

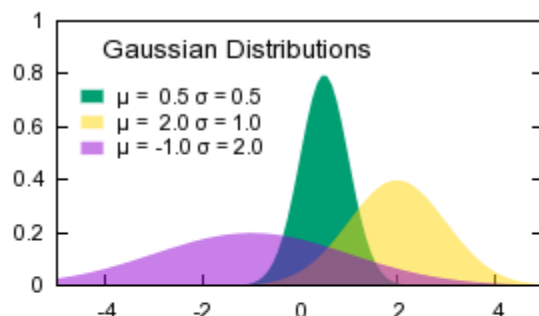
# Podstawy systemu gnuplot

Jacek Golak

Narzędzia obliczeniowe fizyki

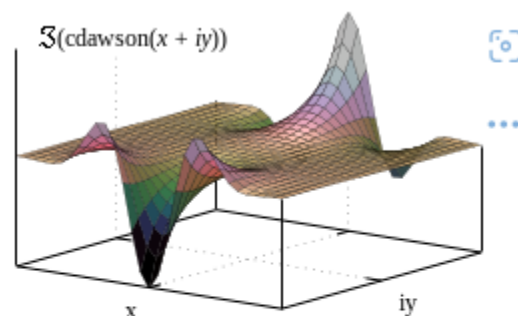
UJ WFAIS 2023/2024

# gnuplot homepage



[FAQ](#)  
[Documentation](#)  
[Demos](#)  
[Download](#)

[Contributed scripts](#)  
[External Links](#)  
[Tutorials and guides](#)



**Gnuplot** is a portable command-line driven graphing utility for Linux, OS/2, MS Windows, OSX, VMS, and many other platforms. The source code is copyrighted but freely distributed (i.e., you don't have to pay for it). It was originally created to allow scientists and students to visualize mathematical functions and data interactively, but has grown to support many non-interactive uses such as web scripting. It is also used as a plotting engine by third-party applications like Octave. Gnuplot has been supported and under active development since 1986.

**Gnuplot supports many different types of 2D and 3D plots**

Here is a [Gallery of demos](#).

**Gnuplot supports many different types of output**

interactive screen display:

direct output to file:

mouseable web display formats:

cross-platform (Qt, wxWidgets, x11) or system-specific (MS Windows, OS/2)

postscript (including eps), pdf, png, gif, jpeg, LaTeX, metafont, emf, svg, ...

HTML5, svg

## Version 5.4 (current stable)

- [Release 5.4.10 \(Oct 2023\)](#)
- [Release Notes](#)
- [User Manual \(PDF\)](#)
- [Demo gallery](#)
- [contributed binaries for macOS](#)

## Gnuplot 6.0 release candidate

- [Gnuplot 6.0 rc3 \(November 2023\)](#)
- [Release Notes](#)
- [User Manual \(PDF\)](#)
- [Demo gallery](#)
- [On-line documentation \(HTML\)](#)

## Development version 6.1

- New features are added regularly. Git repository [here](#).
- [User Manual \(PDF\)](#)
- [Demo gallery](#)
- [On-line documentation \(HTML\)](#)

gnuplot jest (darmowym !) narzędziem służącym do:

- tworzenia wykresów dwuwymiarowych i trójwymiarowych funkcji jednej lub dwóch zmiennych; funkcje mogą być dane w postaci jawnej lub parametrycznej, a rysowanie odbywa się nie tylko we współrzędnych kartezjańskich, ale też biegunowych
- przedstawiania w różnej formie wyników pomiarów obarczonych różnego typu niepewnościami
- przedstawiania w różnej formie wyników obliczeń dostępnych w formie danych zapisanych na pliku (łącznie z ich interpolacją)
- wykonywania (nie tylko bardzo prostych) obliczeń (definiowanie zmiennych, własnych funkcji itp.)
- współpracy z innymi środowiskami takimi jak Octave lub LaTeX

Jego głównymi autorami są Thomas Williams i Colin Kelley.  
Jest rozwijany od 1986 roku !

gnuplot jest dostępny na wielu platformach: linux, MS Windows, OSX

## Release History

- gnuplot 5.4 Jul 2020 ([5.4.9](#) Sep 2023)
- gnuplot 5.2 Sep 2017 ([5.2.8](#) Dec 2019)
- gnuplot 5.0 Jan 2015
- gnuplot 4.6 2012
- gnuplot 4.4 2010
- gnuplot 4.2 2007
- gnuplot 4.0 2004
- gnuplot 3.7 1999
- gnuplot 3.5 1993
- gnuplot 2.0 1990
- gnuplot 1.0 1986

## Your copy of gnuplot documentation

First of all, gnuplot documentation should have been included in your local gnuplot installation.

- `gnuplot.pdf` — full User Manual
- `faq.pdf` — Frequently Asked Questions
- `gnuplot.gih` — on almost all platforms: this is the file from which the `help` commands get their help
- `gnuplot.inf` and `gnupmdrv.inf` on IBM OS/2 and eComStation
- `gnuplot.hlp` on MS Windows

## Official gnuplot online documentation

The following official documentation is **available online**:

- [Gnuplot 5.5](#) (HTML) Covers both the current release (5.4) and the development version (5.5)

### Current

- Version 5.4 (English) [PDF format](#).
- Version 5.4 (日本語) [PDF format](#).

### Old

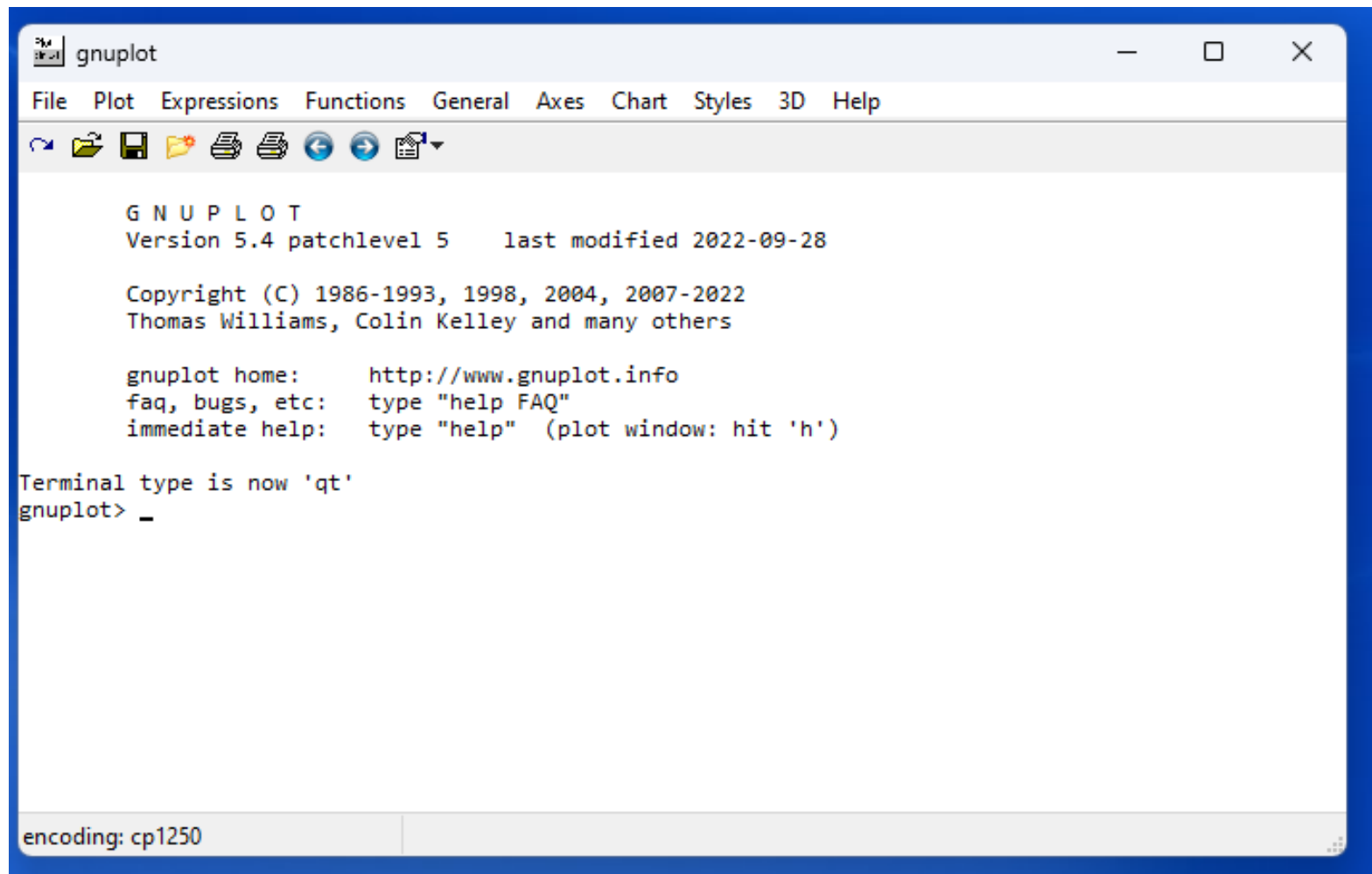
- Version 5.2 (English) [PDF format](#).

### Older

- Version 4.6 (English) [PDF format](#).
- Version 4.6 (日本語) [PDF format](#).

W systemach Windows wystarczy ściągnąć i uruchomić odpowiedni plik instalacyjny (obecnie gp548-win64-mingw.exe)

Po uruchomieniu pojawia się okno, gdzie możemy wpisywać komendy:



Bardzo liczne przykłady <http://www.gnuplot.info/demos/>

## gnuplot demo plots

Gnuplot comes with a large collection of demonstration plots. You can step through these interactively by typing the command below in gnuplot's `demo/` directory — it should be part of your installation, otherwise get it from the [source code archive](#) or file-by-file from the [git repository](#).

```
gnuplot> load "all.dem"
```

Hint: you can get a single big PDF file of all the demo plots by the following sequence of commands, followed by hitting <enter> many times.

```
gnuplot> set terminal pdf
gnuplot> set output "all_demos.pdf"
gnuplot> load "all.dem"
gnuplot>
```

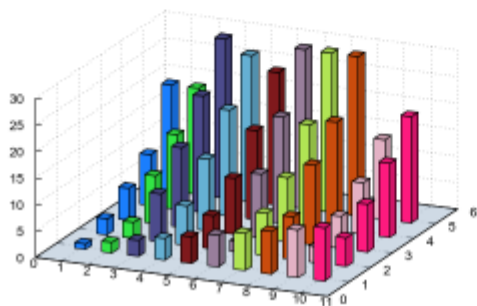
More practical on systems with a linux-like shell is the single line command

```
gnuplot -e "set term pdf" all.dem < /bin/yes > all_demos.pdf
```

Indexed online versions of the demos are available by clicking below:

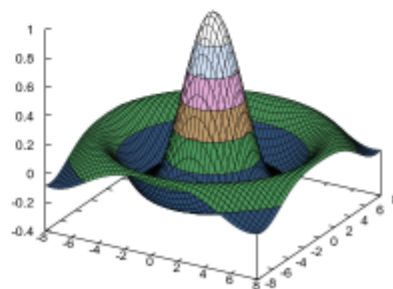
### Version 5.4 demos

Full treatment: 3D boxes with pm3d depth sorting and lighting

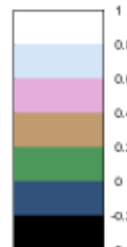


### Version 6.0 demos

set contourfill cblines  
splot with contourfill

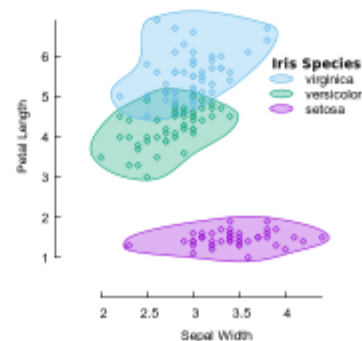


set palette cubehelix  
maxcolors 7



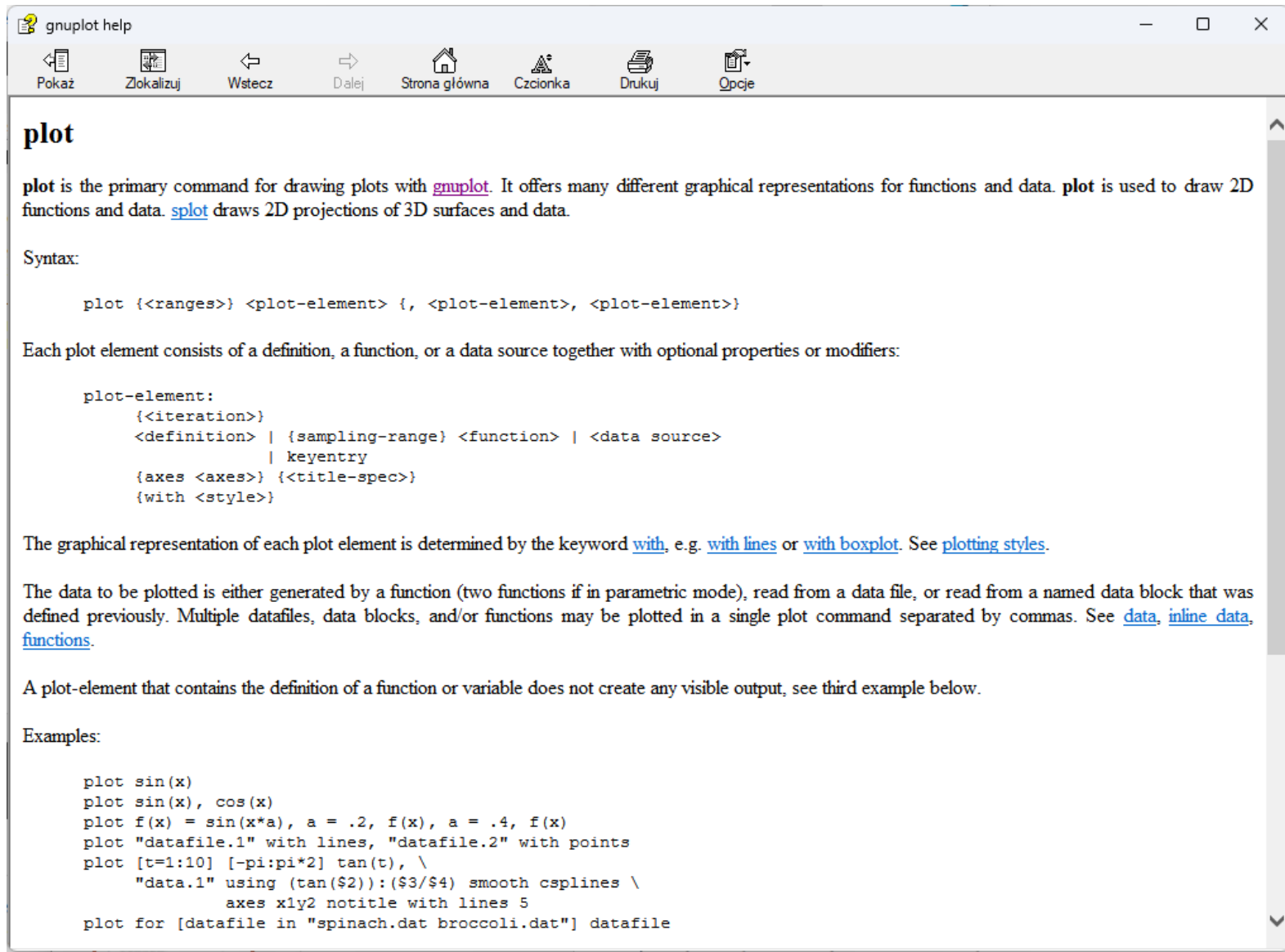
### Development version (6.1)

Concave Hull chi\_length = 1.5 expand 0.1



Z każdą wersją dostępny jest obszerny przewodnik po programie (nazywa się skromnie „Gnuplot\_5.4.pdf”, jest utworzony przy pomocy LaTeX’a i liczy blisko 270 stron.

Można też uzyskać pomoc „on-line” na przykład o komendzie plot: help plot





gnuplot może produkować output bezpośrednio na ekran lub w formatach plików graficznych:

Portable Network Graphics (PNG),  
Encapsulated PostScript (EPS),  
Scalable Vector Graphics (SVG),  
Joint Photographic Experts Group (JPEG) i wielu innych.

Może też wyprodukować komendy LaTeXa do tworzenia rysunków, które mogą być włączone bezpośrednio do dokumentów LaTeX-a (korzystając z fontów i łatwości tworzenia wyrażeń matematycznych w LaTeX-u)

Gnuplot może być używany interaktywnie, gdy wpisujemy kolejne komendy po wywołaniu programu

```
gnuplot> plot sin(x)
```

lub w tzw. trybie wsadowym, gdy najpierw zapisujemy zbiór poleceń w pliku tekstowym, na przykład „witam.gp”, a potem piszemy w terminalu

```
[golak@olcha ~]$ gnuplot witam.gp
```

gnuplot przychodzi z ogromnym zbiorem przykładowych rysunków umieszczonych w kartotece .../demo danej instalacji.

Można je zbadać interakcyjnie

(pojedyncze pliki \*.dem) lub „załadować” wszystkie przykłady zawarte w pliku „all.dem”:

```
gnuplot> load "all.dem"
```

Uwaga: można też otrzymać pojedynczy plik postscriptowy, wykonując pod Linuksem następujący ciąg poleceń

```
gnuplot> set terminal postscript color solid
```

```
gnuplot> set output "all_demos.ps"
```

```
gnuplot> load "all.dem,,
```

( .... trzeba wiele razy naciskać <Enter> ... )

```
gnuplot> quit
```

(aby zakończyć działanie programu)

Następnie plik postscriptowy można przepisać do pdf.

Korzystam ze strony <http://people.duke.edu/~hpgavin/gnuplot.html>

Funkcje dostępne w programie gnuplot:

Function	Returns
abs(x)	absolute value of x,  x
acos(x)	arc-cosine of x
asin(x)	arc-sine of x
atan(x)	arc-tangent of x
cos(x)	cosine of x, x is in radians.
cosh(x)	hyperbolic cosine of x, x is in radians
erf(x)	error function of x
exp(x)	exponential function of x, base e
inverf(x)	inverse error function of x
invnorm(x)	inverse normal distribution of x
log(x)	log of x, base e
log10(x)	log of x, base 10
norm(x)	normal Gaussian distribution function
rand(x)	pseudo-random number generator
sgn(x)	1 if x > 0, -1 if x < 0, 0 if x=0
sin(x)	sine of x, x is in radians
sinh(x)	hyperbolic sine of x, x is in radians
sqrt(x)	the square root of x
tan(x)	tangent of x, x is in radians
tanh(x)	hyperbolic tangent of x, x is in radians

Bessel, gamma, ibeta, igamma, and lgamma functions are also supported. Many functions can take complex arguments. Binary and unary operators are also supported.

## Uwagi:

W zasadzie dowolne wyrażenie matematyczne dostępne w C, Pascalu, Fortranie może być użyte i narysowane w gnuplocie. Kolejność operatorów jest taka, jak w języku C !

Dla potęgowania gnuplot używa zapisu z Fortranu:

$x^{**3}$  oznacza „x do potęgi 3”

# (hash) jest znakiem komentarza dla gnuplota; część linii po znaku „#” jest ignorowana.

gnuplot używa standardowo zmiennej x przy rysowaniu wykresu funkcji jednej zmiennej

```
gnuplot> plot sqrt(x)
```

oraz zmiennych (x,y) do rysowania wykresu funkcji dwóch zmiennych (powierzchni)

```
gnuplot> splot sin(x+0.5*y)
```

## Uwagi (cd)

gnuplot używa standardowo zmiennej  $t$  przy rysowaniu wykresu 2D funkcji danej w postaci parametrycznej

```
gnuplot> set parametric  
gnuplot> plot cos(t),sin(t)  
gnuplot> splot cos(t),sin(t),2*t
```

oraz zmiennych  $(u,v)$  do rysowania wykresu funkcji dwóch zmiennych (powierzchni) danej w postaci parametrycznej

```
gnuplot> set parametric  
gnuplot> splot sin(u)*cos(v),sin(u)*sin(v),cos(u)
```

Jedna ze zmiennych „ $u$ ” lub „ $v$ ” powinna być także użyta przy rysowaniu krzywej w 3D danej w postaci parametrycznej

```
gnuplot> set parametric  
gnuplot> splot cos(v),sin(v),2*v
```

# Definiowanie zmiennych i funkcji

gnuplot rozróżnia duże i małe litery !

Definiowanie zmiennych odbywa się w chwili pierwszego przypisania im wartości. Zasadniczą rolę odgrywa podział na zmienne całkowite i zmiennoprzecinkowe.

```
gnuplot > a=2.0 # definiujemy zmienną zmiennoprzecinkową i nadajemy jej wartość 2
```

```
gnuplot> b=2 # definiujemy zmienną całkowitą i nadajemy jej wartość 2
```

Podobnie jak w C jest różnica między a i b !

```
gnuplot> a=2.0
```

```
gnuplot> b=2
```

```
gnuplot> c=a/3
```

```
gnuplot> d=b/3
```

```
gnuplot> print a,b,c,d
```

```
2.0 2 0.6666666666666667 0
```

```
gnuplot>
```

Uwaga na **specjalne** traktowanie zmiennych  $x$ ,  $y$ ,  $t$ ,  $u$  i  $v$   
przywołaniu funkcji „plot”

Przykład:

```
gnuplot> a=2.0  
gnuplot> print a  
2.0                # wartość zmiennej wynosi 2.0  
gnuplot> plot a     # program narysuje wykres funkcji  $y=2$ 
```

```
gnuplot> x=2.0  
gnuplot> print x  
2.0                # wartość zmiennej wynosi 2.0  
gnuplot> plot x     # program narysuje wykres funkcji  $y=x$  !
```

Zdefiniowane zmienne możemy zobaczyć przy pomocy komendy  
**show variables**

```
gnuplot> show variables
```

User and default variables:

pi = 3.14159265358979

NaN = NaN

GNUTERM = "wxt"

a = 2.0

b = 2

c = 0.6666666666666667

d = 0

x = 2.0



W programie gnuplot możemy zdefiniować własne funkcje

```
gnuplot> f(x)=2*x+sqrt(x)
gnuplot> h(t)=2*t
gnuplot> g(x,y)=exp(-x**2-4*y**2)
gnuplot> i(u,v)=log(u)+v/(1+v**2)
gnuplot> print f(2), h(2.5), g(0,1), i(exp(1),0)
5.41421356237309 5.0 0.0183156388887342 1.0
gnuplot> plot f(x)
gnuplot> splot g(x,y)
```

wykonywać przy pomocy tych funkcji obliczenia oraz używać ich przy rysowaniu (plot) i dopasowywaniu funkcji (fit – za chwilę)

Zdefiniowane funkcje można sprawdzić przy pomocy komendy

```
gnuplot > f(x)=sqrt(x)
gnuplot> show functions
```

User-Defined Functions:  
 $f(x)=2*x$

Dzięki temu, że w programie gnuplot mamy „instrukcję warunkową”  
 $a ? b : c$

(taką samą jak w C) możemy definiować funkcję w kilku przedziałach

Przykład 1 (wartość bezwzględna z x)

```
gnuplot> w1(x)=(x<0) ? -x : x
gnuplot> print w1(-2),w1(0),w1(4)
2 0 4
```

Przykład 2 (help gnuplota: funkcja, która jest równa

$\sin(x)$  dla  $0 \leq x < 1$ ,

$1/x$  dla  $1 \leq x < 2$ ,

i jest niezdefiniowana dla pozostałych wartości x

(tam gdzie jest niezdefiniowana, nie pojawi się też na rysunku))

```
gnuplot> w2(x) = 0<=x && x<1 ? sin(x) : 1<=x && x<2 ? 1/x : 1/0
gnuplot > pr w2(pi/6),w2(1.5)
0.5 0.666666666666667
gnuplot> pr w2(pi/6),w2(1.5),w2(4)
```

^

undefined value

## Przykład prostych obliczeń z liczbami zespolonymi

```
gnuplot> a={3,5}
gnuplot> b={-2,5}
gnuplot> z1=a+b
gnuplot> z2=a-b
gnuplot> z3=a*b
gnuplot> z4=a/b
gnuplot> print z1
{1.0, 10.0}
gnuplot> print z2
5.0
gnuplot> print z3
{-31.0, 5.0}
gnuplot> print z4
{0.655172413793103, -0.862068965517241}
gnuplot>
```

## Dopasowanie parametrów funkcji

Rozważmy ten sam przykład, który liczyliśmy przy pomocy programu Mathematica® (notebook „regresja\_klasyczna\_ab.nb” i plik z danymi w formie dwóch kolumn (x,y)

#### dopasowanie prostej -> regresja klasyczna ####

```
gnuplot> f(x)=a*x+b      # postać funkcji
gnuplot> a=1.0           # początkowa wartość parametru a
gnuplot> b=1.0           # początkowa wartość parametru b
gnuplot> FIT_LIMIT=1.0e-8 # kryterium zbieżności; kiedy
                        # zaprzestać dalszych iteracji
gnuplot> fit f(x) 'regresja_klasyczna_ab_input.txt' via a,b
                        # z którego pliku
```

..... końcowe wyniki (właściwie ich zasadnicza część) .....

After 6 iterations the fit converged.

final sum of squares of residuals : 133.9

rel. change during last iteration : -2.05568e-09

degrees of freedom (FIT\_NDF) : 3

rms of residuals (FIT\_STDFIT) = sqrt(WSSR/ndf) : 6.68082

variance of residuals (reduced chisquare) = WSSR/ndf : 44.6333

Final set of parameters                      Asymptotic Standard Error

=====

a            = 3.71            +/- 0.2113      (5.695%)

b            = -263.35        +/- 18.2        (6.913%)

gnuplot potrafi bardzo, bardzo wiele, ale ...

Reszta jest ćwiczeniem !