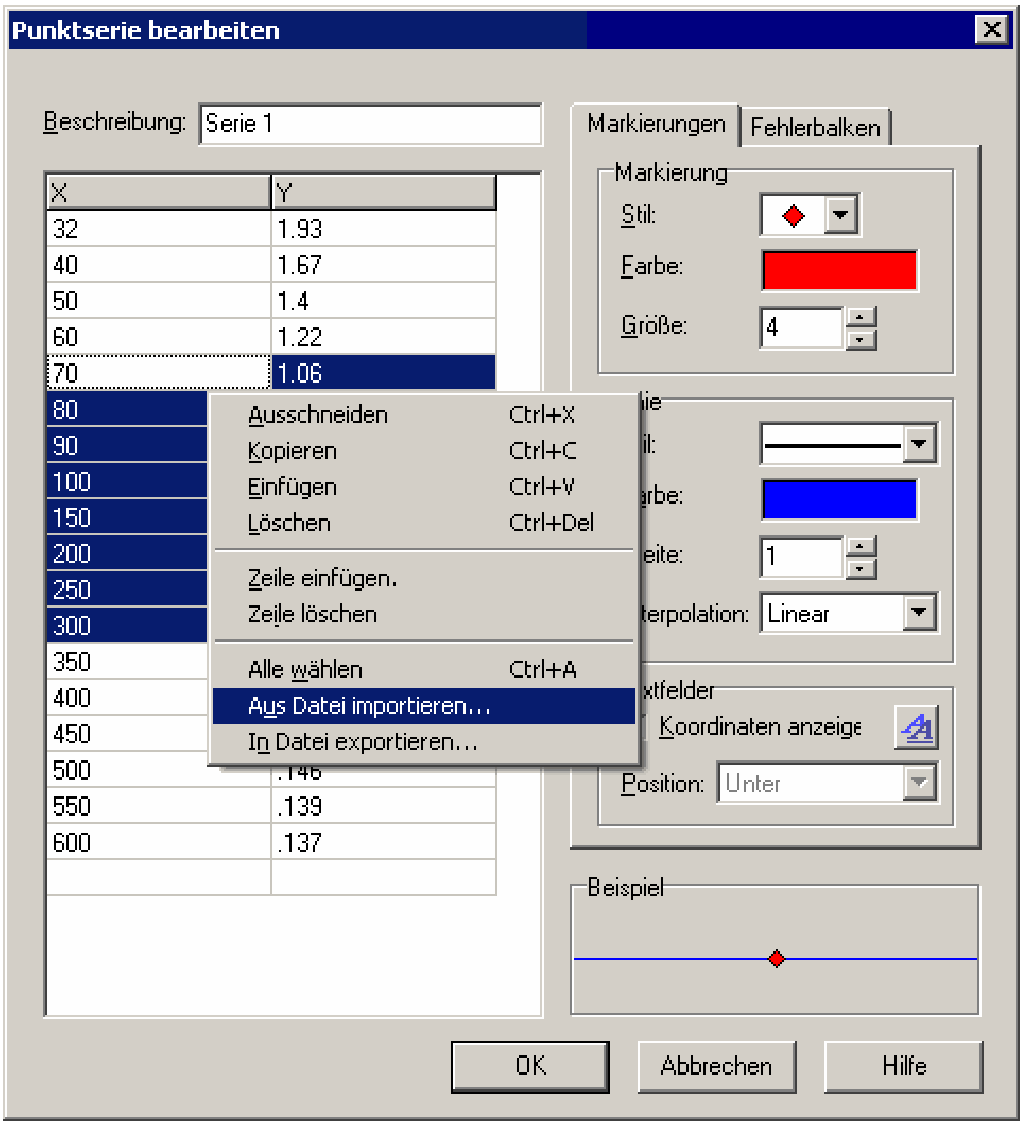
Schraffierungsintervall für die zweite Funktion

Hiermit wählt man den Intervall für die zweite Funktion, genauso wie Sie im Reiter *Einstellungen* den Intervall für die erste Funktion eingestellt haben. Dies ist nur für Parameterfunktionen möglich, denn bei den Standardfunktionen haben die erste und die zweite Funktion immer denselben Intervall. Geben Sie bei einer Parameterfunktion weder Intervall-Beginn noch -Ende an, 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.

Coordinate type

You need to choose between the type of coordinates used for the points. *Cartesian* is used when you want to specify (x,y)-coordinates. *Polarform* is used when you want to specify (θ,r)-coordinates, where θ is the angle and *r* is the distance from the origin. The angle θ is in *Bogenmaß* or degrees depending on the current setting.

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.
You can choose between three types of interpolation: *Linear* will draw straight lines between the markers. *Kubische Spline* will draw a natural cubic spline [http://en.wikipedia.org/wiki/Cubic_splines] which is a nice smooth line connecting all the points with 3rd degree polynomials. *Trigonometrisch* will draw half cosine curves between the points, which might not look as smooth as the cubic splines but they never undershoot/overshoot like the cubic splines can do.

Textfelder

Put a check in *Koordinaten anzeigen* to show Cartesian or polar coordinates at each point. You may use the button to change the font, and the drop down box to select whether the labels are shown over, below, to the left or to the right of the points.

Fehlerbalken

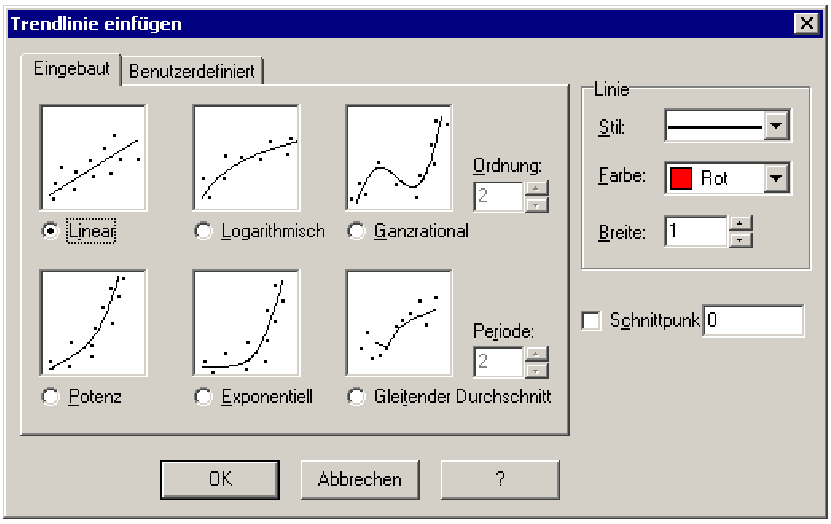
Here you can choose to show horizontal or vertical error bars, also known as uncertainty bars. They are shown as thin bars at each point in the point series indicating the uncertainty of the point. There are three ways to indicate the size of the error bars: *Fest* is used to specify that all points have the same uncertainty. *Relativ* is used to specify a percentage of the x- or y-coordinate for each point as uncertainty. *Benutzerdefiniert* will add an extra column to the table where you may specify a different uncertainty value for each point. All uncertainties are ±values. Custom Y-errors are also used to weight the points when creating trendlines.

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*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 $\Sigma(y_i-f(x_i))^2$ 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*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 $f(x) = a_n*x^n + ... + a_3*x^3 + a_2*x^2 + a_1*x + a_0$ dargestellt wird. $a_0...a_n$ sind Konstanten. n ist der Grad des Polynoms. Die Anzahl der Punkte muß höher als der Grad sein.

Potenz

Eine Potenzfunktion wird mit $f(x) = a*x^b$ 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 doppeltlogarithmischen Koordinatensystem. Die Punktserie wird deshalb in ein doppeltlogarithmisches Koordinatensystem übertragen, und die Potenzfunktion mit der kleinsten Summe der Quadrate wird ermittelt.

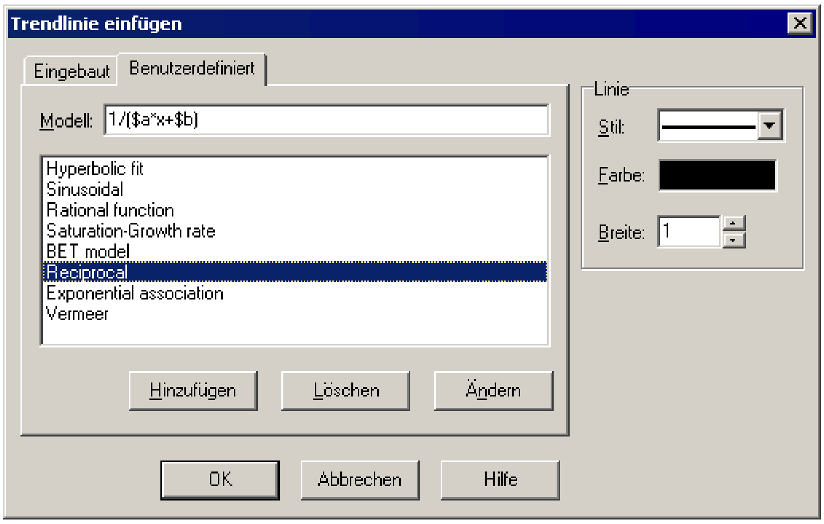
Exponentiell

Eine Exponentialfunktion wird mit $f(x) = a*b^x$ 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)=$a*x^$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 $\Sigma(y_i-f(x_i))^2$ 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.

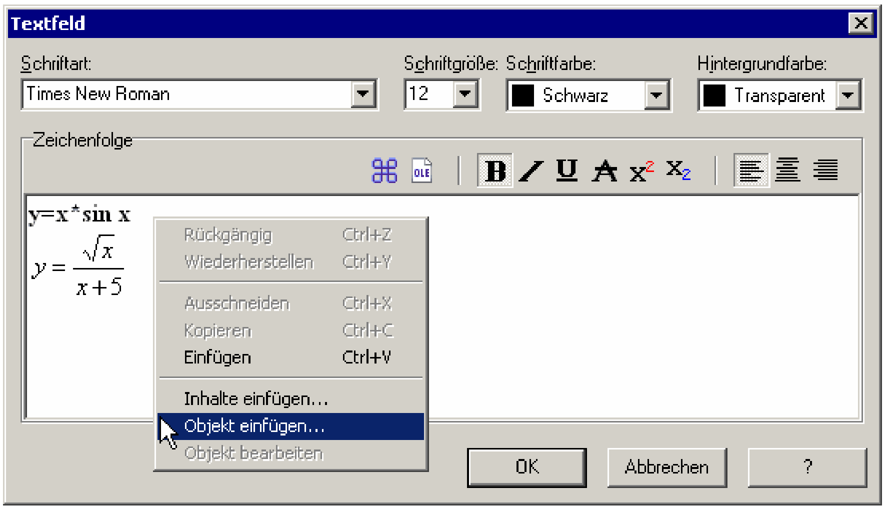
In seltenen Fällen existiert mehr als ein Minimum. In diesem Fall wird das Minimum gefunden, das am nächsten an der Schätzung liegt, obwohl es vielleicht nicht das beste ist.

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/($a*x+$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 → 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 ⌘ 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 als einem OLE-Objekt in der Zwischenablage läßt sich mit Spezielles 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 un 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

This dialog is used to insert a relation in the coordinate system. `Relation` is a common name for inequalities and equations, also known as implicit functions. To insert a relation you use the menu item Funktion → Relation einfügen.... To change an existing relation, you first select it in the *Funktionsliste* and use Funktion → Bearbeiten....

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 Wert müssen durch ein = getrennt sein. Die Gleichung `x^2 + y^2 = 25` wird z. B. einen Kreis mit dem Radius 5 zeichnen.

Eine Gleichung ist eine Aussage, daß zwei Werte ungleich gleich sind; die Werte müssen durch einen der vier folgenden Operatoren getrennt sein: <, >, <=, >=. Eine Ungleichung ist beispielsweise `abs(x) + 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

Here you can enter optional constraints, which can be any *numerischer Ausdruck*. The relation will only be valid and plotted where the constraints are fulfilled, i.e. evaluates to a non-zero value. The constraints usually consist of a series of inequalities separated with the logical operators (`and`, `or`, `xor`). As for the relation, *x* and *y* are used as the independent variables.

Z. B. Bei der Relation `x^2 + y^2 < 25` (ein schraffierter Kreis), wird durch die Nebenbedingungen `x > 0 und 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.

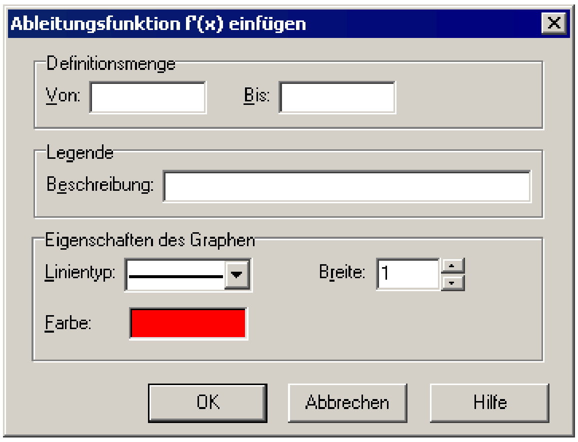
Eigenschaften

Hier können Sie Schraffurstile für Ungleichungen, Farbe und Breite für Gleichungenwählen.Gleichungs. Die Schraffur *Stil* wird für Ungleichungen benutzt und für Gleichungen ignoriert. Man kann nur überlappende Ungleichungen sehen, wenn sie unterschiedlichen Schraffurstilen haben. Die *Breite* zeigt an das Größe der Linie für Gleichungen und die Breite der Grenzlinie für Ungleichungen. Bei Ungleichungen kann die Breite auf 0 gesetzt werden, um das Zeichnen eine 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 → Ableitungsfunktion f'(x) einfügen....

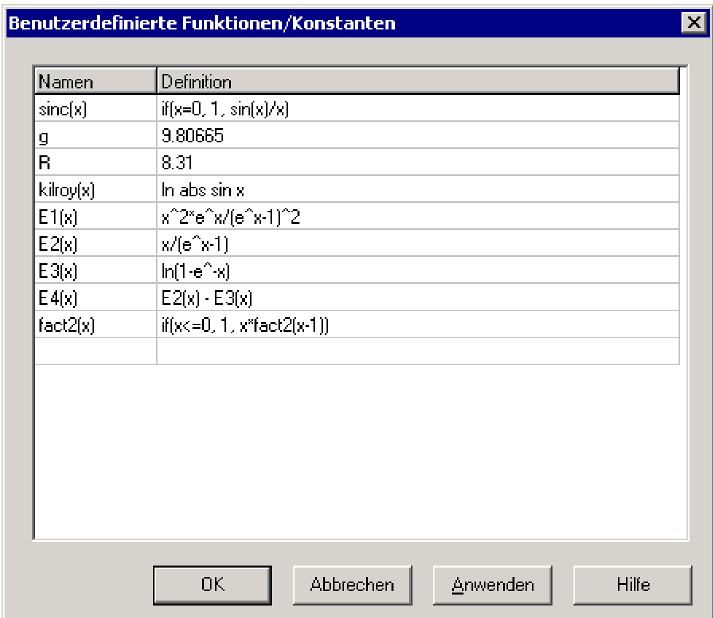
Handelt es sich um eine Standardfunktion, gibt die erste Ableitungsfunktion die Steigung der abgeleiteten Funktion an. Die Ableitungsfunktion erhält man in diesem Fall durch Differenzierung der gegebenen Funktion nach x: f'(x) = df(x)/dx



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 getrannt, 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.

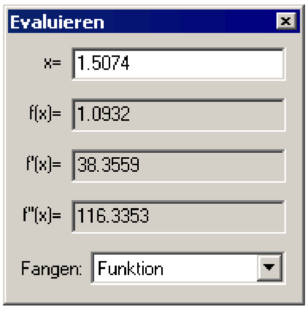
Evaluieren

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+b**i**, a∠ 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
: The cursor will snap to the nearest point of the selected function.

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.

Extreme y-value

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

Fläche

When Berechnen → Fläche is selected the dialog is used to calculate signed areas. For standard functions, parametric functions and tangents the result is the signed area between the graph and the x-axis (the real x-axis and not necessarily the visible one) for the given range.

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

Bei den übrigen Funktionen erhält jene Fläche ein negatives Vorzeichen, wenn der sie begrenzende Funktionsgraph unterhalb der X-Achse verläuft oder wenn die Funktion von einem höheren auf einen geringeren X-Wert fällt.

You can either enter the range in the edit boxes or select the range with the mouse. The calculated area will be shown below the range, and the area will be marked with a shading in the coordinate system. The calculation is done using the Gauss-Kronrod 21-point integration rule adaptively with as much accuarcy as possible. If an estimated relative error less than 10^{-4} cannot be reached, no result will be shown.

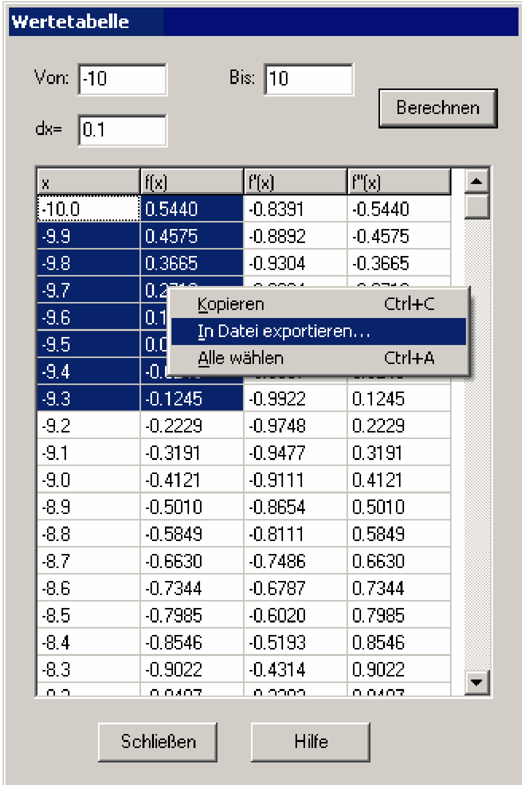
Länge eines Kurvenstücks

Berechnen → Länge eines Kurvenstücks wird benutzt, um die Länge eines Kurvenstücks zwischen zwei Punkten zu berechnen. Die Punkte können entweder direkt eingegeben ode rmit 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

The dialog shown below is used to evaluate the selected function for a range. First select a function in the *Funktionsliste* and use the menu item Berechnen → Tabelle to show the dialog. You specify the first and last value of the independent variable in the *Von* and *Bis* fields. In the Δ*x* or Δ*t* field you specify the increment of the independent variable at each evaluation.

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 können lassen sich im Kontextmenü ausblenden. Bei länger dauernden Berechnungen wird ein Fortschrittsbalken angezeigt.



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.

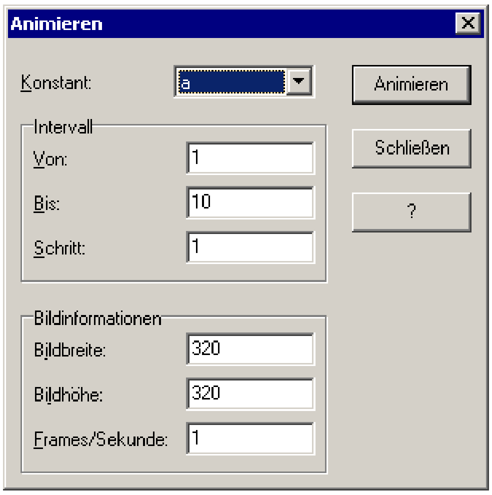
Auf der linken Seite der Tabellen verwandelt sich der Mauszeiger in einen Rechts-Pfeil. Jetzt lassen sich mit der Maus ganze Reihen auswählen. Im oberen Tabellenbereich verwandelt sich der Mauszeiger in einen Nach-Unten-Pfeil. Jetzt lassen sich mit der Maus ganze Spalten auswählen. Die komplette Tabelle 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 Systemspeicher 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.

Frame information
: 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
: Wenn aktiviert, wird die Animation in einer Endlosschleife abgespielt.

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.

Aktuelles Bild speichern...
: 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 Aktuelles Bild speichern... für jedes Einzelbild der Animation.

Als Bild speichern

Use the menu item Datei → Als Bilddatei abspeichern... to save the shown coordinate system as an image file. When the menu item has been chosen, a standard *Speichern als* dialog will appear. In this dialog you write a filename, choose a directory and select one of the following image types:

Windows Enhanced Metafile (emf)
: Metafiles are usually preferred because they are small and look nice even when scaled. Though emf files are widely supported under MS Windows, they are not very portable.

Scalable Vector Graphics (svg)
: This is a format for portable metafiles and should therefore be preferred for files placed on the Internet. However the format is still not supported by all browsers.

Portable Network Graphics (png)
: Portable Network Graphics (png) is a format that is better compressed than bmp files. This is the most sustainable format for web pages, because it is small and can be understood by all browsers.

Windows Bitmap (bmp)
: Windows Bitmap (bmp) is a standard format supported by almost all Windows programs that can read graphics files.

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.

The Options... button in the save dialog can be used to change the image size. You may also be able to change other settings depending on the chosen image format.

Plugins

To use the plugin system in Graph you need to install Python 3.1 from http://www.python.org. Documentation of the Python language may be found installed with Python or online [http://docs.python.org/3.1/].

Plugins

Plugins are Python scrips and are usually distributed in source form as .py files but the can also be distributed as compiled .pyc files. The plugin files are placed in the `Plugin` directory where Graph is installed, and will automatically be found and loaded by Graph.

Warnung

Plugins are scripts, which are just small programs that run inside Graph and interacts with Graph. But a plugin can do anything that a program with the same rights can do. This means that if Graph is run with administrator rights, it is possible to write a plugin that erases the whole harddrive. Therefore you should be careful about which plugins you use and only install plugins from a trusted source, or at least you should check the source code for suspicious parts.

Python interpreter

The plugin system also gives access to a Python interepreter by pressing **F11**. In this interpretter you can write Python expressions and that way do very advanced things in Graph. It is also an easy way to test code before it is used in a plugin.

Acknowledgements

Bibliotheken

dxgettext
Translation library.
Copyright © Lars B. Dybdahl (`Lars@dybdahl.dk`) et al.
http://dybdahl.dk/dxgettext/

EasyNSE
Library for creating shell extensions.
Copyright © 2005 Cool Breeze Software
http://www.mustangpeak.net

PDFlib-Lite
Used to create PDF files.
Copyright © 1997-2005 Thomas Merz & PDFlib GmbH
http://www.pdflib.com

Python
Used for plugin support and advanced interaction.
Copyright © 2001-2006 Python Software Foundation
http://www.python.org

GNU Scientific Library
Numeric library.
Copyright © 2009 Free Software Foundation, Inc.
http://www.gnu.org/software/gsl/

Boost
Peer-reviewed C++ library.
http://www.boost.org

Übersetzungen

Sprache	Programm	Hilfedatei	Übersetzer
Arabisch	Ja	Nein	Abdellah Chelli
Baskisch	Ja	Nein	Xabier Maiza
Chinesisch (traditionell)	Ja	Nein	Dung Jian-Jie
Chinesisch (vereinfacht)	Ja	Nein	Lala Sha
Kroatisch	Ja	Nein	Hasan Osmanagić
Tschechisch	Ja	Nein	Martin Stružský & Pavlína Krausová
Dänisch	Ja	Ja	Ivan Johansen, Michael Bach Ipsen & Erik Lyngholt Nielsen
Niederländisch	Ja	Nein	Etienne Goemaere
Englisch	Ja	Ja	Ivan Johansen
Finnisch	Ja	Nein	Pekka Lerssi
Französisch	Ja	Ja	Jean-Pierre Fontaine
Deutsch	Ja	Ja	Frank Httemeister, Michael Bach Ipsen & Sebastian Stütz
Griechisch	Ja	Nein	Theodoros Kannas
Ungarisch	Ja	Nein	Gabor Magyari
Hebräisch	Ja	Nein	ינד ינדור
Italienisch	Ja	Ja	Serena Alessandro & 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 Głąb
Portugiesisch (Brasilien)	Ja	Nein	Jorge, Mara, Deivid & Fernanda
Portugiesisch (Portugal)	Ja	Nein	Jorge Geraldes
Russisch	Ja	Nein	Ivans Leonovs
Serbisch	Ja	Nein	Jasmina Malinovic & Branimir Krstic
Slovenisch	Ja	Ja	Jernej Baša, Rok Štokelj & Barbara Pušnar
Spanisch	Ja	Ja	Dr. Gustavo Criscuolo, Francisco Oliver & Alejandro Arce
Schwedisch	Ja	Nein	Pär Smårs
Türkisch	Ja	Nein	Mumtaz Murat Arik
Vietnamesisch	Ja	Nein	Trung

Verschiedene

The icon for Graph was designed by Jonathan Holvey.

Glossar

Bogenmaß
: Das Bogenmaß ist ein dimensionsloses 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 $180°/\pi$ bzw. der in Grad angegebene Winkel mit $\pi/180°$ 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 ...,-3,-2,-1,0,1,2,3,... 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
: Complex numbers are a superset of real numbers. Complex numbers are two dimensional and is most often written on rectangular form as a+b**i** where *a* is the real part and *b* is the imaginary part. The imaginary unit **i** is defined as **i**^2=-1. Complex numbers can also be shown on polar form as a∠θ where *a* is the absolute value of the number and θ is the angle of the number in radians or degrees.

  Komplexe Zahlen können im Dialog *Evaluieren* 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 ovn 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 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.