



Universidade Federal de Viçosa  
**DEPARTAMENTO DE FÍSICA**  
**FIS 271 (Física computacional I)**

**Gnuplot** é um software gratuito, usado por comando de linha, para análise de dados e criação de gráficos 2D e 3D (<http://www.gnuplot.info>). O gnuplot é multiplataforma e pode ser usado com Linux, OS/2, MS Windows, OSX, VMS, etc.

**Comandos básicos:**

```
hallan@Infoway-ST-4254 ~  
Arquivo Editar Ver Pesquisar Terminal Ajuda  
hallan@Infoway-ST-4254 ~ $ gnuplot  
  
G N U P L O T  
Version 5.0 patchlevel 3    last modified 2016-02-21  
  
Copyright (C) 1986-1993, 1998, 2004, 2007-2016  
Thomas Williams, Colin Kelley and many others  
  
gnuplot home:      http://www.gnuplot.info  
faq, bugs, etc:   type "help FAQ"  
immediate help:    type "help" (plot window: hit 'h')  
  
Terminal type set to 'qt'  
gnuplot> 
```

- gnuplot** (inicializa o Gnuplot a partir de um terminal (veja a figura ao lado))
- plot** (comando primário para se criar gráficos 2D)
- splot** (cria gráficos/superfícies 3D)
- replot** (redesenha o último *plot* com ou sem argumentos novos)
- test** (disponibiliza os *tipos de linhas* e *pontos* existente para o terminal em uso)
- set title** (define um *título* para todos os gráficos a serem criados)
- set xlabel** (define um *nome* padrão para o eixo x)
- set ylabel** (define um *nome* padrão para o eixo y)
- set border** (define uma *borda* para os gráficos a serem criados)
- set grid** (cria o gráficos como uma *rede quadriculada*)
- set xrange** (define os limites padrões para o eixo x)
- set yrange** (define os limites padrões para o eixo y)
- set term** (especifica para o gnuplot que tipo de *saída* usar: tela, EPS, JPG, etc)
- set output** (redireciona a saída para um arquivo ou periférico específico)
- fit** (faz ajustes de funções definidas em um conjunto de dados)
- #** (comentário)

**Criando gráficos a partir de um conjunto de dados:**

- `plot "data.txt"`  
# cria um gráfico 2D utilizando-se as duas primeiras colunas (padrão) do arquivo data.txt
- `plot "data.txt" using 1:3`  
#cria um gráfico 2D utilizando-se as colunas 1 e 3 do arquivo data.txt
- `plot "data.txt" u 1:3`  
#o mesmo que o anterior
- `plot "data.txt" using 1:2, "data.txt" using 1:3`  
#cria um gráfico 2D com dois *plots*, o primeiro *plot* usando-se as colunas 1 e 2 do arquivo data.txt e o #segundo *plot* utilizando-se as colunas 1 e 3 do mesmo arquivo
- `plot "data.txt" using 1:2 title 'amostra 1' , "data.txt" u 1:3 t 'amostra 2'`  
#o mesmo que o anterior com legenda "amostra 1" para o primeiro *plot* e legenda "amostra 2" para o #segundo *plot*
- `plot "data.txt" using 1:2 title 'amostra 1' with lines, "data.txt" u 1:3 t 'amostra 2' w linespoints`  
#o mesmo que o anterior sendo o primeiro *plot* feito com linha e o segundo *plot* com linha e pontos

plot "data.txt" using 1:2 title 'amostra 1' with lines lt 4 lw 2, "data.txt" u 1:3 t 'amostra 2' w linespoints lt 8 lw 2  
#o mesmo que o anterior sendo o primeiro *plot* feito com linha do tipo 4 e espessura 2 e o segundo *plot*  
#com linha/ponto do tipo 8 e espessura também igual a 2 (opções para tipos de linha/ponto ver comando  
test)

### Algumas funções do Gnuplot:

<b>abs(x)</b> módulo de x,  x	<b>acos(x)</b> arc-cosseno de x
<b>asin(x)</b> arc-seno de x	<b>atan(x)</b> arc-tangente de x
<b>cos(x)</b> cosseno de x (radianos)	<b>exp(x)</b> exponencial de x
<b>log(x)</b> log de x, base e	<b>log10(x)</b> log of x, base 10
<b>sin(x)</b> seno de x (radianos)	<b>sqrt(x)</b> raiz quadrada de x
<b>tan(x)</b> tangente de x (radianos)	

### Criando gráficos a partir de funções pré-definidas:

plot sin(x)/x  
#cria um gráfico 2D com funções pré-definidas do gnuplot

replot cos(x)  
#o mesmo *plot* anterior, com um segundo *plot* usando a função cos(x)

splot x\*y  
#cria um gráfico 3D com funções pré-definidas do gnuplot

replot x/y  
#o mesmo *plot* anterior, com um segundo *plot* usando a função x/y

### Definindo funções:

f(x)= x\*\*2-2\*x-15  
#cria uma função do segundo grau, definida pelo usuário, com o nome f(x)  
#Note que gnuplot usa aritmética "real" e "integer" como FORTRAN e C.

plot f(x)  
#cria um gráfico 2D da função f(x) definida pelo usuário

### Algumas opções de apresentação:

```
set term png
#muda a saída para um arquivo do tipo PNG

set output "grafico.png"
#define o nome do arquivo a ser usado como saída

set title "Física Computacional"
#define o título Física Computacional como padrão

set xlabel "eixo X"
#define eixo x para o nome do eixo x

set ylabel "eixo Y"
#define eixo y para o nome do eixo y
```

```
set border
#define bordas para o gráfico

set grid
#insere um grid para o gráfico (ajuda na visualização)

set xrange [-10:10]
#define os limites para o eixo x no intervalo de -10 até 10

set yrange [-20:10]
#define os limites para o eixo y no intervalo de -20 até 10
```

### Fazendo ajustes de funções com o Gnuplot:

```
f1(x) = a1*x**b1
```

```
#define a função a ser ajustada
```

```
a1 = 1; b1 = 1
```

```
#valores iniciais para a1 e b1 respectivamente
```

```
fit f1(x) 'data2.txt' using 1:2 via a1, b1
```

```
#ajusta a função f1(x), definida pelo usuário, ao conjunto de dados do arquivo data2.txt
```

### Fazendo Scripts com o Gnuplot:

Um opção que pode ser vantajosa é escrever os comandos que seriam executados no ambiente Gnuplot em um arquivo de texto (*script*).

No exemplo a seguir, o arquivo de texto chamado script.gnp, contém as seguintes linhas Gnuplot:

```
----- script.gnp -----
```

```
set term png
```

```
set output "teste.png"
```

```
set title "Física Computacional"
```

```
set xlabel "eixo X"
```

```
set ylabel "eixo Y"
```

```
set border
```

```
set grid
```

```
set xrange [-10:10]
```

```
set yrange [-20:10]
```

```
f(x)= x**2-2*x-15
```

```
plot f(x)
```

```
-----
```

Para ler os comando do arquivo texto e criar o gráfico, basta chamar o arquivo texto com o Gnuplot em um terminal:

```
gnuplot script.gnp
```