

# Tutorial Básico de MATLAB

## Introdução

O MATLAB (MATrix LABoratory) é uma plataforma de cálculo numérico amplamente utilizada em engenharia, física e outras áreas científicas. Este tutorial introdutório é destinado a iniciantes que nunca usaram o MATLAB antes. Ele cobre conceitos fundamentais, comandos básicos e exemplos práticos.

## 1. Interface do MATLAB

Quando você abre o MATLAB, encontra os seguintes elementos principais:

- **Command Window:** Onde você digita comandos e executa códigos.
- **Workspace:** Mostra as variáveis que você criou e seus valores.
- **Editor:** Utilizado para escrever scripts e funções.
- **Current Folder:** Exibe os arquivos do diretório de trabalho atual.

## 2. Operações Básicas

No MATLAB, você pode realizar operações matemáticas diretamente no **Command Window**:

### 2.1. Operações Matemáticas Simples

```
x = 5; % Atribuição de valor a uma variável
y = 10;
z = x + y; % Soma
w = x * y; % Multiplicação
```

### 2.2. Matrizes

MATLAB foi projetado para lidar com matrizes com facilidade.

```
A = [1, 2; 3, 4]; % Matriz 2x2
B = [5, 6; 7, 8];
C = A + B; % Soma de matrizes
D = A * B; % Multiplicação de matrizes
```

### 2.3. Gráficos

```
x = linspace(0, 2*pi, 100); % Cria um vetor de 100 pontos entre 0 e 2*pi
y = sin(x);
plot(x, y); % Plota o gráfico da função seno
title('Gráfico de Seno');
xlabel('x');
ylabel('sin(x)');
```

## 3. Estruturas Básicas

### 3.1. Scripts

Scripts são arquivos que contêm uma série de comandos MATLAB. Crie um script clicando em **New Script** no Editor.

#### Exemplo:

```
% Meu primeiro script
x = 0:0.1:10; % Vetor de 0 a 10 com passo 0.1
y = x.^2;    % Eleva cada elemento de x ao quadrado
plot(x, y);
title('Gráfico de x^2');
```

Salve o script com a extensão .m e execute-o na **Command Window** digitando seu nome (sem a extensão).

### 3.2. Loops

Loops permitem executar comandos repetidamente.

#### Loop for:

```
for i = 1:10
    disp(i); % Mostra os valores de 1 a 10
end
```

#### Loop while:

```
i = 1;
while i <= 10
    disp(i);
    i = i + 1;
end
```

### 3.3. Condicionais

```
x = 5;
if x > 0
    disp('x é positivo');
elseif x == 0
    disp('x é zero');
else
```

```
disp('x é negativo');  
end
```

## 4. Funções Úteis

### 4.1. `fminsearch`

Minimiza uma função sem restrições.

```
f = @(x) x^2 + 3*x + 2; % Função objetivo  
x0 = 0; % Ponto inicial  
x_min = fminsearch(f, x0);
```

### 4.2. `simulannealbnd`

Realiza minimização global usando o algoritmo de recozimento simulado.

```
f = @(x) x^2 + 3*x + 2;  
x0 = 0;  
x_min = simulannealbnd(f, x0, -10, 10);
```

### 4.3. `plot`

Cria gráficos bidimensionais.

```
x = linspace(0, 10, 100);  
y = sin(x);  
plot(x, y);
```

## 5. Dicas para Iniciantes

1. **Comentar seu código:** Use % para adicionar comentários e explicar cada etapa.
2. **Salvar seu trabalho:** Salve scripts e variáveis importantes.
3. **Explorar a documentação:** Digite help seguido do nome da função para aprender mais sobre ela.
4. **Usar o MATLAB Online:** Caso não tenha o software instalado, experimente a versão online.

## 6. Recursos Adicionais

- **Documentação Oficial:** [MATLAB Documentation](#)
- **Tutoriais em Vídeo:** Disponíveis no YouTube e no site da MathWorks.
- **MATLAB Online:** Experimente diretamente no navegador.

Este tutorial cobre apenas os conceitos básicos do MATLAB. Para tarefas mais complexas, explore tópicos como criação de funções personalizadas, uso de pacotes especializados e integração com outras linguagens de programação.