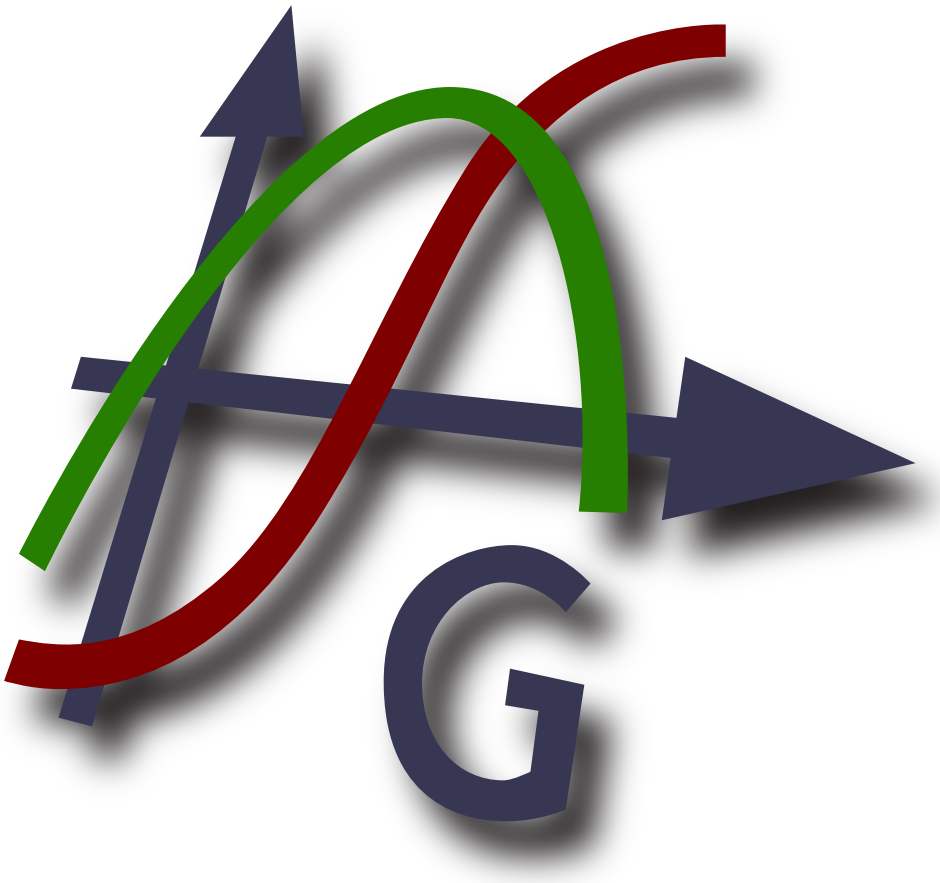
Graph



Version 4.4

Innehållsförteckning

Vad är Graph?

Graph is a program designed to draw graphs of mathematical functions in a coordinate system and similar things. The program is a standard Windows program with menus and dialogs. The program is capable of drawing standard functions, parametric functions, polar functions, tangents, point series, shadings and relations. It is also possible to evaluate a function for a given point, trace a graph with the mouse and much more. For more information on using the program see *How to use 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 Windows 2000, Windows XP, Windows Vista och 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 supportforum [http://www.padowan.dk/forum].

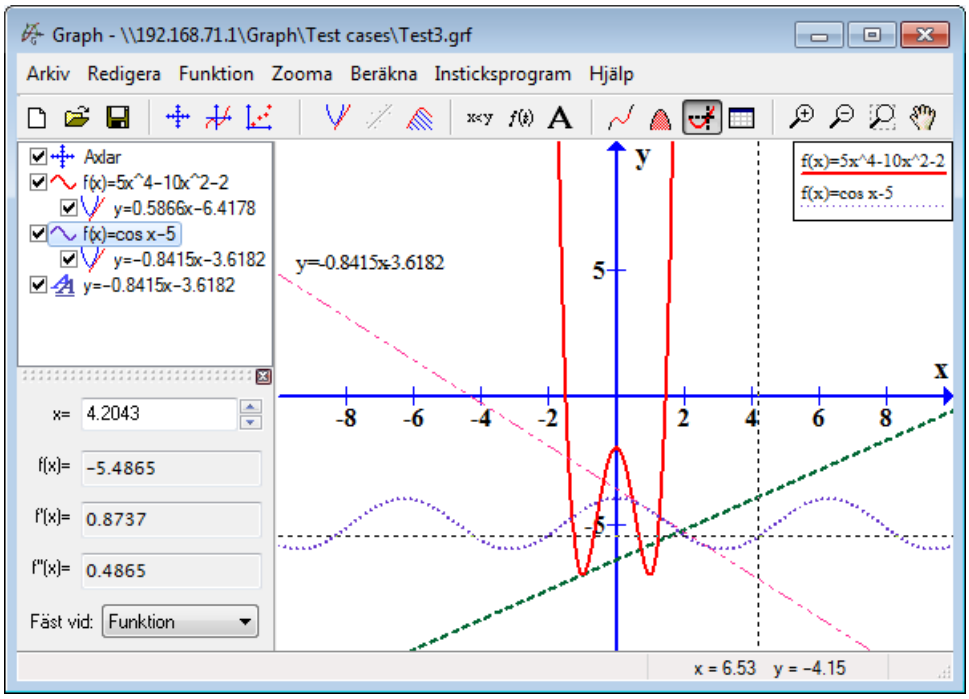
When you send a bug report, please write the following:

- Which version are you using? This is shown in the Hjälp → Om Graph... dialog box. You should check that you are using the newest version as the bug may have been fixed already.
- Explain what happens and what you expected to happen.
- Explain carefully how I can reproduce the bug. If I can't see what you see, it is very difficult for me to solve the problem.

How to use Graph

When the program starts, you will see the main window shown below. This window shows the graphing area to the right with the coordinate system where the graphs you insert will be shown. You can use the menu or the buttons on the toolbar to show different dialog boxes to insert a function, edit functions, delete functions etc. You can find a description of all the menu items.

The toolbar may be customized by right clicking on the bar and selecting Anpassa verktygsfältet... from the popup menu. You can then customize the toolbar by dragging commands to and from the bar. The status bar at the bottom of the window shows tooltips or other information to the left and the coordinates located at the mouse pointer to the right.



You can add new elements to the coordinate system from the Function menu. For example if you want to add a new function you use the menu item Funktion → Lägg till funktion...

The *funktionslista* to the left shows a list of functions, tangents, point series, shadings and relations you have added. If you want to manipulate anything in the list, just select it and use the Funktion menu. You can also right click on an item in the list to get the context menu with available commands. An item may be edited by double clicking on it.

The Beräkna menu contains commands to make calculations on functions, for example evaluations at specific coordinates or given intervals.

Installation och start

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:

Fil(er)	Beskrivning
`Graph.exe`	Programfilen.
`PDFlib.dll`	Library used to create PDF files.
`Thumbnails.dll`	Shell extension for showing thumbnails of grf-files in Explorer.
`Locale\*.mo`	Översättningar av programvaran.
`Help\*.chm`	Hjälpfiler på olika språk.
`Plugins\*.py`	Some examples of plugins. Custom plugins can be placed here too.
`Lib\*.py`	Library files used by plugins.
`Examples\*.grf`	Some examples that can be opened in Graph.

The installation will create a shortcut in the Start menu, which may be used to start the program. During the installation you select the preferred language. This can later be changed from the Tillval dialog.

If an older version of the program is already installed, the installation suggests you install in the same directory. You can just install over the old version. There is no need to uninstall the old version first, but make sure the old version is not running while installing.

The Graph Setup can take the parameters specified in the table below. These are especially useful when you want to automate the installation.

Parameter	Beskrivning
/SILENT	Instructs the Setup to be silent, which means that the wizard and the background window are not displayed but the installation progress window is. Everything else is normal so for example error messages during installation are displayed. If a restart is necessary, a *Starta om nu?* message box is displayed.
/VERYSILENT	Instructs the Setup to be very silent. This is the same as silent with the addition that the installation progress window is not displayed either. If a restart is necessary, the Setup will reboot without asking.
/NORESTART	Instructs Setup not to reboot even if it's necessary.
/LANG=language	Specifies the language to use. *language* specifies the English name of the language. When a valid /LANG parameter is used, the *Välj språk* dialog will be suppressed.
/DIR=x:\dirname	Overrides the default directory name displayed on the *Select destination location* wizard page. A fully qualified pathname must be specified.

Deinstallation

Uninstallation is done from *Add/Remove Programs* in the *Control Panel*. Just select Graph and click on the *Change/Remove button*. This will remove all traces of the program. If files were added to the installation directory after the installation, you will be asked if you want to delete them. Make sure Graph is not running while uninstalling.

Start

Usually Graph is started from the link in the Start menu. A .grf file can be passed as parameter, in which case Graph will open the specified file. In addition to this the parameters in the table below can be passed to Graph on the command line.

Parameter	Beskrivning
`/SI=file`	Used to save an opened .grf file as an image file. The file type can be any of the image formats supported by Graph.
`/WIDTH=width`	Used in combination with /SI to specify the width in pixels of the image to be saved.
`/HEIGHT=height`	Used in combination with /SI to specify the height in pixels of the image to be saved.

Vanliga frågor

Fråga: What are the system requirements of Graph?

A: Graph requires Microsoft Windows 2000 or newer. It has been tested under Windows 2000, Windows XP, Windows Vista och Windows 7.

Fråga: Will Graph run under Linux?

A: Graph is a native Windows application and not tested under Linux, but several users have informed me that Graph runs without problems under Linux with Wine.

Fråga: Will Graph run on a Macintosh?

A: As with the above, you cannot run Graph directly on a Mac. But it should be possible with some kind of Windows emulator.

Fråga: När släpps nästa version?

A: När den är klar.

Fråga: How can I move the coordinate system?

A: When you hold down the **Ctrl** key you can use the arrow keys to move the coordinate system. You can also use Zooma → Flytta systemet and drag the coordinate system around with the mouse.

Fråga: How can I easily zoom in and out?

A: When you hold down the **Ctrl** key you can use the + and - keys to zoom in and out. The scroll wheel on the mouse can be used for zooming at the position of the mouse pointer. When you move the scroll wheel up the program will zoom into the coordinate system and center the graphing area at the position of the mouse pointer. When you move the scroll wheel down the program zooms out.

Fråga: How do I save default settings?

A: Set the desired default settings in the Redigera koordinataxlarna dialog, and put a mark in *Spara som standard* before pressing the OK button. Next time you create a new coordinate system, the saved default settings will be used.

Fråga: Can I make the program remember the size and position of the window?

A: When you select *Spara fönsterposition vid avslut* in the Tillval dialog. Graph will save the position and size of the main window when the program quits. The next time the program starts the same size and position is used.

Fråga: Why does the program not accept a comma as decimals separator?

A: I know a lot of countries use comma to separate the decimal part from the integer part, but Graph uses comma for separating function arguments. The program always uses a period to separate decimals from the integer value, no matter your local settings.

Fråga: How do I plot a vertical line?

A: A vertical line can be drawn as a parametric function. Select *Parametric function* 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.

Fråga: How do I plot a function x=f(y)?

A: To draw a function with y as the independent variable, you need to use a parametric function. Select *Parametric function* as *Funktionstyp* when adding the function. If you want to draw the function

x=sin(y), you can now enter the function as **`x(t)=sin(t), y(t)=t`**. Alternatively you can draw it as a relation where you can enter **`x=sin(y)`** directly.

Fråga: How do I plot a circle?

A: You need to use a parametric function to draw a circle. When inserting the function, select *Parametric function* as *Funktionstyp*. You can now add a circle with radius 5 and center in (2,3) as **`x(t)=5cos(t)+2, y(t)=5sin(t)+3`**. You may need to use Zooma → Kvadratisk to make the axes equally scaled. Else the circle may look like an ellipse. A circle can also be added as a polar function, but only with center in (0,0). A circle with radius 5 may be added as the polar function **`r(t)=5`**. Alternatively you can use a relation and add the circle as **`(x-2)^2+(y-3)^2=5^2`**.

Fråga: How do I calculate the area between two functions?

A: If you want to find the area between two functions f1(x)=3x and f2(x)=x^2, the easiest way is to create a new function that is the difference between the two functions: f(x)=f1(x)-f2(x)=3x-x^2. You can then use Beräkna → Integrera to calculate the area for a given interval.

Fråga: How do I plot the inverse of any given function?

A: You can use a parametric function for this. If you want to plot the inverse of $f(x)=x^2-2x$, you can insert it as the parametric function **`x(t)=t^2-2t, y(t)=t`**.

Fråga: How can I draw the negative part of f(x)=sqrt(x+2) ?

A: For each value x, f(x) will evaluate to at most one value. **`f(x)=sqrt(x+2)`** will therefore only have positive values of f(x). To plot it for negative f(x) too, you will have to create two separate functions: **`f(x)=sqrt(x+2)`** and **`f(x)=-sqrt(x+2)`**. Alternatively you can plot it as the relation: **`y^2=x+2`**.

Fråga: How do I plot a complex function like f(t)=e^(i*t)?

A: You probably want to show the real part on the x-axis and the imaginary part on the y-axis. In that case you can draw the function as the parametric function **`x(t)=re(e^(i*t)), y(t)=im(e^(i*t))`**. Notice that *Beräkna med komplexa tal* must be enabled in the Redigera koordinataxlarna dialog.

Fråga: How can I make Graph plot functions with vertical asymptotes correctly?

A: Functions like **`f(x)=tan(x)`** with vertical asymptotes may not always be shown correctly. As default Graph will evaluate the function for each pixel on the x-axis. But if the graph has a steep slope that goes against infinite and back between two pixels, Graph will not notice it. To plot the function correctly you can tell Graph how many evaluations to perform. This may be entered in the *Steg* field in the Infoga funktion dialog. A number around 100000 will usually show the function correctly.

Fråga: How to create a PDF file from Graph?

A: You can choose to save as PDF in the Spara som bild dialog.

Fråga: Why will the program not start under Windows 95?

A: Graph no longer supports Windows 95. The last version to run under Windows 95 was Graph 4.2.

OLE-server/-klient

OLE-server

Graph has been implemented as an OLE (Object Linking and Embedding) server, which means that Graph objects can be placed (embedded) into an OLE client. Many applications can work as OLE clients, for example Microsoft Word.

You can use Redigera → Kopiera bild in Graph to copy the current content to the clipboard. Afterwards you can select Redigera → Klistra in in Word (or similar in another OLE client) to insert the Graph object from the clipboard. When you double click on the object a new instance of Graph will start where you can edit the object. If you don't want to paste the data as a Graph object into Word, you can use Redigera → Paste Special... in Word to paste as a picture instead.

You may create a new Graph object in Word by choosing the Lägg till → Objekt... menu item and selecting *Graph-system* as *Object type*. The same dialog can be used to create an embedded Graph object from an existing grf-file. If you select *Link to file*, you will get a linked object instead of an embedded object. This way all changes to the object will be reflected in the original grf-file. If the grf-file is not available you will not be able to edit the object, but you can still see the image in Word.

To edit a Graph object you must have Graph installed on the system. If Graph is not installed you will still be able to see the image but not edit it.

OLE-klient

Graph can work as an OLE client as a text label in Graph is an OLE container. This means that you can paste images and OLE objects into the editor used to add labels. As in any other OLE container you can edit the object by double clicking on it. From the context menu you can use Infoga objekt... to create a new OLE object in the label. The same dialog can be used to create an object from a file. You can for example insert an image file this way. To edit an OLE object the server must be installed on the system, else you will only be able to see object but not edit it.

List of menu items

The following is a list of all the menu items in the program:

Arkiv → Nytt (**Ctrl+N**)
: Use this to create a new coordinate system for drawing graphs in.

Arkiv → Öppna... (**Ctrl+O**)
: Reads an earlier saved coordinate system from a .grf file.

Arkiv → Spara (**Ctrl+S**)
: Saves the coordinate system to a file.

Arkiv → Spara som...
: Saves the coordinate system to a file with a new name.

Arkiv → Spara som bild... (**Ctrl+B**)
: Sparar aktuellt koordinatsystem som en bild.

Arkiv → Importera → Graph-fil...
: Imports the contents of another Graph file into the current coordinate system.

Arkiv → Importera → Punktserie...
: Imports one or several point series from a tab, comma or semicolon separated data file. The first column shall contain the x-coordinates. The following columns shall contain the y-coordinates. Graph will create as many point series as there are columns with y-coordinates in the file. There is no limit to the number of point series possible in the data file as long as they share the same x-coordinates.

Arkiv → Skriv ut... (**Ctrl+P**)
: Sends the coordinate system and graphs to a printer.

Arkiv → Avsluta (**Alt+F4**)
: Quits the program. You may be asked to save the file.

Redigera → Ångra (**Ctrl+Z**)
: Use this to undo the last thing you did. You can choose how many undo steps that are saved in the Tillval dialog.

Redigera → Gör om (**Ctrl+Y**)
: Use this to redo the last thing undone. This is only available after you have selected Redigera → Ångra.

Redigera → Klipp ur (**Ctrl+X**)
: This will copy the selected *graph element* to the clipboard. The element will be deleted afterwards.

Redigera → Kopiera (**Ctrl+C**)
: This will copy the selected *graph element* to the clipboard.

Redigera → Klistra in (**Ctrl+V**)
: This will paste an earlier copied *graph element* from the clipboard into the coordinate system.

Redigera → Kopiera bild (**Ctrl+I**)
: Copies the shown coordinate system to the clipboard as an image. You can then paste it into another program, i.e. Microsoft Word.

Redigera → Axlar... (**Ctrl+A**)
: Edit specifications for the axes, e.g. scale, colors, legend placement, etc.

Redigera → Inställningar...
: This will change global settings for Graph, e.g. association of .grf files, showing of tooltips, maximum number of undo stored undo steps, etc.

Funktion → Lägg till funktion... (**Ins**)
: Inserts a function into the coordinate system. Functions may be added with different width and color, and you can choose to only show the graph in a specified interval, and specify other settings too.

Funktion → Insert tangent... (**F2**)
: Use this dialog to add a tangent to an already shown function at a user specified point. The tangent will be added to the function selected in the *funktionslista*.

Funktion → Skuggning... (**F3**)
: This menu item is used to add a shading to the selected function. You may choose between different styles of shading and different colors. The shading may be added above the function, below the function, between the function and the x-axis, between the function and the y-axis, inside the function or between two functions.

Funktion → Derivata f'(x)... (**F7**)
: This dialog is used to add the first derivative to the selected function.

Funktion → Punktserie... (**F4**)
: Inserts a new point series into the coordinate system. An infinite number of points defined by their x- and y-coordinates may be added. It is possible to choose color, size, and style of the point series.

Funktion → Trendlinje... (**Ctrl+T**)
: Inserts a trendline as the curve of best fit for the selected point series. You may choose between different kinds of functions for the trendline.

Funktion → Lägg till relation... (**F6**)
: This inserts an equation or inequality into the coordinate system. Equations and inequalities are used to express relations between x- and y-coordinates with the same operators etc. as for graphs of functions. Relations may be added with different shading styles and colors.

Funktion → Lägg till etikett... (**F8**)
: This will show a dialog, which may be used to create a formatted text label. The label will always be created at the center of the graphing area but can afterwards be dragged to another place with the mouse.

Funktion → Redigera... (**Enter**)
: This will show a dialog where you can change the selected *graph element* in the *funktionslista*.

Funktion → Ta bort (**Del**)
: This will delete the selected *graph element* in the *funktionslista*.

Funktion → Skapa funktioner... (**Ctrl+F**)
: This shows a dialog used to create custom functions and constants in addition to the built-in ones.

Zooma → In (**Ctrl++**)
: This will zoom in at the center of the graphing area, so you will see a ¼ of the previous graphing area.

Zooma → Ut (**Ctrl+-**)
: This will zoom out so you see 4 times as much as on the previous graphing area.

Zooma → Markering (**Ctrl+W**)
: Hold down the left mouse button while you select the area you want to fill the whole graphing area. Right click or press **Esc** to cancel the command.

Zooma → Kvadratisk (**Ctrl+Q**)
: This changes the y-axis to the same scale as the x-axis. It will make a circle look correctly instead of showing as an ellipse. The axes will stay equally scaled until disabled again.

Zooma → Standard (**Ctrl+D**)
: Returns the axes settings to the same default settings used when creating a new coordinate system.

Zooma → Flytta systemet (**Ctrl+M**)
: When selected the mouse pointer changes to a hand. You may now use the mouse to drag the coordinate system around. Select the menu item again, right click or press **Esc** to return to normal mode. As an alternative to this menu item, you may hold down the **Shift** key and drag the coordinate system around.

Zooma → Anpassa
: This will change the axes settings to show all parts of the selected *graph element*.

Zooma → Anpassa alla
: This will change the axes settings to show all parts of all the elements in the *funktionslista*.

Beräkna → Linjelängd
: Calculates the distance along the path between two points on the selected graph.

Beräkna → Integrera
: Calculates the definite integral for a specified domain range. This is the same as the signed area between the graph and the x-axis.

Beräkna → Utvärdera (**Ctrl+E**)
: This will evaluate the selected function for a given value. For standard functions f(x), f'(x) and f''(x) are evaluated. For parametric functions x(t), y(t), dx/dt, dy/dt and dy/dx are evaluated. For polar functions r(t), x(t), y(t), dr/dt and dy/dt are evaluated.

Beräkna → Tabell...
: This dialog fills a table with a user specified range of values and the result of evaluating the selected function for the values.

Beräkna → Animera...
: This dialog allows you to create an animation from the data in the coordinate system by changing an existing custom constant. This makes it easy to see what happens when the constant changes. The animation may be saved to a disk file.

Hjälp → Innehåll och register (**F1**)
: Shows the contents and index of the help file.

Hjälp → Lista funktioner (**Ctrl+F1**)
: Shows a list of functions and constants that may be used for plotting graphs.

Hjälp → Vanliga frågor
: This will show a list of frequently asked questions and their answers.

Hjälp → Dagens tips
: This will show some tips about using Graph in a more optimal way, and certain features of Graph you may not know about.

Hjälp → Internet → Graphs hemsida
: Shows the web site for Graph in your default web browser.

Hjälp → Internet → Assistans
: Shows the support forum for Graph in your default web browser.

Hjälp → Internet → Donera
: Shows the web page that allows you to donate to the Graph project to support its development.

Hjälp → Internet → Finns ny version?
: This will check if a new version of Graph is available. If there is a new version, you will be asked if you want to visit the web site of Graph to download the new version.

Hjälp → Om Graph (**Alt+F1**)
: Shows version number, copyright and license information for Graph.

Felmeddelanden

Error 01: An error occurred while evaluating power function.

This error occurs when a number raised to the power of another number resulted in an error. For example (-4)^(-5.1) gives an error, because a negative number cannot be raised to a negative non integer number when calculating with *real numbers*.

Error 02: Tangent to pi/2+n*pi (90°+n180° in degrees) is undefined.

tan(x) is undefined for x= π/2+πp = 90°+p180°, where p is an integer.

Error 03: Fact can only be calculated for positive integers.

fact(x), which calculates the factorial number of x, is only defined for positive integers of x.

Error 04: Cannot take logarithm to number equal or less than zero.

The logarithmic functions ln(x) and log(x) are undefined for x≤0, when the calculation is done for real numbers. When the calculations are done with complex numbers, x is only undefined at 0.

Error 05: sqrt is undefined for negative numbers.

sqrt(x) is undefined for x<0, when the calculations are done for real numbers. sqrt(x) is defined for all numbers, when the calculations are done with complex numbers.

Error 06: A part of the evaluation gave a number with an imaginary part.

This error may occur when calculations are done with real numbers. If a part of the calculation resulted in a number with an imaginary part, the calculation cannot continue. An example of this is: sin(x+**i**)

Error 07: Division by zero.

The program tried to divide by zero when calculating. A function is undefined for values where a division by zero is needed. For example the function f(x)=1/x is undefined at x=0.

Error 08: Inverse trigonometric function out of range [-1;1]

The inverse trigonometric functions asin(x) and acos(x) are only defined in the range [-1;1], and they are not defined for any numbers with an imaginary part. The function atan(x) is defined for all numbers without an imaginary part. This error may also happen if you are trying to take arg(0).

Error 09: The function is not defined for this value.

This error may occur for functions that are not defined for a specific point. This is for example the case for sign(x) and u(x) at x=0.

Error 10: atanh evaluated at undefined value.

Inverse hyperbolic tangent atanh(x) is undefined at x=1 and x=-1, and not defined outside the interval x=]-1;1[when calculating with real numbers only.

Error 11: acosh evaluated at undefined value.

Inverse hyperbolic cosine acosh(x) is only defined for x≥1 when using *real numbers*. acosh(x) is defined for all numbers when calculating with *complex numbers*.

Error 12: arg(0) is undefined.

The argument of zero is undefined because 0 does not have an angle.

Error 13: Evaluation failed.

This error occurs when a more complicated function like W(z) is evaluated, and the evaluation failed to find an accurate result.

Error 14: Argument produced a function result with total loss of precision.

An argument to a function call produced a result with total loss of significant digits, such as sin(1E70) which gives an arbitrary number in the range [-1;1].

Error 15: The custom function/constant '%s' was not found or has the wrong number of arguments.
A custom function or constant no longer exists. You can either define it again or remove all uses of the symbol. This may also happen if a custom constant has been changed to a function or vice versa, or if the number of arguments to a custom function has been changed.

Error 16: Too many recursive calls
Too many recursive calls have been executed. This is most likely caused by a function that calls itself recursively an infinite number of times, for example foo(x)=2*foo(x). The error may also occur if you just call too many functions recursively.

Error 17: Overflow: A function returned a value too large to handle.
A function call resulted in value too large to handle. This for example happens if you try to evaluate sinh(20000)

Error 18: A plugin function failed.
A custom function in a Python plugin did not return a result. The Python interpreter window may show more detailed information.

Error 50: Unexpected operator. Operator %s cannot be placed here
An operator +, -, *, / or ^ was misplaced. This can happen if you try entering the function f(x)=^2, and it usually means that you forgot something in front of the operator.

Error 55: Right bracket missing.
A right bracket is missing. Make sure you have the same number of left and right brackets.

Error 56: Invalid number of arguments supplied for the function '%s'
You passed a wrong number of arguments to the specified function. Check the Lista funktioner to find the required number of arguments the function needs. This error may occur if you for example write sin(x,3).

Error 57: Comparison operator misplaced.
Only two comparison operators in sequence are allowed. For example "sin(x) < y < cos(x)" is okay while "sin(x) < x < y < cos(x)" is invalid because there are three <-operators in sequence.

Error 58: Invalid number found. Use the format: -5.475E-8
Something that looked like a number but wasn't has been found. For example this is an invalid number: 4.5E. A number should be on the form nnn.fffEeee where nnn is the whole number part that may be negative. fff is the fraction part that is separated from the integer part with a dot '.'. The fraction part is optional, but either the integer part or the fraction part must be there. E is the exponent separator and must be an 'E' in upper case. eee is the exponent optionally preceded by '-'. The exponent is only needed if the E is there. Notice that 5E8 is the same as 5*10^8. Here are some examples of numbers:
`-5.475E-8, -0.55, .75, 23E4`

Error 59: String is empty. You need to enter a formula.
You didn't enter anything in the box. This is not allowed. You need to enter an expression.

Error 60: Comma is not allowed here. Use dot as decimal separator.
Commas may not be used as decimal separator. You have to use a '.' to separate the fraction from the integer part.

Error 61: Unexpected right bracket.
A right bracket was found unexpectedly. Make sure the number of left and right brackets match.

Error 63: Number, constant or function expected.
A factor, which may be a number, constant, variable or function, was expected.

Error 64: Parameter after constant or variable not allowed.
Brackets may not be placed after a constant or variable. For example this is invalid: f(x)=x(5). Use f(x)=x*5 instead.

Error 65: Expression expected.
An expression was expected. This may happen if you have empty brackets: f(x)=sin()

Error 66: Unknown variable, function or constant: %s
You entered something that looks like a variable, function or constant but is unknown. Note that "x5" is not the same as "x*5".

Error 67: Unknown character: %s
An unknown character was found.

Error 68: The end of the expression was unexpected.
The end of the expression was found unexpectedly.

Error 70: Error parsing expression
An error happened while parsing the function text. The string is not a valid function.

Error 71: A calculation resulted in an overflow.
An overflow occurred under the calculation. This may happen when the numbers get too big.

Error 73: An invalid value was used in the calculation.
An invalid value was used as data for the calculation.

Error 74: Not enough points for calculation.
Not enough data points were provided to calculate the trendline. A polynomial needs at least one more point than the degree of the polynomial. A polynomial of third degree needs at least 4 points. All other functions need at least two points.

Error 75: Illegal name %s for user defined function or constant.
Names for user defined functions and constants must start with a letter and only contain letters and decimal digits. You cannot use names that are already used by built-in functions and constants.

Error 76: Cannot differentiate recursive function.
It is not possible to differentiate a recursive function because the resulting function will be infinitely large.

Error 79: Function %s cannot be differentiated.
The function cannot be differentiated, because some part of the function does not have a first derivative. This is for example the case for arg(x), conj(x), re(x) and im(x).

Error 86: Not further specified error occurred under calculation.
An error occurred while calculating. The exact cause is unknown. If you get this error, you may try to contact the programmer with a description of how to reproduce the error. Then he might be able to improve the error message or prevent the error from occurring.

Error 87: No solution found. Try another guess or another model.
The given guess, which may be the default one, did not give any solution. This can be caused by a bad guess, and a better guess may result in a solution. It can also be because the given trendline model doesn't fit the data, in which case you should try another model.

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 numerical integral produced a result with a too high estimated error.

Error 99: Internal error. Please notify the programmer with as much information as possible.
An internal error happened. This means that the program has done something that is impossible but happened anyway. Please contact the programmer with as much information as necessary to reproduce the problem.

Funktioner

Lista funktioner

The following is a list of all variables, constants, operators and functions supported by the program. The list of operators shows the operators with the highest precedence first. The precedence of operators can be changed through the use of brackets. (), {} and [] may all be used alike. Notice that expressions in Graph are case insensitive, i.e. there are no difference between upper and lower case characters. The only exception is `e` as Euler's constant and E as the exponent in a *number* in scientific notation.

Konstant	Beskrivning
`x`	The independent variable used in standard functions.
`t`	The independent variable called parameter for parametric functions and polar angle for polar functions.
`e`	Euler's constant. In this program defined as e=2.71828182845904523536028 7
`pi`	The constant π, which in this program is defined as pi=3.14159265358979323846264 3
`undef`	Always returns an error. Used to indicate that part of a function is undefined.
`i`	The imaginary unit. Defined as $\mathbf{i}^2 = -1$. Only useful when working with complex numbers.
`inf`	The constant for infinity. Only useful as argument to the `integrate` function.
rand	Evaluates to a random number between 0 and 1.

Operator	Beskrivning
Exponentiation (^)	Raise to the power of an exponent. Example: f(x)=2^x
Negation (-)	The negative value of a factor. Example: f(x)=-x
Logical NOT (not)	`not` *a* evaluates to 1 if *a* is zero, and evaluates to 0 otherwise.
Multiplication (*)	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
Subtraction (-)	Subtracts two terms. Example: f(x)=2-x
Greater than (>)	Indicates if an expression is greater than another expression.
Greater than or equal to (>=)	Indicates if an expression is greater or equal to another expression.
Less than (<)	Indicates if an expression is less than another expression.
Less than or equal to (<=)	Indicates if an expression is less or equal to another expression.
Equal (=)	Indicates if two expressions evaluate to the exact same value.
Not equal (<>)	Indicates if two expressions does not evaluate to the exact same value.
Logical AND (and)	*a* `and` *b* evaluates to 1 if both *a* and *b* are non-zero, and evaluates to 0 otherwise.
Logical OR (or)	*a* `or` *b* evaluates to 1 if either *a* or *b* are non-zero, and evaluates to 0 otherwise.
Logical XOR (xor)	*a* `xor` *b* evaluates to 1 if either *a* or *b*, but not both, are non-zero, and evaluates to 0 otherwise.

Funktion	Beskrivning
Trigonometrisk	
sin	Returns the sine of the argument, which may be in radians or degrees.
cos	Returns the cosine of the argument, which may be in radians or degrees.
tan	Returns the tangent of the argument, which may be in radians or degrees.
asin	Returns the inverse sine of the argument. The returned value may be in radians or degrees.
acos	Returns the inverse cosine of the argument. The returned value may be in radians or degrees.
atan	Returns the inverse tangent of the argument. The returned value may be in radians or degrees.
sec	Returns the secant of the argument, which may be in radians or degrees.
csc	Returns the cosecant of the argument, which may be in radians or degrees.
cot	Returns the cotangent of the argument, which may be in radians or degrees.
asec	Returns the inverse secant of the argument. The returned value may be in radians or degrees.
acsc	Returns the inverse cosecant of the argument. The returned value may be in radians or degrees.
acot	Returns the inverse cotangent of the argument. The returned value may be in radians or degrees.
Hyperbolic	
sinh	Returns the hyperbolic sine of the argument.
cosh	Returns the hyperbolic cosine of the argument.
tanh	Returns the hyperbolic tangent of the argument.
asinh	Returns the inverse hyperbolic sine of the argument.
acosh	Returns the inverse hyperbolic cosine of the argument.
atanh	Returns the inverse hyperbolic tangent of the argument.
csch	Returns the hyperbolic cosecant of the argument.
sech	Returns the hyperbolic secant of the argument.
coth	Returns the hyperbolic cotangent of the argument.
acsch	Returns the inverse hyperbolic cosecant of the argument.
asech	Returns the inverse hyperbolic secant of the argument.
acoth	Returns the inverse hyperbolic cotangent of the argument.
Potens och logaritm	
sqr	Returns the square of the argument, i.e. the power of two.
exp	Returns e raised to the power of the argument.
sqrt	Returns the square root of the argument.
root	Returns the n^{th} root of the argument.
ln	Returns the logarithm with base e to the argument.
log	Returns the logarithm with base 10 to the argument.
logb	Returns the logarithm with base n to the argument.
Complex	
abs	Returns the absolute value of the argument.
arg	Returns the angle of the argument in radians or degrees.
conj	Returns the conjugate of the argument.

Funktion	Beskrivning
re	Returns the real part of the argument.
im	Returns the imaginary part of the argument.
Rounding	
trunc	Returns the integer part of the argument.
fract	Returns the fractional part of the argument.
ceil	Rounds the argument up to nearest integer.
floor	Rounds the argument down to the nearest integer.
round	Rounds the first argument to the number of decimals given by the second argument.
Piecewise	
sign	Returns the sign of the argument: 1 if the argument is greater than 0, and -1 if the argument is less than 0.
u	Unit step: Returns 1 if the argument is greater than or equal 0, and 0 otherwise.
min	Returns the smallest of the arguments.
max	Returns the greatest of the arguments.
range	Returns the second argument if it is in the range of the first and third argument.
if	Returns the second argument if the first argument does not evaluate to 0; Else the third argument is returned.
Special	
integrate	Returns the numeric integral of the first argument from the second argument to the third argument.
sum	Returns the sum of the first argument evaluated for each integer in the range from the second to the third argument.
product	Returns the product of the first argument evaluated for each integer in the range from the second to the third argument.
fact	Returns the factorial of the argument.
gamma	Returns the Euler gamma function of the argument.
beta	Returns the beta function evaluated for the arguments.
W	Returns the Lambert W-function evaluated for the argument.
zeta	Returns the Riemann Zeta function evaluated for the argument.
mod	Returns the remainder of the first argument divided by the second argument.
dnorm	Returns the normal distribution of the first argument with optional mean value and standard deviation.

Notice the following relations:

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)

Konstanter

rand constant

Returns a random number in the range 0 to 1.

Syntax

rand

Beskrivning

`rand` is used as a constant but returns a new pseudo-random number each time it is evaluated. The value is a real number in the range [0;1].

Anmärkningar

Because `rand` returns a new value each time it is evaluated, a graph using `rand` will not look the same each time it is drawn. A graph using `rand` will also change when the program is forced to redraw, e.g. because the coordinate system is moved, resized or zoomed.

implementering

`rand` uses a multiplicative congruential random number generator with period 2 to the 32^{nd} power to return successive pseudo-random numbers in the range from 0 to 1.

Se även

Wikipedia [http://en.wikipedia.org/wiki/Random_number_generator#Computational_methods]
MathWorld [http://mathworld.wolfram.com/RandomNumber.html]

Trigonometrisk

sin-funktion

Returns the sine of the argument.

Syntax

sin(z)

Beskrivning

The `sin` function calculates the sine of an angle *z*, which may be in *radianer* or degrees depending on the current settings. *z* may be any *numeric expression* that evaluates to a *real number* or a *complex number*. If *z* is a real number, the result will be in the range -1 to 1.

Anmärkningar

For arguments with a large magnitude, the function will begin to lose precision.

Se även

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

cos-funktion

Returns the cosine of the argument.

Syntax

cos(z)

Beskrivning

The `cos` function calculates the cosine of an angle *z*, which may be in *radianer* or degrees depending on the current settings. *z* may be any *numeric expression* that evaluates to a *real number* or a *complex number*. If *z* is a real number, the result will be in the range -1 to 1.

Anmärkningar

For arguments with a large magnitude, the function will begin to lose precision.

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

tan-funktion
Returns the tangent of the argument.

Syntax
tan(z)

Beskrivning
The `tan` function calculates the tangent of an angle *z*, which may be in *radianer* or degrees depending on the current settings. *z* may be any *numeric expression* that evaluates to a *real number* or a *complex number*.

Anmärkningar
For arguments with a large magnitude, the function will begin to lose precision. `tan` is undefined at z = p*π/2, where p is an *heltal*, but the function returns a very large number if *z* is near the undefined value.

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

asin-funktion
Returns the inverse sine of the argument.

Syntax
asin(z)

Beskrivning
The `asin` function calculates the inverse sine of *z*. The result may be in *radianer* or degrees depending on the current settings. *z* may be any numeric expression that evaluates to a *real number*. This is the reverse of the `sin` function.

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

acos-funktion
Returns the inverse cosine of the argument.

Syntax
acos(z)

Beskrivning
The `acos` function calculates the inverse cosine of *z*. The result may be in *radianer* or degrees depending on the current settings. *z* may be any *numeric expression* that evaluates to a *real number*. This is the reverse of the `cos` function.

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

atan-funktion
Returns the inverse tangent of the argument.

Syntax
atan(z)

Beskrivning

The `atan` function calculates the inverse tangent of *z*. The result may be in *radianer* or degrees depending on the current settings. *z* may be any *numeric expression* that evaluates to a *real number*. This is the reverse of the `tan` function.

Se även

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

sec-funktion

Returns the secant of the argument.

Syntax

sec(z)

Beskrivning

The `sec` function calculates the secant of an angle *z*, which may be in *radianer* or degrees depending on the current settings. sec(z) is the same as 1/cos(z). *z* may be any *numeric expression* that evaluates to a *real number* or a *complex number*.

Anmärkningar

For arguments with a large magnitude, the function will begin to lose precision.

Se även

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

csc-funktion

Returns the cosecant of the argument.

Syntax

csc(z)

Beskrivning

The `csc` function calculates the cosecant of an angle *z*, which may be in *radianer* or degrees depending on the current settings. csc(z) is the same as 1/sin(z). *z* may be any *numeric expression* that evaluates to a *real number* or a *complex number*.

Anmärkningar

For arguments with a large magnitude, the function will begin to lose precision.

Se även

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

cot-funktion

Returns the cotangent of the argument.

Syntax

cot(z)

Beskrivning

The `cot` function calculates the cotangent of an angle *z*, which may be in *radianer* or degrees depending on the current settings. cot(z) is the same as 1/tan(z). *z* may be any numeric expression that evaluates to a *real number* or a *complex number*.

Anmärkningar

For arguments with a large magnitude, the function will begin to lose precision.

Se även

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

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

asec-funktion

Returns the inverse secant of the argument.

Syntax

asec(z)

Beskrivning

The `asec` function calculates the inverse secant of *z*. The result may be in *radianer* or degrees depending on the current settings. asec(z) is the same as acos(1/z). *z* may be any *numeric expression* that evaluates to a *real number*. This is the reverse of the `sec` function.

Se även

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

acsc-funktion

Returns the inverse cosecant of the argument.

Syntax

acsc(z)

Beskrivning

The `acsc` function calculates the inverse cosecant of *z*. The result may be in *radianer* or degrees depending on the current settings. acsc(z) is the same as asin(1/z). *z* may be any *numeric expression* that evaluates to a *real number*. This is the reverse of the `csc` function.

Se även

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

acot-funktion

Returns the inverse cotangent of the argument.

Syntax

acot(z)

Beskrivning

The `acot` function calculates the inverse cotangent of *z*. The result may be in *radianer* or degrees depending on the current settings. acot(z) is the same as atan(1/z). *z* may be any *numeric expression* that evaluates to a *real number*. This is the reverse of the `cot` function.

Anmärkningar

The `acot` function returns a value in the range]-π/2;π/2] (]-90;90] when calculating in degrees), which is the most common definition, though some may define it to be in the range]0;π[.

Se även

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

Hyperbolic

sinh-funktion

Returns the hyperbolic sine of the argument.

Syntax

sinh(z)

Beskrivning

The `sinh` function calculates the hyperbolic sine of *`z`*. *`z`* may be any *numeric expression* that evaluates to a *real number* or a *complex number*.

Hyperbolic sine is defined as: sinh(z) = ½(e^{z}-e^{-z})

Se även

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

cosh-funktion

Returns the hyperbolic cosine of the argument.

Syntax

cosh(z)

Beskrivning

The `cosh` function calculates the hyperbolic cosine of *`z`*. *`z`* may be any *numeric expression* that evaluates to a *real number* or a *complex number*.

Hyperbolic cosine is defined as: cosh(z) = ½(e^{z}+e^{-z})

Se även

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

tanh-funktion

Returns the hyperbolic tangent of the argument.

Syntax

tanh(z)

Beskrivning

The `tanh` function calculates the hyperbolic tangent of *`z`*. *`z`* may be any *numeric expression* that evaluates to a *real number* or a *complex number*.

Hyperbolic tangent is defined as: tanh(z) = sinh(z)/cosh(z)

Se även

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

asinh-funktion

Returns the inverse hyperbolic sine of the argument.

Syntax

asinh(z)

Beskrivning

The `asinh` function calculates the inverse hyperbolic sine of *`z`*. *`z`* may be any *numeric expression* that evaluates to a *real number* or a *complex number*. `asinh` is the reverse of `sinh`, i.e. asinh(sinh(z)) = z.

Se även

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

acosh-funktion

Returns the inverse hyperbolic cosine of the argument.

Syntax

acosh(z)

Beskrivning

The `acosh` function calculates the inverse hyperbolic cosine of *`z`*. *`z`* may be any *numeric expression* that evaluates to a *real number* or a *complex number*. `acosh` is the reverse of `cosh`, i.e. acosh(cosh(z)) = z.

Se även

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

atanh-funktion

Returns the inverse hyperbolic tangent of the argument.

Syntax

atanh(z)

Beskrivning

The `atanh` function calculates the inverse hyperbolic tangent of *`z`*. *`z`* may be any *numeric expression* that evaluates to a *real number* or a *complex number*. `atanh` is the reverse of `tanh`, i.e. atanh(tanh(z)) = z.

Se även

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

csch-funktion

Returns the hyperbolic cosecant of the argument.

Syntax

csch(z)

Beskrivning

The `csch` function calculates the hyperbolic cosecant of *`z`*. *`z`* may be any *numeric expression* that evaluates to a *real number* or a *complex number*.

Hyperbolic cosecant is defined as: csch(z) = 1/sinh(z) = $2/(e^{z}-e^{-z})$

Se även

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

sech-funktion

Returns the hyperbolic secant of the argument.

Syntax

sech(z)

Beskrivning

The `sech` function calculates the hyperbolic secant of *`z`*. *`z`* may be any *numeric expression* that evaluates to a *real number* or a *complex number*.

Hyperbolic secant is defined as: sech(z) = 1/cosh(z) = $2/(e^{z}+e^{-z})$

Se även

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

coth-funktion

Returns the hyperbolic cotangent of the argument.

Syntax

coth(z)

Beskrivning

The `coth` function calculates the hyperbolic cotangent of *`z`*. *`z`* may be any *numeric expression* that evaluates to a *real number* or a *complex number*.

Hyperbolic cotangent is defined as: coth(z) = 1/tanh(z) = cosh(z)/sinh(z) = $(e^{z} + e^{-z})/(e^{z} - e^{-z})$

Se även

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

acsch-funktion

Returns the inverse hyperbolic cosecant of the argument.

Syntax

acsch(z)

Beskrivning

The `acsch` function calculates the inverse hyperbolic cosecant of *`z`*. *`z`* may be any *numeric expression* that evaluates to a *real number* or a *complex number*. `acsch` is the reverse of `csch`, i.e. acsch(csch(z)) = z.

Se även

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

asech-funktion

Returns the inverse hyperbolic secant of the argument.

Syntax

asech(z)

Beskrivning

The `asech` function calculates the inverse hyperbolic secant of *`z`*. *`z`* may be any *numeric expression* that evaluates to a *real number* or a *complex number*. `asech` is the reverse of `sech`, i.e. asech(sech(z)) = z.

Se även

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

acoth-funktion

Returns the inverse hyperbolic cotangent of the argument.

Syntax

acoth(z)

Beskrivning

The `acoth` function calculates the inverse hyperbolic cotangent of *`z`*. *`z`* may be any *numeric expression* that evaluates to a *real number* or a *complex number*. `acoth` is the reverse of `coth`, i.e. acoth(coth(z)) = z. For real numbers `acoth` is undefined in the interval [-1;1].

Se även

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

Dignitet och logaritm

sqr-funktion

Returns the square of the argument.

Syntax

sqr(z)

Beskrivning

The `sqr` function calculates the square of z, i.e. z raised to the power of 2. z may be any *numeric expression* that evaluates to a *real number* or a *complex number*.

exp-funktion

Returns *e* raised to the power of the argument.

Syntax

exp(z)

Beskrivning

The `exp` function is used to raise *e*, Euler's constant, to the power of z. This is the same as e^z. z may be any *numeric expression* that evaluates to a *real number* or a *complex number*.

Se även

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

sqrt-funktion

Returns the square root of the argument.

Syntax

sqrt(z)

Beskrivning

The `sqrt` function calculates the square root of z, i.e. z raised to the power of ½. z may be any *numeric expression* that evaluates to a *real number* or a *complex number*. If the calculation is done with real numbers, the argument is only defined for $z \geq 0$.

Se även

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

root-funktion

Returns the n^{th} root of the argument.

Syntax

root(n, z)

Beskrivning

The `root` function calculates the n^{th} root of z. n and z may be any *numeric expression* that evaluates to a *real number* or a *complex number*. If the calculation is done with real numbers, the argument is only defined for $z \geq 0$.

Anmärkningar

When the calculation is done with real numbers, the function is only defined for $z<0$ if n is an odd *heltal*. For calculations with complex numbers, `root` is defined for the whole complex plane except at the pole $n=0$. Notice that for calculations with complex numbers the result will always have an imaginary part when $z<0$ even though the result is real when calculations are done with real numbers and n is an odd integer.

Exempel

Istället för x^(1/3) kan du använda root(3, x).

Se även

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

ln-funktion

Returns the natural logarithm of the argument.

Syntax
ln(z)

Beskrivning
The `ln` function calculates the logarithm of *z* with base *e*, which is Euler's constant. ln(z) is commonly known as the natural logarithm. *z* may be any *numeric expression* that evaluates to a *real number* or a *complex number*. If the calculation is done with real numbers, the argument is only defined for *z*>0. When calculating with complex numbers, *z* is defined for all numbers except *z*=0.

Se även
Wikipedia [http://en.wikipedia.org/wiki/Natural_logarithm]
MathWorld [http://mathworld.wolfram.com/NaturalLogarithm.html]

log-funktion
Returns the base 10 logarithm of the argument.

Syntax
log(z)

Beskrivning
The `log` function calculates the logarithm of *z* with base 10. *z* may be any *numeric expression* that evaluates to a *real number* or a *complex number*. If the calculation is done with real numbers, the argument is only defined for *z*>0. When calculating with complex numbers, *z* is defined for all numbers except *z*=0.

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

logb-funktion
Returns the base *n* logarithm of the argument.

Syntax
logb(z, n)

Beskrivning
The `logb` function calculates the logarithm of *z* with base *n*. *z* may be any *numeric expression* that evaluates to a *real number* or a *complex number*. If the calculation is done with real numbers, the argument is only defined for *z*>0. When calculating with complex numbers, *z* is defined for all numbers except *z*=0. *n* must evaluate to a positive real number.

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

Complex

abs-funktion
Returns the absolute value of the argument.

Syntax
abs(z)

Beskrivning
The `abs` function returns the absolute or numeric value of *z*, commonly written as |z|. *z* may be any *numeric expression* that evaluates to a *real number* or a *complex number*. abs(z) always returns a positive real value.

Se även
Wikipedia [http://en.wikipedia.org/wiki/Absolute_value]

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

arg-funktion

Returns the argument of the parameter.

Syntax

arg(z)

Beskrivning

The `arg` function returns the argument or angle of *z*. *z* may be any *numeric expression* that evaluates to a *real number* or a *complex number*. arg(z) always returns a real number. The result may be in *radianer* or degrees depending on the current settings. The angle is always between -π and π. If *z* is a real number, arg(z) is 0 for positive numbers and π for negative numbers. arg(0) is undefined.

Se även

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

conj-funktion

Returns the conjugate of the argument.

Syntax

conj(z)

Beskrivning

The `conj` function returns the conjugate of *z*. *z* may be any *numeric expression* that evaluates to a *real number* or a *complex number*. The function is defined as: conj(z) = re(z) - **i***im(z).

Se även

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

re-funktion

Returns the real part of the argument.

Syntax

re(z)

Beskrivning

The `re` function returns the real part of *z*. *z* may be any *numeric expression* that evaluates to a *real number* or a *complex number*.

Se även

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

im-funktion

Returns the imaginary part of the argument.

Syntax

im(z)

Beskrivning

The `im` function returns the imaginary part of *z*. *z* may be any *numeric expression* that evaluates to a *real number* or a *complex number*.

Se även

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

Rounding

trunc-funktion

Removes the fractional part of the argument.

Syntax

trunc(z)

Beskrivning

The `trunc` function returns the *heltal* part of `z`. The function removes the decimal part of `z`, i.e. rounds against zero. `z` may be any *numeric expression* that evaluates to a *real number* or a *complex number*. If `z` is a complex number, the function returns trunc(re(z))+trunc(im(z))**i**.

Se även

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

fract-funktion

Returns the fractional part of the argument.

Syntax

fract(z)

Beskrivning

The `fract` function returns the fractional part of `z`. The function removes the *heltal* part of `z`, i.e. fract(z) = z - trunc(z). `z` may be any *numeric expression* that evaluates to a *real number* or a *complex number*. If `z` is a complex number, the function returns fract(re(z))+fract(im(z))**i**.

Se även

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

ceil-funktion

Rounds the argument up.

Syntax

ceil(z)

Beskrivning

The `ceil` function finds the smallest *heltal* not less than `z`. `z` may be any *numeric expression* that evaluates to a *real number* or a *complex number*. If `z` is a complex number, the function returns ceil(re(z))+ceil(im(z))**i**.

Se även

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

floor-funktion

Rounds the argument down.

Syntax

floor(z)

Beskrivning

The `floor` function, which is also called the greatest integer function, gives the largest *heltal* not greater than `z`. `z` may be any *numeric expression* that evaluates to a *real number* or a *complex number*. If `z` is a complex number, the function returns floor(re(z))+floor(im(z))**i**.

Se även

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

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

round-funktion

Rounds a number to the specified number of decimals.

Syntax

round(z,n)

Beskrivning

The `round` function rounds z to the number of decimals given by n. z may be any *numeric expression* that evaluates to a *real number* or a *complex number*. If z is a complex number, the function returns round(re(z),n)+round(im(z),n)**i**. n may be any numeric expression that evaluates to an *heltal*. If $n<0$, z is rounded to n places to the left of the decimal point.

Exempel

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

Se även

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

Piecewise

sign-funktion

Returns the sign of the argument.

Syntax

sign(z)

Beskrivning

The `sign` function, which is also called signum, returns the sign of z. z may be any *numeric expression* that evaluates to a *real number* or a *complex number*. When z is a real number, sign(z) returns 1 for $z>0$ and -1 for $z<0$. sign(z) returns 0 for $z=0$. When z evaluates to a complex number, sign(z) returns z/abs(z).

Se även

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

u-funktion

The unit step function.

Syntax

u(z)

Beskrivning

u(z) is commonly known as the unit step function. z may be any *numeric expression* that evaluates to a *real number*. The function is undefined when z has an imaginary part. u(z) returns 1 for $z\geq 0$ and 0 for $z<0$.

Se även

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

min-funktion

Finds and returns the minimum of the values passed as arguments.

Syntax
min(A,B,...)

Beskrivning
The `min` function returns the minimum value of its arguments. `min` can take any number of arguments not less than 2. The arguments may be any *numeric expressions* that evaluate to *real numbers* or *complex numbers*. If the arguments are complex numbers, the function returns min(re(A), re(B), ...) + min(im(A), im(B), ...)**i**.

max-funktion
Finds and returns the maximum of the values passed as arguments.

Syntax
max(A,B,...)

Beskrivning
The `max` function returns the maximum value of its arguments. `max` can take any number of arguments not less than 2. The arguments may be any *numeric expressions* that evaluate to *real numbers* or *complex numbers*. If the arguments are complex numbers, the function returns max(re(A), re(B), ...) + max(im(A), im(B), ...)**i**.

range-funktion
Returns the second argument if it is in the range between the first argument and the third argument.

Syntax
range(A,z,B)

Beskrivning
The `range` function returns *z*, if *z* is greater than *A* and less than *B*. If *z* < *A* then *A* is returned. If *z* > *B* then *B* is returned. The arguments may be any *numeric expressions* that evaluate to *real numbers* or *complex numbers*. The function has the same effect as max(A, min(z, B)).

if-funktion
Evaluates one or more conditions and returns a different result based on them.

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

Beskrivning
The `if` function evaluates *cond1* and if it is different from 0 then *f1* is evaluated and returned. Else *cond2* is evaluated and if it is different from 0 then *f2* is returned and so forth. If none of the conditions are true *fz* is returned. *fz* is optional and if not specified `if` returns an error if none of the conditions are true. The arguments may be any *numeric expressions* that evaluate to *real numbers* or *complex numbers*.

Special

integrate-funktion
Returns an approximation for the numerical integral of the given expression over the given range.

Syntax
integrate(f,var,a,b)

Beskrivning
The `integrate` function returns an approximation for the numerical 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 *numeric expressions* that evaluate to *real numbers* 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 an estimated relative error less than 10^{-3}.

Exempel

f(x)=integrate(t^2-7t+1, t, -3, 15) will integrate f(t)=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).

Se även

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)

Beskrivning

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 *numeric expressions* that evaluate to *integers*.

Se även

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)

Beskrivning

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 *numeric expressions* that evaluate to *integers*.

Se även

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

fact-funktion

Returns the factorial of the argument.

Syntax
fact(n)

Beskrivning
The `fact` function returns the factorial of *n*, commonly written as n!. *n* may be any *numeric expression* that evaluates to a positive *heltal*. The function is defined as fact(n)=n(n-1)(n-2)...1, and relates to the `gamma` function as fact(n)=gamma(n+1).

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

gamma-funktion
Returns the value of the Euler gamma function of the argument.

Syntax
gamma(z)

Beskrivning
The `gamma` function returns the result of the Euler gamma function of *z*, commonly written as Γ(z). *z* may be any *numeric expression* that evaluates to a *real number* or a *complex number*. The gamma function relates to the factorial function as fact(n)=gamma(n+1). The mathematical definition of the gamma function is:

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

This cannot be calculated precisely, so Graph is using the Lanczos approximation to calculate the `gamma` function.

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

beta-funktion
Returns the value of the Euler beta function evaluated for the arguments.

Syntax
beta(m, n)

Beskrivning
The `beta` function returns the result of the Euler beta function evaluated for *m* and *n*. *m* and *n* may be any *numeric expressions* that evaluate to *real numbers* or *complex numbers*. The `beta` function relates to the `gamma` function as beta(m, n) = gamma(m) * gamma(n) / gamma(m+n).

Se även
Wikipedia [http://en.wikipedia.org/wiki/Beta_function]
MathWorld [http://mathworld.wolfram.com/BetaFunction.html]

W-funktion
Returns the value of the Lambert W-function evaluated for the argument.

Syntax
W(z)

Beskrivning
The `W` function returns the result of the Lambert W-function, also known as the omega function, evaluated for *z*. *z* may be any *numeric expression* that evaluates to a *real number* or a *complex number*. The inverse of the `W` function is given by $f(W)=W*e^W$.

Anmärkningar

For real values of *z* when *z* < -1/e, the `W` function will evaluate to values with an imaginary part.

Se även

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

zeta-funktion

Returns the value of the Riemann Zeta function evaluated for the argument.

Syntax

zeta(z)

Beskrivning

The `zeta` function returns the result of the Riemann Zeta function, commonly written as ζ(s). *z* may be any *numeric expression* that evaluates to a *real number* or a *complex number*.

Anmärkningar

The `zeta` function is defined for the whole complex plane except for the pole at *z*=1.

Se även

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

mod-funktion

Returns the remainder of the first argument divided by the second argument.

Syntax

mod(m,n)

Beskrivning

Calculates *m* modulo *n*, the remainder of m/n. `mod` calculates the remainder f, where m = a*n + f for some integer a. The sign of f is always the same as the sign of *n*. When *n*=0, `mod` returns 0. *m* and *n* may be any *numeric expressions* that evaluate to *real numbers*.

Se även

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

dnorm-funktion

Returns the normal distribution of the first argument with optional mean value and standard deviation.

Syntax

dnorm(x, [μ,σ])

Beskrivning

The `dnorm` function is the probability density of the normal distribution, also called Gaussian distribution. *x* is the variate, also known as the random variable, *μ* is the mean value and σ is the standard deviation. *μ* and σ are optional and if left out the standard normal distribution is used where *μ*=0 and σ=1. *x*, *μ* and σ may be any *numeric expressions* that evaluate to *real numbers* where σ > 0. The normal distribution is defined as:

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

Se även

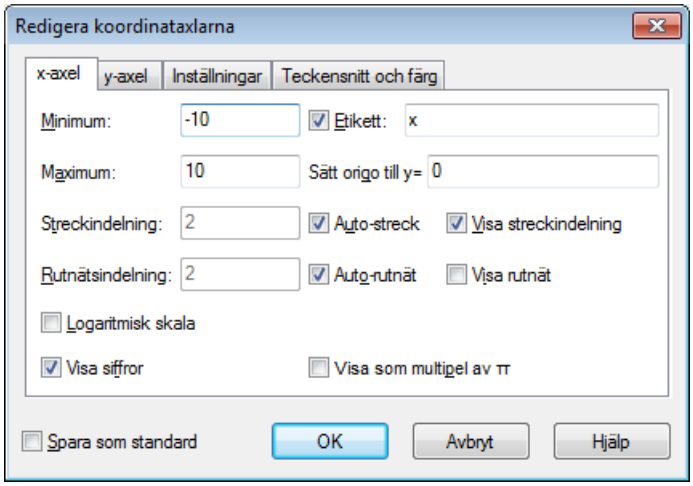
Wikipedia [http://en.wikipedia.org/wiki/Normal_distribution]
MathWorld [http://mathworld.wolfram.com/NormalDistribution.html]

Dialogs

Redigera koordinataxlarna

When you choose the menu item Redigera → Axlar..., the dialog shown below will appear. In this dialog you can configure all options which are related to the axes. The dialog contains 4 tab sheets. The first sheet, shown below, contains options for the x-axis. The tab with options for the y-axis is completely analog to this.

x-axel/y-axel



Minimum
: This is the lowest value on the selected axis. Default: -10

Maximum
: This is the highest value on the selected axis. Default: 10

Streck
: This is the distance between the tick marks on the selected axis. Tick marks are shown as small lines perpendicular to the axis. *Streck* is used for both tick marks and for showing numbers. With a logarithmic axis the *Streck* indicates the factor between the markers. For example *Streck* set to 4 will show 1, 4, 16, 64, etc. on a logarithmic axis while it will show 0, 4, 8, 12, etc. on a normal axis.

Rutnät
: This is the distance between the gridlines perpendicular to the axis. This is only used if grid lines are shown.

Logaritmisk skala
: Check this field if you want the axis to be scaled logarithmically.

Visa siffror
: When this field is checked numbers are shown on the axis with the distance chosen under *Streck*.

Etikett
: When this field is checked, the text in the edit box will be shown just above the x-axis in the right side of the coordinate system. For the y-axis, the text will be shown at the top to the right of the axis. You may use this to show what unit is used for the axes.

The x-axis crosses at / The y-axis crosses at:
: This is the coordinate where the axis will cross the other axis. This is only used when *Koordinataxelstil* is *Korsade*. Default: 0

Auto
: When checked the program will automatically choose a value for *Streck* that is fitting for the axes dimensions and size of the graphing area.

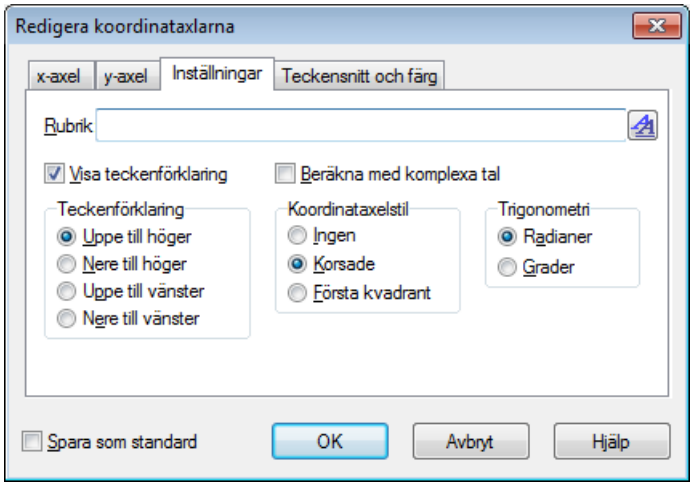
Auto
: When checked, *Rutnät* will have the same value as *Streck*.

Visa
: When this field is checked, tick marks are shown as small lines on the axis with the distance chosen under *Streck*.

Visa
: When this field is checked, grid lines will be shown as dotted lines perpendicular to the axis with the color chosen under *Teckensnitt och färg* and with the distance chosen at *Rutnät*.

Visa som multipel av π
: When this is enabled the numbers on the axis are shown as fractions multiplied by π, for example 3π/2. *Visa siffror* must be enabled for this option to be available.

Inställningar



Rubrik
: Here you can enter a title that is shown above the coordinate system. Use the button to the right to change the font.

Visa teckenförklaring
: Check this to show the *teckenförklaring* with a list of functions and point series in the upper right corner of the coordinate system. You may change the font under *Teckensnitt och färg*.

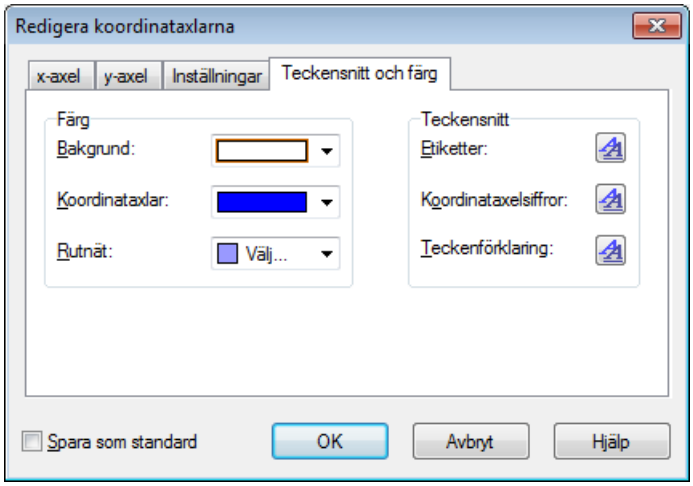
Teckenförklaring
: Here you may choose in which of the four corners you want the *teckenförklaring* to be placed. You can also change this by right clicking on the legend in the graphing area.

Beräkna med komplexa tal
: Check this field to use *complex numbers* for calculations while drawing graphs. This will increase the time for drawing graphs but may be necessary in rare situations where the intermediate result are complex. The final result must be real for the graph to be drawn. This will not interfere with evaluations.

Koordinataxelstil
: Select *Ingen* if you don't want the axes to be shown. Select *Korsade* if you want a normal coordinate system. The location of the axes may be changed from *The y-axis crosses at* and *The x-axis crosses at*. Select *Första kvadrant* if you want the axes to be shown at the bottom and left side of the coordinate system, which will overwrite *The y-axis crosses at* / *The x-axis crosses at*.

Trigonometri
: Choose whether trigonometric functions should calculate in *Radianer* or *Grader*. This is also used for showing *complex numbers* in the polar format.

Teckensnitt och färg



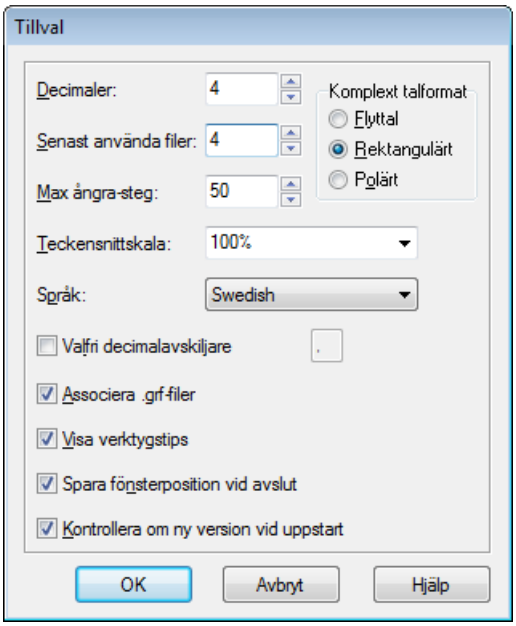
Färg
: You may change the background color, the color of the axes and the color used for drawing grid lines.

Teckensnitt
: You may change the fonts used to show axes labels, the font to draw numbers at the axes, and the font used for the *teckenförklaring*.

Spara som standard
: Check this to save all the current settings in the dialog to be used as default in the future. These settings are used the next time you choose to create a new coordinate system. The default settings are stored in your Windows user profile, i.e. each Windows user will have his/her own default settings in Graph.

Tillval

When you choose the menu item Redigera → Inställningar... the dialog shown below will appear. In this dialog, you can change general program options.



Decimaler
: This is the number of decimals that all results are shown with. The number has no influence on the calculations or the shown graphs.

Senast använda filer
: This is the maximum number of recent used files that is shown in the Arkiv menu. The number must be between 0 and 9. 0 means that no recent used files are shown.

Max ångra-steg
: Each time you make a change, the program will save enough information to undo it. As default *Max ångra-steg* are 50, which mean that you are able to undo the last 50 changes you made in the program. The undo steps will take up a small amount of memory. If your system is low on RAM, you may be able to free some memory by decreasing the *Max ångra-steg*.

Teckensnittskala
: You can use this to change the scale of the fonts and most of the user interface. This is mostly useful if your screen resolution is very high, or for some other reason, you are having difficulties reading the user interface.

Språk
: This shows a list of available languages for the program. The selected language will be the one used by the program in the future. The language can be selected differently for each user.

Valfri decimalavskiljar
: 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.

Associera .grf-filer
: A check mark in this field indicates that the file type .grf is associated with this program. The program will automatically start and load the file when you double-click on it in Explorer.

Visa verktygstips
: When there is a mark in this field, you will see a small box with an explanation when you are holding the mouse pointer over an object, like an edit field, selection box, etc., for a few seconds. The description is also shown in the status bar at the bottom of the main window.

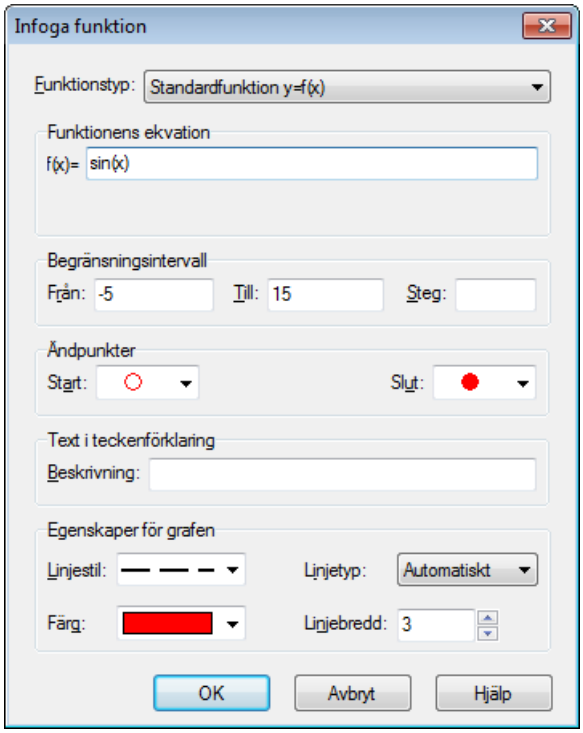
Spara fönsterposition vid avslut
: When there is a mark in this field, Graph will save the size of the main window before it quits. The next time you start the program the saved size will be used. In addition the width of the *funktionslista* is also stored. When the field doesn't have a mark the options that were last saved will be used.

Komplext talformat
: Choose how you want a complex number to be shown in the Utvärdera frame. *Flyttal* means that only *real numbers* are shown. If a number has an imaginary part then the number won't be shown and instead you get an error. *Rektangulärt* means that *complex numbers* are shown as a+b**i**, where a is the real part and b is the imaginary part. *Polärt* means that numbers are shown as a∠θ, where a is the absolute value of the number and θ is the angle of the number. θ is dependent of the choice between *Radianer* and *Grader* under *Trigonometri* in the Redigera koordinataxlarna dialog.
Notice that in some cases you may get a different result in the Utvärdera frame depending on the *Komplext talformat* setting: When *Flyttal* is chosen, Graph will try to find a real result if possible, while *Rektangulärt* and *Polärt* may give a non-real result for the same evaluation.

Kontrollera om ny version vid uppstart
: When checked each time the program starts it will check if a newer version of Graph is available on the Internet. If a new version is found you will be asked if you want to visit the web site for Graph to upgrade. If no new version is available you will not see any messages. If disabled, you can still use Hjälp → Internet → Finns ny version? to see if a new version is available.

Infoga funktion

When you want to insert a function, you use the menu item Funktion → Lägg till funktion... to show the dialog below. To edit an existing function, you select it in the *funktionslista* and use the Funktion → Redigera... menu item.



Funktionstyp

You can choose between three different types of functions: *Standardfunktion*, *parametric function* and *polar function*. A standard function is defined as y=f(x), i.e. for each x-coordinate there is exactly one y-coordinate, though it may be undefined for some x-coordinates.
For a parametric function the x- and y-coordinates are calculated from an independent variable t, called the parameter, i.e. a parametric function is defined as two functions: x(t) and y(t).

A polar function r(t) indicates an equation to calculate the distance from the origin to a point on the function given an angle t. t is the direct angle between the initial ray and the point on the function. This means that the x- and y-coordinates are given as x(t)=r(t)*cos(t), y(t)=r(t)*sin(t).

Funktionens ekvation

Here you enter the equation for the function. This can be f(x), x(t),y(t) or r(t) depending on the function type. Under Lista funktioner you can see all the available variables, constants and functions, which may be used to draw the graphs.

Begränsningsintervall

You can choose an interval for the independent variable. *Från* and *Till* indicates the start and end of the interval. If the function is a standard function, you can leave one or both of them blank to draw the graph from minus infinity to plus infinity. If the function is a parametric function or a polar function, you always have to specify an interval. If the function is a parametric or polar function, you have to specify the number of steps for which you want the function to be evaluated. When you specify a higher number of steps, the graph will appear smoother, but it will take longer to plot. It is preferred to leave the *Steg* field blank for standard functions to let Graph decide the optimal number of steps. You can however enter the number of steps if the graph doesn't show enough details, for example if an asymptote is not shown correctly. Notice that the *Steg* only specify a minimum number of calculations. Graph may add more steps at critical points if *Linjetyp* is set to *Automatiskt*.

Ändpunkter
: Here you can choose to show markers at the start and/or end of the interval. If no range is specified, the endpoints will be showed where the function enters and leaves the graphing area. The default is not to show any markers.

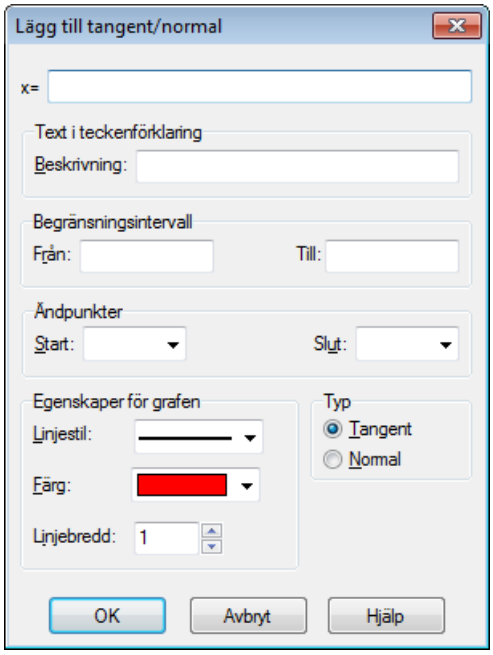
Text i teckenförklaring
: Enter a description to be shown in the *teckenförklaring*. If the text is empty, the function equation will be shown in the legend.

Egenskaper för grafen
: You can choose between different line styles for which you want the graph to be drawn. You can choose between solid, dashed, dotted or a combination of these. *Linjestil* is only available when *Linjetyp* is set to *Linjer* or *Automatiskt*. When *Linjetyp* is *Punkter*, only a dot is shown at each calculated point. Likewise the *Linjer Linjetyp* will connect the calculated points with lines. *Automatiskt* will also draw lines, but Graph will do more calculations at critical points if it thinks it will improve the graph. It will also break the line if it thinks there is an asymptote. You can also choose the width of the graph. The width is notified in screen pixels. There are also a lot of different colors you can choose between. The program will remember and suggest the same properties last used.

Lägg till tangent/normal

You can use the dialog below to insert or edit a tangent or normal to a function. To insert a new tangent or normal, you use Funktion → Lägg till tangent/normal.... To change an existing tangent or normal, you first select it in the *funktionslista* and use Funktion → Redigera....

A tangent is a straight line that touches the graph of the function at a given point without crossing it. The tangent may however cross the graph elsewhere. A normal is a straight line perpendicular to the graph of the function at a given point. If the item is a standard function the point is identified by the x-coordinate, while the point is identified from the independent t-parameter for parametric and polar functions.



Begränsningsintervall
: You can choose an interval for the tangent/normal. *Från* and *Till* indicates the start and end of the interval. You can leave one or both of them blank to draw the graph from minus infinity to plus infinity.

Ändpunkter
: Here you can choose to show markers at the start and/or end of the interval. If no interval is specified, the markers will be shown at the edge of the graphing area. The default is not to show any markers.

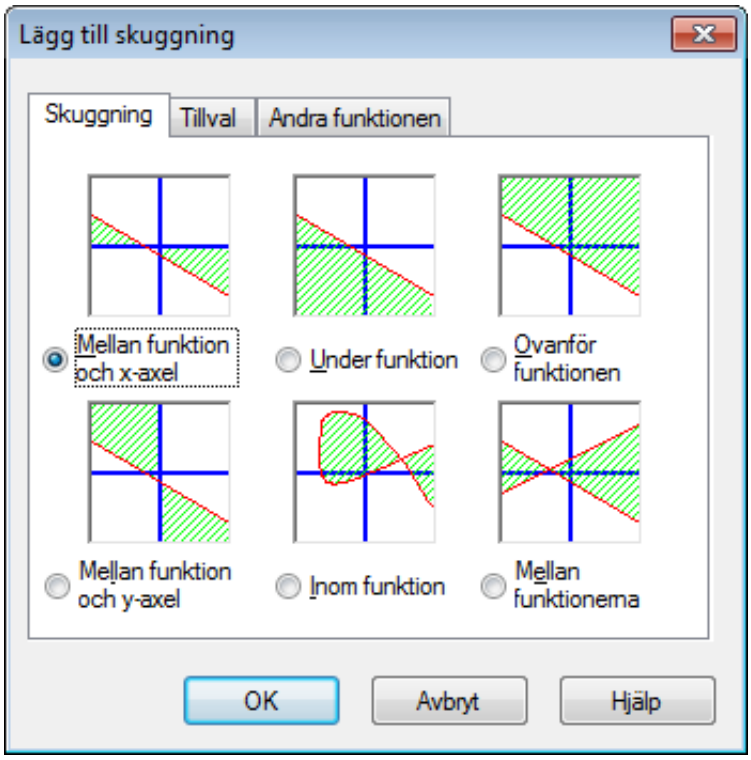
Text i teckenförklaring
: Enter a description to show in the *teckenförklaring*. If empty the function equation will be used.

Egenskaper för grafen
: You can choose between different line styles for which you want the tangent/normal to be drawn. You can choose between solid, dashed, dotted or a combination of these. You can also choose the width of the tangent/normal. The width is notified in screen pixels. There are also a lot of different colors you can choose between.

Lägg till skuggning

The dialog below is used to add a shading to the selected function. To insert a new shading, you use Funktion → Skuggning.... To change an existing shading, you first select it in the *funktionslista* and use Funktion → Redigera.... The shading is used to mark an area between the function graph and something else.

Skuggning



In the *Skuggning* tab you can choose between the following types of shadings:

Mellan funktion och x-axel
: This is the most commonly used type of shading. This will shade the area between the graph of the function and the x-axis in the selected interval. If you check *Minska till skärningspunkt* or *Öka till skärningspunkt*, the interval will decrease or increase until the graph is crossing the x-axis.

Mellan funktion och y-axel
: This will shade the area between the graph of the function and the y-axis in the selected interval. This is rarely used and probably most useful for parametric functions. Notice that you still use the x-coordinates for the interval. If you check *Minska till skärningspunkt* or *Öka till skärningspunkt*, the interval will decrease or increase until the graph is crossing the y-axis.

Under funktion
: This will shade the area below the graph of the function down to the bottom of the graphing area in the selected interval. If you check *Minska till skärningspunkt* or *Öka till skärningspunkt*, the interval will decrease or increase until the graph is crossing the bottom of the graphing area.

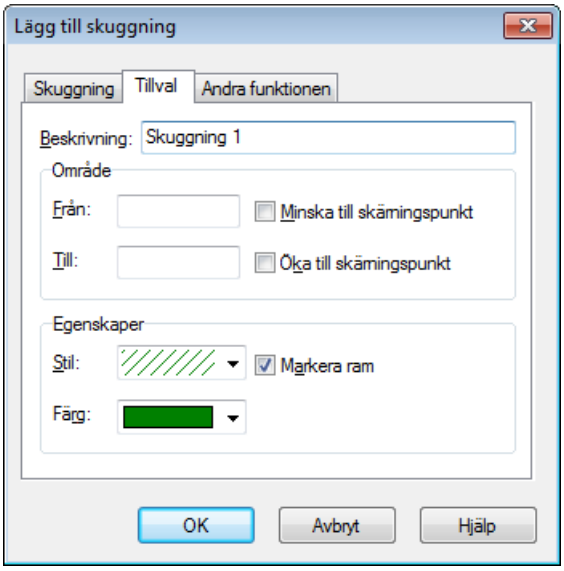
Ovanför funktionen
: This will shade the area above the graph of the function up to the top of the graphing area in the selected interval. If you check *Minska till skärningspunkt* or *Öka till skärningspunkt*, the interval will decrease or increase until the graph is crossing the top of the graphing area.

Inom funktion
: This will shade the area inside the graph of the function in the selected interval. If you check *Minska till skärningspunkt* or *Öka till skärningspunkt*, the interval will decrease or increase until the graph is crossing itself. This is especially useful to shade a closed part of a parametric or polar function, but it can also be used to shade standard functions.

Mellan funktionerna
: This will shade the area between the graphs of two functions. The first function is the one you selected in the *funktionslista* in the main window, before you invoked the dialog. The second function is selected in the list box in the *Andra funktionen* tab. For standard functions, the interval will be the same for the two functions. For parametric functions, you may select different intervals for the two functions. If you don't select an interval for the second function, it will use the same interval as the first function.

Tillval

At the *Tillval* tab shown below, you may change the options for the shading.



Från
: Here you may enter a value, for which you want the shading to start. You specify the x-coordinate if you are using a standard function or the t-parameter if you are using a parametric or polar function. If you don't enter a value, the shading will start at negative infinity. If you place a check mark in *Minska till skärningspunkt*, the start coordinate of the shading will be decreased from the entered value to the coordinate where the graph is crossing the axis, the edge of the graphing area, itself or another graph, depending of the type of shading selected.

Till
: Here you may enter a value, for which you want the shading to stop. You specify the x-coordinate if you are using a standard function or the t-parameter if you are using a parametric or polar function. If you don't enter a value, the shading will continue until positive infinity. If you place a check mark in *Öka till skärningspunkt*, the end coordinate of the shading will be increased from the entered value to the coordinate where the graph is crossing the axis, the edge of the graphing area, itself or another graph depending of the type of shading selected.

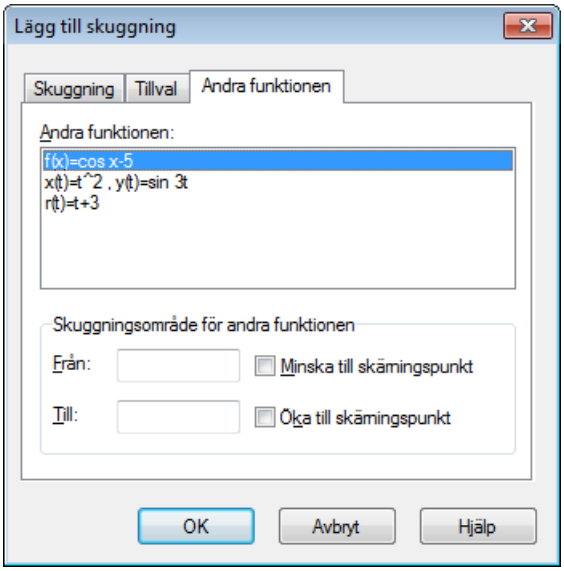
Stil
: Here you may choose between different styles to use for the shading.

Färg
: Here you may choose the color of the shading.

Markera ram
: 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.

Andra funktionen

When you have chosen *Mellan funktionerna* in the *Skuggning* tab, you may select the second function in the *Andra funktionen* tab. The dialog with the *Andra funktionen* tab is shown below.

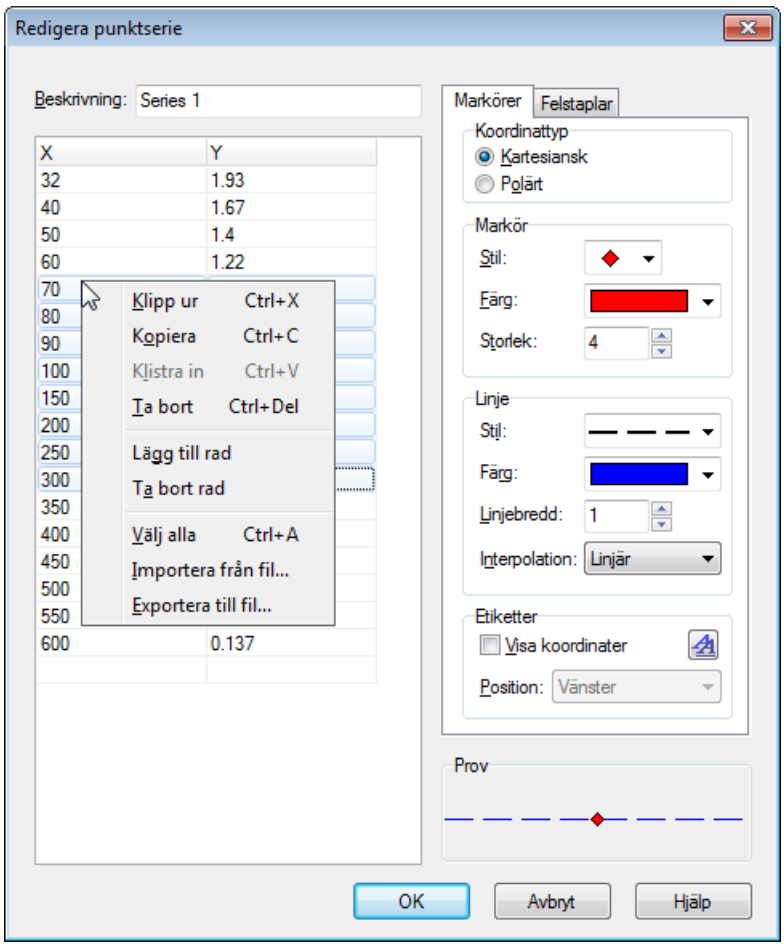


Skuggningsområde för andra funktionen
: This is used to select the interval for the second function, just like you selected the interval for the first function in the *Tillval* tab. This is only available for parametric functions and not for standard functions. For standard functions the interval for the second function is always the same as the interval for the first function. If you enter neither a start nor an end of interval for a parametric function, the values for the first function will be used for the second function as well.

Shadings are a great way to mark an area, but if you get weird results, check that you selected the right function and the right interval. If you try to shade an interval crossing an asymptote or your shading is associated with a weird parametric function, you might get weird results. But really, what did you expect?

Infoga punktserie

You can use the dialog below to add a series of points to the coordinate system. The points will be shown in the coordinate system in the graphing area as a series of markers. To insert a new point series, you use Funktion → Punktserie.... To change an existing point series, you first select it in the *funktionslista* and use Funktion → Redigera....



After adding a point series, you may add a trendline which is the curve of best fit for the points.

In the grid you can enter the x- and y-coordinates of the points. You may enter any number of points you want, but all points need both an x-coordinate and a y-coordinate.

You can select some points and use the right click menu to copy them to another program. Likewise you may copy data from other programs like MS Word or MS Excel and paste them into this the grid in dialog.

From the context menu, you can also choose to import data from a file. Graph can import text files with data separated by either tabs, commas or semicolons. The data will be placed at the position of the caret. This makes it possible to load data from more than one file, or to have x-coordinates in one file and y-coordinates in another file. In the usual case where you have all data in one file, you should make sure that the caret is located at the upper left cell before you import.

Beskrivning
: In the edit box at the top of the dialog, you can enter a name for the series, which will be shown in the *teckenförklaring*.

Koordinattyp
: You need to choose between the type of coordinates used for the points. *Kartesiansk* is used when you want to specify (x,y)-coordinates. *Polärt* 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 *radianer* or degrees depending on the current setting.

Markör
: To the right you can choose between different types of markers. The style may be a circle, a square, a triangle, etc. You may also change the color and size of the markers. If the size is set to 0, neither markers nor error bars will be shown.

Notice that if you select an arrow as marker, the arrow will be shown pointing tangential to the line at the point. The actual direction therefore depends on the *Interpolation* setting. The first point is never shown when the marker is an arrow.

Linje

It is possible to draw lines between the markers. The line will always be drawn between points in the same order they appear in the grid. You can choose between different styles, colors and widths for the lines. You can also choose to draw no line at all.
You can choose between four types of interpolation: *Linjär* will draw straight lines between the markers. *1D kubisk spline* will draw a naturlig kubisk spline [http://en.wikipedia.org/wiki/Cubic_splines], which is a nice smooth line connecting all the points sorted by the x-coordinate with 3rd degree polynomials. *2D kubisk spline* will draw a smooth cubic spline through all points in order. *Halv cosinus* 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.

Etiketter

Put a check in *Visa koordinater* 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.

Felstaplar

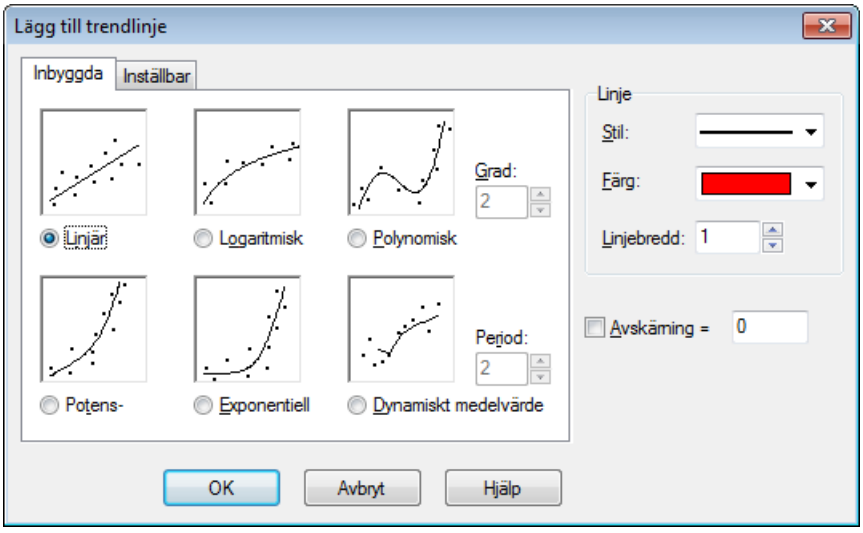
Here you can choose to show horizontal or vertical error bars, also known as uncertainty bars. They are shown as thin bars at each point in the point series indicating the uncertainty of the point. There are three ways to indicate the size of the error bars: *Fast* is used to specify that all points have the same uncertainty. *Relativ* is used to specify a percentage of the x- or y-coordinate for each point as uncertainty. *Inställbar* 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.

Lägg till trendlinje

Use the dialog shown below to insert a trendline that is the function that fits a point series best. A trendline is a function that shows a trend in a series of points, i.e. a trendline is the curve of bets fit of a specific type for a series of points. The trendline is added as an ordinary function. To create a trendline, you select the point series you want to base the trendline on and use Funktion → Trendlinje....

If the point series has custom Y-errors defined, these values are used to weight the points. The weight for each point is $1/\sigma^2$ where σ is the Y-error for the point. X-errors are not used.

Inbyggda



You may choose between the following built-in functions. These functions will give an accurate result. For *Linjär*, *Polynomisk* and *Exponentiell* trendlines, you may select the *Intercept* field and specify the point where you want the trendline to meet the y-axis.

Linjär
: This is a straight line with the function f(x) = a*x+b, where a and b are constants calculated so the line is the best fit to the point series.

  The trendline is calculated so the sum of squares (SSQ) $\Sigma(y_i-f(x_i))^2$ will be as small as possible. If possible the function will cross the points in the series; else the function will be so close to the series that the summation cannot get any smaller.

Logaritmisk
: A logarithmic line of best fit is given as f(x) = a*ln(x)+b, where a and b are constants, and ln is the natural logarithm function. To add a logarithmic function, no point in the series may have an x-coordinate that is negative or zero.
  A logarithmic function is a straight line in a semilogarithmic coordinate system. The point series is therefore converted to a semilogarithmic coordinate system and the logarithmic function with the smallest sum of squares (SSQ) is found.

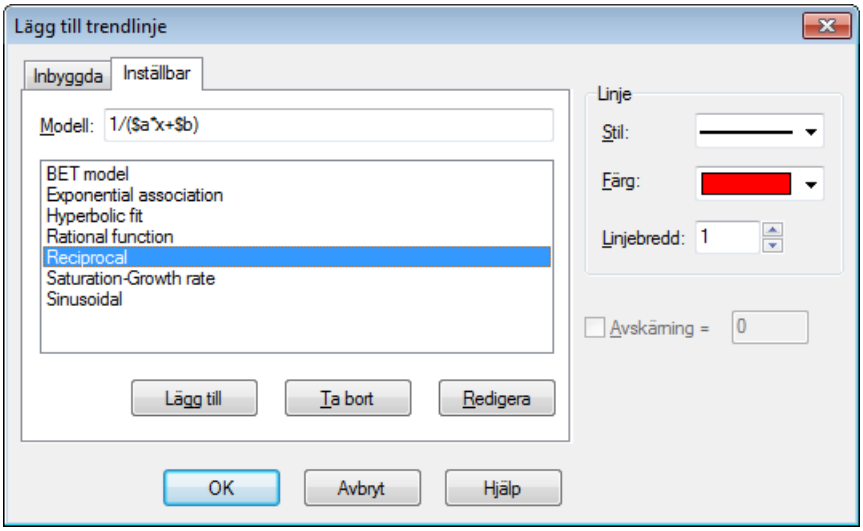
Polynomisk
: A polynomial is a function given by $f(x) = a_n*x^n + ... + a_3*x^3 + a_2*x^2 + a_1*x + a_0$, where $a_0...a_n$ are constants. n is the order of the polynomial. You need at least one more point than the order.

Potens-
: A power function is given by $f(x) = a*x^b$, where a and b are constants calculated so the function is the best fit of the point series. To add a power function, no point in the series may have an x- or y-coordinate that is negative or zero.
  A power function is a straight line in a double logarithmic coordinate system. The point series is therefore converted to a double logarithmic coordinate system and the power function with the smallest sum of squares (SSQ) is found.

Exponentiell
: An exponential function is given by $f(x) = a*b^x$, where a and b are constants calculated so the function is the best fit of the point series. To add an exponential function, no point in the series may have a y-coordinate that is negative or zero.
  An Exponential function is a straight line in a semilogarithmic coordinate system with the y-axis as the logarithmic axis. The point series is therefore converted to a semilogarithmic coordinate system and the exponential function with the smallest sum of squares (SSQ) is found.

Dynamiskt medelvärde
: Moving average is a series of straight lines based on the average of the previous points. The *Period* determines how many points are used for the average. If *Period* is 1 only one point is used, which actually isn't an average. This will draw a line directly between the points. When *Period* is larger than 1, the y-coordinate for the line at each point will not be the same as the y-coordinate of the point. Instead it will be an average of the previous points.

Inställbar



In this tab you can enter your own trendline models. The model is entered as a standard function, where all the constants that you want Graph to find are named with a $ followed by any combination of characters (a-z) and digits (0-9). Examples of valid constants are: $a, $y0, $const.

An example of a model could be f(x)=$a*x^$b+$c. The program tries to calculate the constants $a, $b and $c so that f(x) will be as close to the point series as possible. You can use the Add model button to add the model to the saved list with a name.

The program needs a guess for where to start the search for the optimum. As default the guess for all constants is 1, but this can be changed for models added to the list. A better guess will increase the possibility that an optimum can be found.

Graph will try to find the constants for the model f(x) so the sum of squares $\Sigma(y_i-f(x_i))^2$ will be the smallest possible. The program will start with the guess and move towards the minimum of the sum of squares. If a solution is not found after 100 iterations or the given guess is not valid, the program gives up.

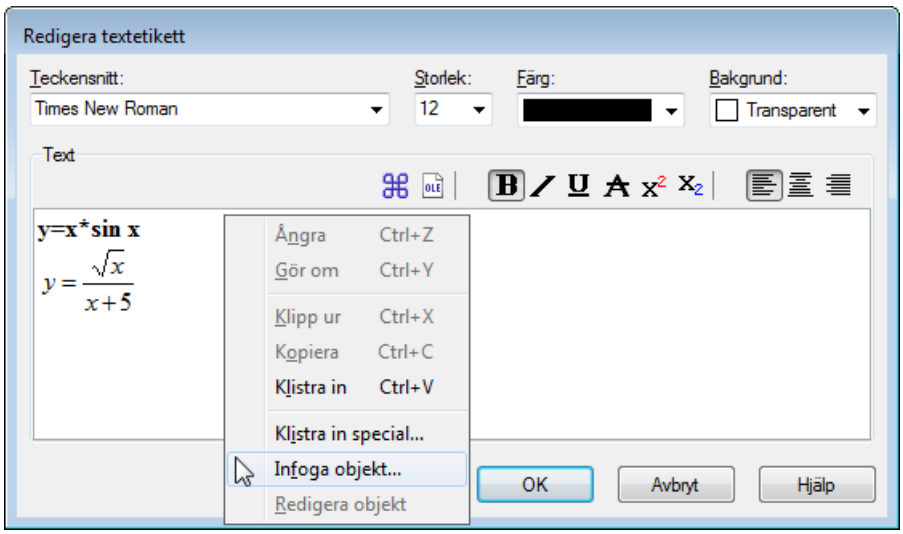
It is possible, even though it rarely happens, that more than one minimum exists. In this case the minimum nearest to the guess will be found, even though this may not be the best.

Notice that you should avoid redundant constants as they might confuse the program. For example this model has a redundant constant: f(x)=$c+$d/($a*x+$b). Notice the relation between the constants $a, $b and $d. If you multiply $a, $b and $d with the same value the resulting function will not be changed. This means that there are an infinite number of combinations of constants with the same resulting function and hence an infinite number of best solutions. This may confuse the program when it tries to find the best one. Therefore either $a, $b or $d should be removed.

When the trendline is added, the correlation coefficient R^2 is shown in the comment. The closer R^2 is to 1 the closer the trendline is to the points.

Lägg till etikett

This dialog is used to insert or edit text labels. To insert a label you use the menu item Funktion → Lägg till etikett.... The label is inserted at the center of the graphing area, but can be dragged to another placement. To change an existing label, you either double click on it in the graphing area or you select it in the *funktionslista* and use Funktion → Redigera....



The text is entered in the editing area. You can change text styles for different parts of the text. The background color, which may be any solid color or transparent, can be set for the whole label only. The ⌘ button can be used to insert special characters like mathematical symbols and Greek characters.

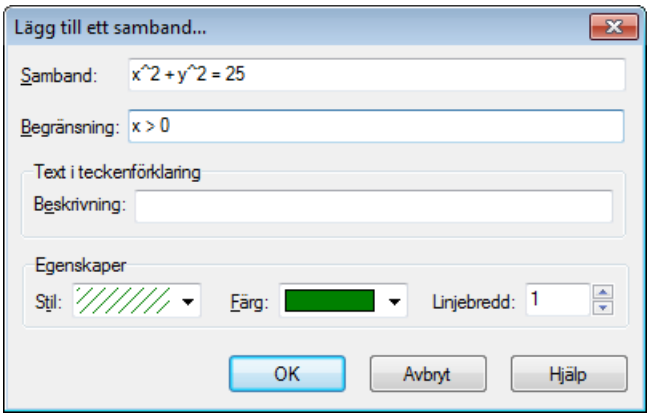
A text label can also contain any OLE object, for example an image or MS Equation. You can paste an OLE object into the editing area like text. A new object can be created at the cursor position by selecting Infoga objekt... in the context menu. If there is more than one representation in the clipboard, you can use Paste special in the context menu to select the representation to paste.

When the OK button is pressed, the label will be shown in the graphing area. The label can be moved by dragging it around with the mouse or it can be locked to one of the axes by right clicking on it and selecting a placement from the context menu. From the context menu it is also possible to rotate the label, for example to show the text vertically.

A label can contain and evaluate a *numeric expression*. This is very useful when you want to show the value of custom constants in a label. Graph will try to evaluate any expression in a label if placed in brackets after a percent sign (%). If you have 3 custom constants a=2.5, b=-3, and c=8.75, you can create a label with the text `f(x)=%(a)x^2+%(b)x+%(c)`. This label will be shown as `f(x)=2.5x^2-3x+8.75` in the graphing area. When you change the constants, the label will be updated to reflect the new values. In the above case, the + preceding %(b) is removed because b evaluates to a negative number.

Lägg till ett samband...

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 → Lägg till relation.... To change an existing relation, you first select it in the *funktionslista* and use Funktion → Redigera....



Samband

Here you enter the relation you want to graph. This must either be an equation or an inequality. *x* and *y* are used as the independent variables. An equation is a statement that one quantity equals another and the quantities must be separated by the = operator. For example the equation `x^2 + y^2 = 25` will plot a circle of radius 5.
An inequality is a statement that one quantity is greater or less than another, and the quantities must be separated by one of the four operators: <, >, <=, >=. An inequality can for example be `abs(x) + abs(y) < 1`. Two operators can be used to specify a range, for example `y < sin(x) < 0.5`.

You can use the same operators and built-in functions as for plotting graphs of functions. In addition you can also create custom functions.

Begränsning

Here you can enter optional constraints, which can be any *numeric expression*. 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.
For example if you have the relation `x^2 + y^2 < 25`, which is a shaded circle, the constraints `x > 0 and y < 0` will only show the part of the circle in the 4th quadrant.

Beskrivning

Here you may enter a descriptive text to show in the *teckenförklaring*. If this field is left empty, the relation and constraints will be shown in the legend.

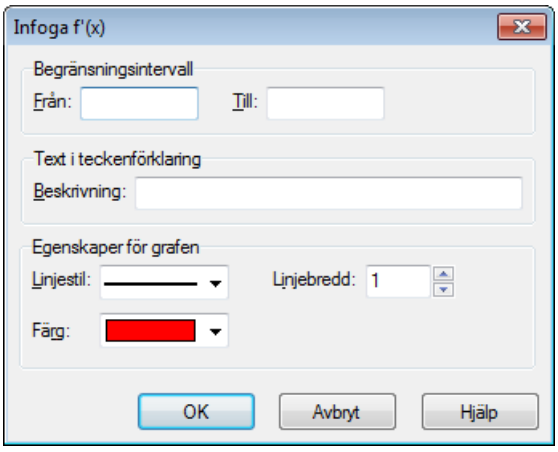
Egenskaper

Here you may select a shading style for inequalities, color and width for equations. The shading *Stil* is only used for inequalities and is ignored for equations. To see overlapping inequalities they must use different styles. The *Linjebredd* indicates the size of the line drawn for equations and the width of the border line for inequalities. For inequalities the width can be set to 0 to avoid drawing the border line.

Infoga f'(x)

The dialog shown below is used to create the first derivative of a function. To create a derivative, you select the function you want to differentiate and use Funktion → Derivata f'(x)....

If the function is a standard function, the first derivative is the slope of the function, and it is defined as the function differentiated with respect to x: f'(x) = df(x)/dx

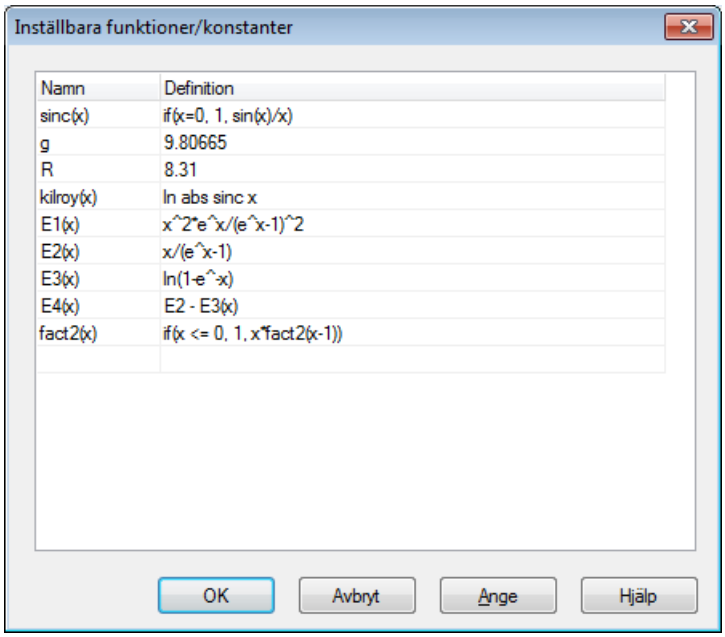


You can select an interval, line style, width in pixels and color for the derivative of the function. The derivative is inserted as a function and can be edited as such. The derivative will not change if you edit the original function.

Inställbara funktioner/konstanter

Graph allows you to define your own custom functions and constants, which you can use in other expressions in the program. You may want to use this to factor out frequently used constants and subexpressions to make

it faster and easier to use these items. Use the Funktion → Skapa funktioner... menu item to show the dialog.



Entering functions

The function and constant names are entered in the first column. The name may contain any combination of letters, digits and underscore but it must always start with a letter. You may not use a name that is already assigned to a built-in function or variable.
Function arguments are entered after the name in brackets separated by comma, e.g. `f(x,y,z)` is a function named f taking three arguments named x, y and z. Like the function name, the argument names must start with a letter and only contain letters and digits.

The expressions you want to define are entered in the second column. The expressions can use the arguments specified in the first column and all built-in functions, other custom functions and constants, and even call themselves recursively. A comment can be written after a # symbol at the end of an expression.

Changing and removing functions

You can remove a function or constant by clearing the name and definition or selecting Ta bort rad from the context menu. All elements using the deleted function or constant will fail when evaluated.

When you press OK or Apply in the shown dialog, all elements are updated to reflect any changes to the functions and constants.

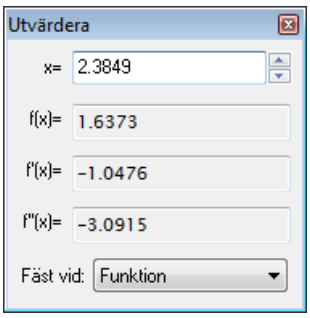
Utvärdera

This dialog is used for interactive calculations on functions. The dialog may be docked below the function list, which is default, or undocked as a floating dialog.

Utvärdera

When Beräkna → Utvärdera is selected the dialog is used to evaluate the selected function at a given value either entered in the dialog or traced with the mouse.

Below you can see the dialog, that appears when a standard function is selected. The dialog will look a little different when a parametric function, polar function or tangent is selected.



You can enter a value for which you want to evaluate the function. The value will be evaluated for the function selected in the *funktionslista*. If the result is on the graph within the shown coordinate system, it will be marked with a dashed cross. You can also trace the drawn graph with the mouse. Just click on the graph with the mouse and the function will be evaluated at the nearest point.

It may happen that the result of an evaluation is a complex number with an imaginary part. This number will either be written as a+b**i**, a∠θ or not written at all depending of the choice under Tillval.

When you click with the mouse on the graphing area you may choose what the cursor will snap to:

Funktion
: The cursor will snap to the nearest point of the selected function.

Skärningspunkt
: The cursor will snap to the nearest intersection between the selected function and every function displayed (including the function itself).

x-axel
: The cursor will snap to the nearest intersection between the selected function and the x-axis.

y-axel
: The cursor will snap to the nearest intersection between the selected function and the y-axis. Not available for standard functions.

Extremvärde för x
: The cursor will snap to the nearest local extreme value for the x-coordinate. Not available for standard functions.

Extremvärde för y
: The cursor will snap to the nearest local extreme value for the y-coordinate.

Area

When Beräkna → Area is selected the dialog is used to calculate the definite integral for the selected function over a specified domain range. For standard functions, parametric functions and tangents the definite integral is the same as the signed area between the graph and the x-axis (the real x-axis and not necessarily the visible one) for the given range.

For polar functions, the definite integral is the same as the signed area between the graph in the given range and the origin. The area is considered negative when the angle goes from a higher to a lower value (clockwise).

For the other functions the area is considered negative when the graph is below the x-axis or when the function goes from a higher to a lower x-value.

You can either enter the range in the edit boxes or select the range with the mouse. The calculated integral will be shown below the range, and the matched 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 accuracy as possible. If an estimated relative error less than 10^{-4} cannot be reached, no result will be shown.

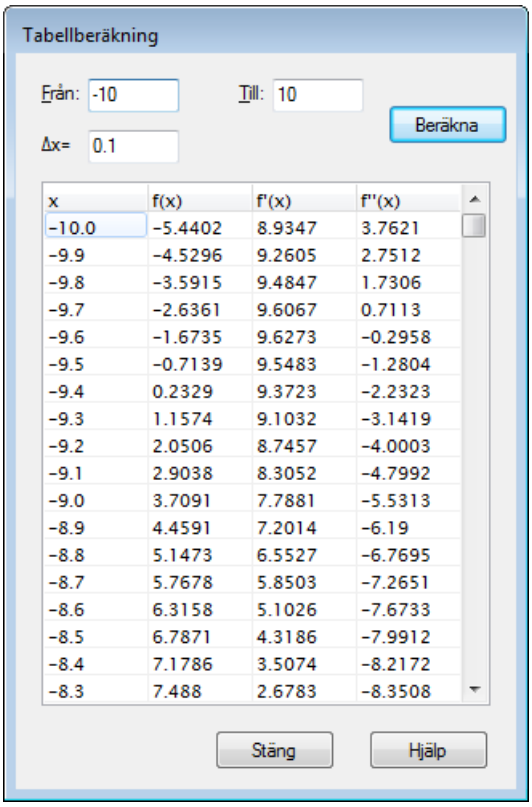
Linjelängd

When Beräkna → Linjelängd is selected the dialog is used to calculate the distance between two points on the function along the curve. You may either enter the range in the dialog or use the mouse to mark it. The range will be marked in the coordinate system. The calculation is done by converting it to an integration and using Simpson's formula through 1000 iterations.

Tabell

The dialog shown below is used to evaluate the selected function for a range. First select a function in the *funktionslista* and use the menu item Beräkna → Tabell to show the dialog. You specify the first and last value of the independent variable in the *Från* and *Till* fields. In the Δx or Δt field you specify the increment of the independent variable at each evaluation.

When you press the Calc button, the table will be filled with the independent variable in the first column. The rest of the columns depend on the type of function. For a standard function, the table will show f(x), f'(x) and f''(x). For a parametric function, the table will show x(t), y(t), dx/dt, dy/dt and dy/dx. For a polar function, the table will show r(t), x(t), y(t), dr/dt and dy/dx. Unneeded columns can be hidden from the context menu. If the calculations takes a long time, a progress indicator will be shown.



You can select some cells with the mouse and right click with the mouse and use Kopiera from the context menu to copy the cells to the clipboard. From the clipboard the data may be pasted into another program, e.g. Microsoft Excel.

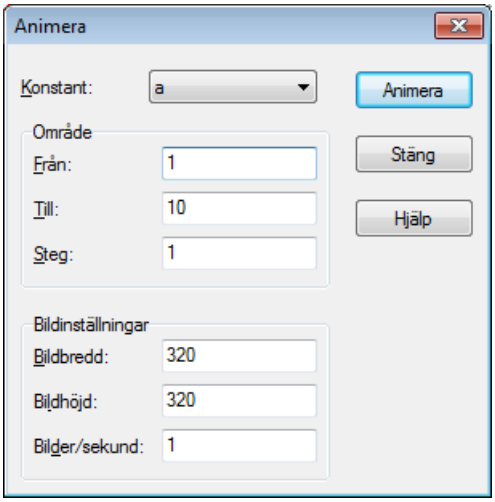
When you move the mouse to the left side of the table the mouse pointer changes to a right pointing arrow. Now you can select whole rows with the mouse. When you move the mouse to the top of the table, the pointer changes to a down pointing arrow. Now you can select whole columns with the mouse. The whole table may be selected by right clicking and selecting Välj alla. It is also possible to select cells by holding down the shift key and using the arrow keys on the keyboard.

From the context menu it is also possible to export the selected data to a file as comma or tab delimited text.

Note that if you choose to create a lot of values in the table, it may take some time to calculate them. Many values may also take up a lot of memory from the system.

Animera

This dialog is used to create an animation by changing a custom constant. The animation can be played directly, saved to a file or copied into a document. The animation can contain all elements supported by Graph, for example functions, relations, point series, labels, etc.



Konstant
: Here you select which constant you want to change in the animation. The constant must already have been created in the Inställbara funktioner/konstanter dialog. The selected constant will be changed in each frame in the animation.

Område
: In the *Från* and *Till* fields you need to specify the range of the selected constant in the animation. The *Stega framåt* value indicates how much the constant is changed between two frames. The number of frames is given by (*Till* - *Från*) / *Stega framåt*. More frames will give a smoother animation but it will also take longer to create and take up more space in the disk file.

Bildinformation
: You can specify the image size of the animation. This will affect the file size and the time it takes to create the animation. The *Bilder/sekund* indicates the default speed of the animation. Most players will be able to adjust the speed when the animation is played.

When you press the Animate button, an animation is created from the settings you have specified. This may take some time depending on what elements exist in the coordinate system and how many frames are needed.

When the animation is finished, a very simple player is shown. You can use this to play the animation. The button will give you some additional options.

Hastighet
: Here you can change the playback speed. This will only affect the playback and not the file saved.

Repetera
: When checked the animation will continue playing. When finished it will start over again.

Autoreversera
: This will make the animation run backwards when it reaches the end. This is most useful in combination with the Repetera option, which will make the animation oscillate between the two ends.

Spara som...
: This will save the animation as an Audio Video Interleave (avi) file, which can be played by any media player.

Spara bild...
: This will save the currently shown frame as a bitmap file. This can be either Windows Bitmap (bmp), Portable Network Graphics (png) or Joint Photographic Experts Group (jpeg).

Spara alla bilder...
: This will save all frames as single bitmap files. This is the same as repeating Spara bild... for each frame in the animation.

Spara som bild

Use the menu item Arkiv → Spara som bild... to save the shown coordinate system as an image file. When the menu item has been chosen, a standard *Spara som* 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) is a bitmap format with loss. It is supported but not recommended because graphs will usually become blurred.

Portable Document Format (pdf)
: Portable Document Format (pdf) is actually not an image format. It is a way to store documents as postscript in a portable way. Graph will store the image as Portable Network Graphics inside the pdf file.

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.

Insticksprogram

To use the plugin system in Graph you need to install Python 3.2 from http://www.python.org. Documentation of the Python language may be found installed with Python or online [http://docs.python.org/3.2/].

Insticksprogram

Plugins are Python scripts 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 `Plugins` directory where Graph is installed, and will automatically be found and loaded by Graph.

Varning

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 hard disk. 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-interpreterande

The plugin system also gives access to a Python interpreter by pressing **F11**. In this interpreter 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

Libraries

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

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

PDFlib-Lite
Användas för att skapa pdf-filer.
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
Numerical library.
Copyright © 2009 Free Software Foundation, Inc.
http://www.gnu.org/software/gsl/

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

Översättningar

Språk	Program	Hjälpfil	Translators
Arabiska	Ja	Nej	Abdellah Chelli
Baskiska	Ja	Nej	Xabier Maiza
Kinesiska (förenklad)	Ja	Nej	Lala Sha
Kinesiska (traditionell)	Ja	Nej	Jian-Jie Dong
Kroatiska	Ja	Ja	Hasan Osmanagić
Tjeckiska	Ja	Nej	Pavel Simerka Martin Stružský Pavlína Krausová
Danska	Ja	Ja	Michael Bach Ipsen Erik Lyngholt Nielsen
Nederländska	Ja	Ja	Etienne Goemaere
Engelska	Ja	Ja	Ivan Johansen
Finska	Ja	Nej	Pekka Lerssi
Franska	Ja	Ja	Jean-Pierre Fontaine
Tyska	Ja	Ja	Frank Hüttemeister Sebastian Stütz Michael Bach Ipsen
Grekiska	Ja	Nej	Theodoros Kannas
Ungerska	Ja	Nej	Gabor Magyari
Italienska	Ja	Ja	Alessandro Serena Attilio Ridomi
Koreanska	Ja	Nej	Choe Hyeon-gyu
Mongoliska	Ja	Nej	Batnasan Davaa
Norska	Ja	Nej	Tore Ottinsen
Persiska	Ja	Nej	Shayan Abyari Yashar PourMohammad
Polska	Ja	Nej	Paweł Baczyński
Portugisiska (Brasilien)	Ja	Ja	Jorge Costa Andre Fduarte Haroldo Luiz Bertoldo Janete Flor de Maio Fonseca Aldemar Calazans Filho
Portugisiska (Portugal)	Ja	Nej	Jorge Geraldes
Ryska	Ja	Nej	Ivans Leonovs
Serbiska	Ja	Nej	Jasmina Malinovic Branimir Krstic
Slovenska	Ja	Ja	Jernej Baša Rok Štokelj Barbara Pušnar Sergej Pušnar

Språk	Program	Hjälpfil	Translators
Spanska	Ja	Ja	Francisco Oliver Alejandro Arce
Svenska	Ja	Nej	Pär Smårs Michael Bach Ipsen
Turkiska	Ja	Nej	Mumtaz Murat Arik
Vietnamesiska	Ja	Nej	Trung

Miscellaneous

The icon for Graph was designed by Jonathan Holvey.

Ordlista

complex number
: 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.
Complex numbers are used in the *Utvärdera* dialog for standard functions and for graphing functions when *Beräkna med komplexa tal* is enabled under the Inställningar tab in the *Redigera koordinataxlarna* dialog.

funktionslista
: The function list is shown in the left side of the main window. This list shows a list of all functions, tangents, point series, shadings and relations. When you want to manipulate an item, you first have to select it. The selected item is normally marked in blue, but it will be marked in gray when something other than the function list has focus. You can manipulate the selected element through the Funktion menu or through the context menu appearing when you right click on the element.

graph element
: A graph element is something that is shown in the coordinate system. This can be a function, point series, label, relation, etc. The graph elements are also shown in the function list where they can be manipulated from the Funktion menu or the context menu.

heltal
: The set of numbers ...,-3,-2,-1,0,1,2,3,... is called integers and is a subset of the real numbers. A given integer n may be negative, zero or positive.

numeric expression
: An expression that can be evaluated as a number is called a numeric expression. The expression can include any combination of numbers, constants, variables, operators and functions.

radianer
: Radians are a way to describe the size of an angle similar to degrees, but radians are not a unit like degrees. The angle of a whole circle is 360° or 2π radians. An angle in radians can be converted to degrees by multiplying with 180°/π. An angle in degrees can be converted to radians by multiplying with π/180°. You can choose to use radians or degrees for trigonometric functions in the *Redigera koordinataxlarna* dialog under the Inställningar tab.

real number
: A real number is on the form nnn.fffEeee where nnn is the whole number part that may be negative. fff is the fraction part that is separated from the integer part with a dot '.'. The fraction part is optional, but either the integer part or the fraction part must be there. E is the exponent separator and must be an 'E' in upper case. eee is the exponent optionally preceded by '-'. The exponent is only needed if the E is there. Notice that 5E8 is the same as 5*10^8.

teckenförklaring
: The legend is a box that per default is placed at the upper right corner of the graphing area, and shows a list of the plotted functions, tangents, shadings, and point series in the coordinate system. Select *Visa teckenförklaring* under Inställningar in the *Redigera koordinataxlarna* dialog to show the legend. Right click on an item in the function list and deselect Visa i teckenförklaringen if you don't want the item shown in the legend. When editing an item you can enter the text to be shown in the legend. For functions and tangents the function equation will be shown if no legend text is entered.