

# MINICURSO

# Octave

## BÁSICO

MÓDULO 2





## Roteiro Módulo 2

- Funções e rotinas;
- Gráficos;
- Polinômios;
- Técnicas de cálculo.



# Funções e rotinas



# Funções

- Mais algumas funções já definidas no Octave

<code>abs(x)</code>	valor absoluto de x
<code>acos(x)</code>	arco cosseno de x
<code>asin(x)</code>	arco seno de x
<code>atan(x)</code>	arco tangente de x
<code>cos(x)</code>	cosseno de x
<code>sin(x)</code>	seno de x
<code>tan(x)</code>	tangente de x
<code>exp(x)</code>	exponencial de x
<code>log(x)</code>	logaritmo natural de x
<code>log10(x)</code>	logaritmo de x na base 10
<code>ged(x,y)</code>	máximo divisor comum de x e y
<code>lem(x,y)</code>	mínimo múltiplo comum de x e y
<code>rem(x,y)</code>	resto da divisão de x por y

- Ângulos:  
`degtorad(ângulo em graus)`  
`radtodeg(ângulo em radianos)`



# Criando funções

- Dentro do corpo do código:  
function F=nome(x)  
    Código que a função executará  
endfunction
- Funções anônimas:  
@(parâmetros)(código)

Exemplo:

```
quad(@(x)(log(x)), 0,2)
```

- Comandos curtos
- Função única



# Rotinas ou Arquivos M-files

- Vantagens:
  - Portabilidade;
  - Possibilidade de edição;
  - Código limpo;
- Condição:
  - Arquivo com o mesmo nome da função;
  - Apenas letras, números ou underlines;



# Gráficos



# Funções úteis

<code>title( 'Título' )</code>	Insera a palavra "Título" como título do gráfico
<code>xlabel( 'legenda' )</code>	Insera a palavra "legenda" como legenda do eixo x
<code>ylabel( 'texto' )</code>	insere a palavra "texto" como legenda do eixo y
<code>text( x, y, 'texto' )</code>	Insera a palavra "texto" na coordenada de (x,y)
<code>gtext( 'texto' )</code>	Adiciona a palavra "texto" na posição do mouse
<code>legend( 'texto1' )</code>	Insera legendas nos gráficos
<code>grid on</code>	Insera grade que pode ser retirada mudando <b>on</b> para <b>off</b>
<code>hold on</code>	Plotas várias curvas no mesmo gráfico até que o código <b>hold off</b> seja executado
<code>axis(v)</code>	Plotar em uma faixa de valores (v é o vetor formado por xmin xmax ymin ymax)





# Gráficos 2D

- Coordenadas cartesianas:
  - `plot()`;
  - `semilogx()`;
  - `semilogy()`;
  - `loglog()`;
- Coordenadas polares:
  - `polar()`
  - `scatter()`
- Outros gráficos:
  - `stairs()` - degrau;
  - `bar()` - histogramas;
- Mais de um gráfico:
  - `hold on`;
  - `figure`;
  - `subplot`;



# Parâmetros para gráficos 2D

- Valores de x;
- Valores de y;
- Código de aparência:
  - Tipo de linha;
  - Marcador dos pontos;
  - Cor;

Cor	Código
Azul	b
Branco	w
Ciano	c
Preto	k
Roxo	m
Verde	g
Vermelho	r
Amarelo	y

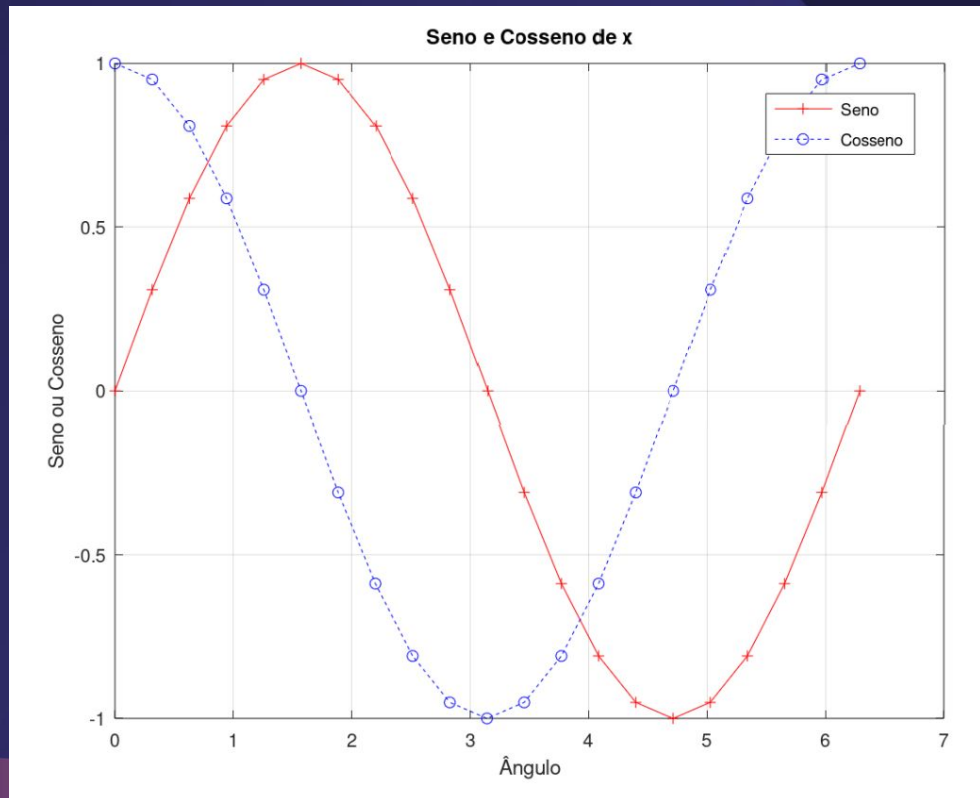
Marcador	Código
Mira	+
Círculo	o
Estrela	*
Ponto	.
Cruz	x
Quadrado	s
Diamante	d
Pentagrama	p

Linha	Código
Sólida	" -
Tracejada	"- -
Pontilhada	":
Traço-ponto	"-.



# Exemplo de gráfico 2D

```
x = 0:pi/10:2*pi;  
y1=sin(x);  
y2=cos(x);  
plot(x, y1, "+r")  
hold on  
plot(x, y2, "bo-")  
legend('Seno','Cosseno')  
title('Seno e Cosseno de x')  
xlabel('Ângulo') ylabel('Seno ou Cosseno')  
grid on
```

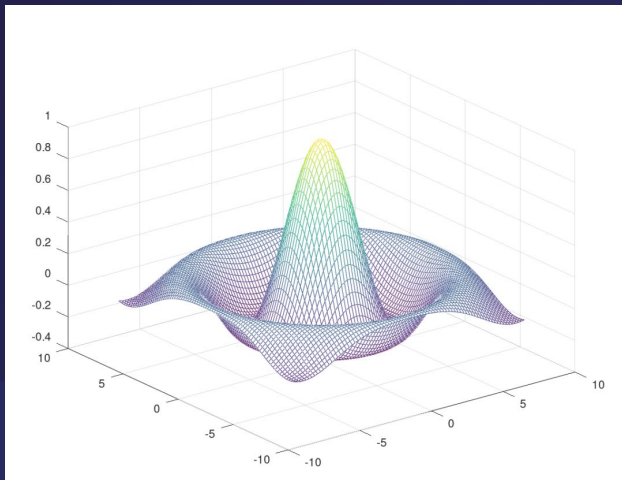


# Gráficos 3D

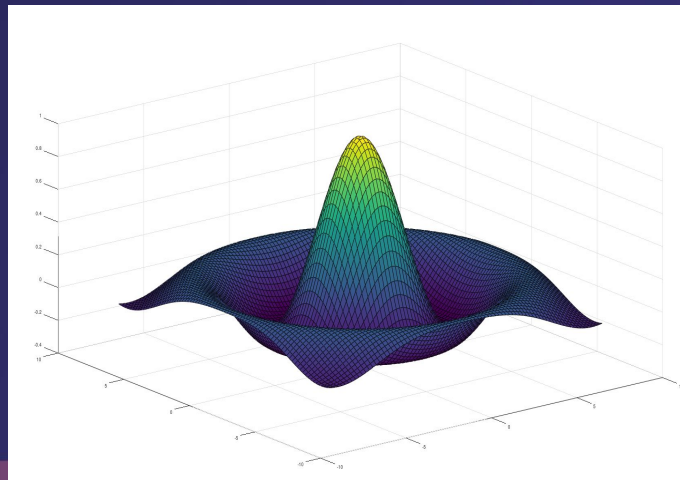
- Curvas tridimensionais:
  - `plot3()`;
- Malhas de superfícies:
  - `meshgrid()` e `mesh()`;
- Superfície sólida:
  - `surf()`;

# Exemplo de gráfico 3D

```
[x,y]=meshgrid(-8:0.2:8,-8:0.2:8)
r=sqrt(x.^2 +y.^ 2 )
z=sin(r)./r
mesh(x,y,z)
```



```
[x,y]=meshgrid(-8:0.2:8,-8:0.2:8)
r=sqrt(x.^2 +y.^ 2 )
z=sin(r)./r
surf(x,y,z)
```

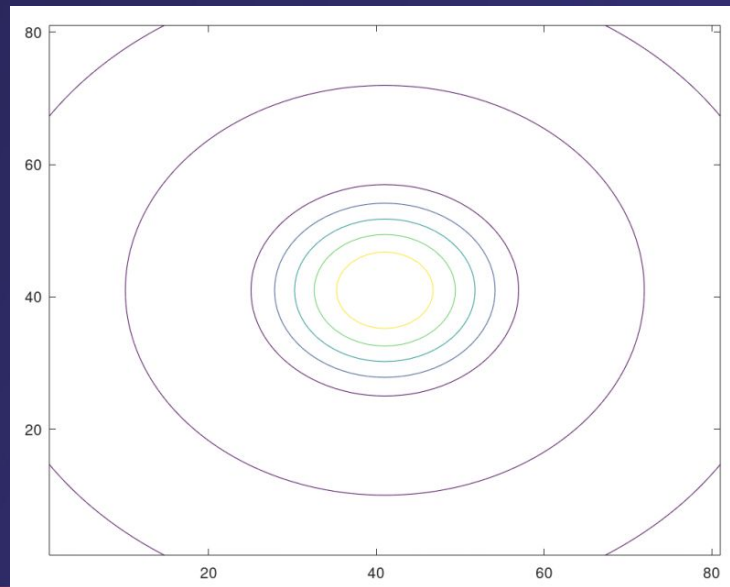




# Curvas de nível

- `contourn(z, numero de curvas)`

```
[x,y]=meshgrid(-8:0.2:8,-8:0.2:8)  
r=sqrt(x.^2 +y.^ 2 )  
z=sin(r)./r  
contour(z,5)
```





# Salvando Gráficos

- Opções:
  - Print;
  - Save as
  - Via código:

`print -dformato nome.formato`

- png, jpg ou pdf



# Polinômios





# Polinômios no Octave

- Escrito com um vetor de coeficientes;
  - Exemplo:

$$6x^2 + 5x - 8$$

polinomio = [6 5 -8]



# Raízes do polinômio

- Determinar as raízes:
  - `roots(nome)`
  
- Polinômio a partir das raízes:
  - `poly(vetor)`

Exercício: Encontre a raiz do polinômio a seguir e depois recrie o polinômio a partir das raízes.

$$-9x^2 + 8x + 10$$



# Produto e divisão

- Produto:
  - $\text{conv}(\mathbf{p1}, \mathbf{p2})$
  
- Divisão:
  - $\text{deconv}(\mathbf{p1}, \mathbf{p2})$
  - Cuidado com a ordem!



# Avaliação de um polinômio

- `polyval(p,x)`

$$-9x^2 + 8x + 10$$

Para  $x = (0, 1, 2, 3, 4, 5)$



# Interpolação polinomial

`polyfit(x,y,n)`

- $x$  é o vetor que contém os valores de  $x$ ;
- $y$  é o vetor que contém os valores de  $y$ ;
- $n$  é o grau do polinômio desejado.
  
- Funções não lineares;

---

$$y = bx^m$$

`polyfit(log(x),log(y),n)`

---

$$y = be^{mx} \text{ ou } y = b10^{mx}$$

`polyfit(x,log(y),n)` ou `polyfit(x,log10(y),n)`

---

$$y = mlnx + b \text{ ou } y = mlogx + b$$

`polyfit(log(x),y,n)` ou `polyfit(log10(x),y,n)`

---

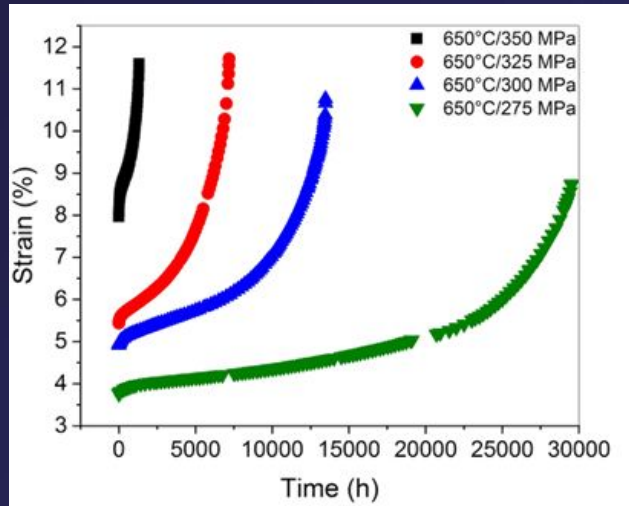
$$y = \frac{1}{mx+b}$$

`polyfit(x,1./y,n)`

---

# Aplicação da interpolação polinomial

Figura 1 - Resultado do ensaio de fluência.



Fonte: Bagui, S., Sahu, B.P., Laha, K. et al.  
 Creep Deformation Behavior of Inconel 617  
 Alloy in the Temperature Range of 650 °C to  
 800 ° C. Metall Mater Trans A 52, 94 -107  
 (2021). <https://doi.org/10.1007/s11661-020-06066-8>

$$\dot{\epsilon}_m = k_1 \sigma^{n_1}$$

$$\log(\dot{\epsilon}_m) = n_1 \log(k_1 \sigma)$$

$$\log(\dot{\epsilon}_m) = n_1 (\log(k_1) + \log(\sigma))$$

$$\text{poly fit}(\log(\sigma), \log(\dot{\epsilon}_m), 1)$$



# Técnicas de cálculo

# Mínimos

- `fminbnd(função, intervalo)`

$$f(x) = x^2 + 2x \quad \text{em } (-2,2)$$

- Função anônima

```
[xmin ymin] = fminbnd(@(x)(x^2+2*x),-2,2)
```

- Função nomeada

```
function y=poly1(x)
```

```
    y=x^2 +2 * x;
```

```
endfunction
```

```
[xmin ymin] = fminbnd('poly1',-2,2);
```





# Máximo

- `fminbnd(função, intervalo)`

$$f(x) = x^2 + 2x \text{ em } (-2,2)$$

- Função anônima

```
[xmax ymax] =  
fminbnd(@(x)(-x^2-2*x),-2,2)
```

- Função nomeada

```
function y=poly1(x)  
    y=-x^2-2 * x;  
endfunction  
[xmax ymax] = fminbnd('poly1',-2,2);
```

Os valores de ymax estão multiplicados por -1, por que toda a função também foi.



# Derivação

- Derivada de polinômios:

`polyder(polinômio)`

$$f(x) = x^2 + 2x$$

`poli= [1 2 0]`  
`polyder(poli)`

# Derivação

- Derivada de outras funções:

$$\frac{dy}{dx} \approx \frac{y_2 - y_1}{x_2 - x_1}$$

- Vetor com valores de x e vetor com valores de y;
- diff(x) e diff(y)



# Derivação

- Derivada de outras funções:

$$y = \sin(\cos(t))$$

```
function y=funcao(t)
    y=sin(cos(t))
endfunction
x=1:0.01:2
y=funcao(x)
dfdx=diff(y)./diff(x)
```

ou

```
x=1:0.01:2
y=@(x)(sin(cos(t)))
dfdx=diff(y)./diff(x)
```



# Integração

- Integração de polinômios: `polyint(polínomio)`:
  - Integral indefinida;
  - No Octave:

Integral de  $2x^2 + 3x + 4$



# Integração

- Quadratura de Simpson:
  - Integral simples: `quad(funcao, intervalo)`
  - Integral dupla: `dblquad(função, intervalo1, intervalo2)`
  - Integral tripla: `triplequad(quad, intervalo1, intervalo2, intervalo3)`

- Exemplo de integral simples:

```
function y=funcao(x)
    y=log(x);
endfunction
quad('funcao', 0,2)
```

- Exemplo de integral tripla:

```
triplequad(@(x,y,z)(x+2*y-ez ),0,1,0,2,0,3)
```



# Integração

- Outros métodos:
  - `integral(funcao, intervalo)`
  - `integral2(funcao, intervalo1, intervalo2)`
  - `integral3(função, intervalo1, intervalo2, intervalo3)`
- Qual a melhor maneira?



# Dúvidas?