Pontifícia Universidade Católica
do Rio de Janeiro

Introdução à programação com MATLAB

Parte II: Estruturas de controle

Fabricio Fernández

✉ fabriciof@puc-rio.br

Sumário

Arquivos fontes e apresentação

Links de download

https://github.com/fabricix/introduction-to-matlab

Instalação usando o comando git (opcional)

```
git clone https://github.com/fabricix/introduction-to-matlab.git
```

Estruturas de controle

Permitem modificar a ordem em que instruções são executadas dentro do programa

Iterativas (loop)

Definem partes de código que serão executadas repetidamente (`for`, `while`)

Condicionais

Definem que parte do código será executado durante a execução do programa (`if`, `switch`)

Palavras-chaves para estruturas de controle no MATLAB

Os loops usam uma palavra-chave `for` ou `while`, e as instruções condicionais usam `if` ou `switch`.

Palavras-chave adicionais fornecem controle mais preciso sobre o fluxo do programa:

`if, elseif, else`	Executa as instruções se a condição for verdadeira
`switch, case, otherwise`	Executa um dos vários grupos de instruções
`for`	Executa as instruções um número especificado de vezes
`while`	Executa as instruções enquanto a condição for verdadeira
`try, catch`	Executa instruções e capture os erros resultantes
`break`	Termina a execução do loop `for` ou `while`
`return`	Retorna o controle para o script ou função
`continue`	Passa para a próxima iteração do loop `for` ou `while`
`pause`	Para a execução do MATLAB temporariamente
`parfor`	Modo paralelo do loop `for`
`end`	Termina o bloco de código

Instruções Iterativas

Os loops usam uma palavra-chave `for` ou `while` para executar repetidamente uma parte do código.

loop for

```
x = zeros(1,10);
for n = 2:6
    x(n) = 2 * x(n - 1);
end
```

- contador
- início e fim
- palavras-chaves

Instruções Iterativas

Os loops usam uma palavra-chave `for` ou `while` para executar repetidamente uma parte do código.

loop for

```
% altura
height = 10;

% número de pontos
npoints = 100;

% peso específico do solo (kg/m3 *10 N/kg)
specific_weight = 2500*10;

% discretizando a altura do solo
soil_height = linspace(0,height,npoints);

% define se um vetor para armazenar cada valor
initial_stress = zeros(1,npoints);

% calcula a tensão em cada ponto
for i = 1:npoints
      initial_stress(i) = soil_height(i)*specific_weight;
end
```

Instruções Iterativas

Os loops usam uma palavra-chave `for` ou `while` para executar repetidamente uma parte do código.

loop for aninhado

```
A = zeros(5,100);
for m = 1:5
    for n = 1:100
        A(m, n) = 1/(m + n - 1);
    end
end
```

- loop dentro de outro (aninhado)
- contadores (x2)
- início e fim(x2)
- palavras-chaves (x2)

Instruções Iterativas

Os loops usam uma palavra-chave `for` ou `while` para executar repetidamente uma parte do código.

loop while

```
% loop principal da simulação
while ( niter<niter_max )

    % obtém o erro da solução (tolerância)
    erro_solucao = update_approximation();

    % incrementa a iteração em 1
    niter = niter + 1;

end
```

Instruções Iterativas

Os loops usam uma palavra-chave `for` ou `while` para executar repetidamente uma parte do código.

loop while com controle de saída usando break

```
% loop principal da simulação
while ( niter<niter_max )

     % obtém o erro da solução (tolerância)
     erro_solucao = update_approximation();

     % incrementa a iteração em 1
     niter = niter + 1;

     if erro_solucao<erro_adm
          break;
     end
end
```

Instruções Condicionais

Definem que parte do código será executado durante a execução do programa (`if`, `switch`)

instrução if

```
% atualiza a porosidade
for i=1:nincre

    p(i) = atualiza_porosidade(po,devol(i));

    % verifica o valor obtido
    if p(i) <= poro_min
         p(i) = poro_min;
    end
end
```

- palavras-chaves
- valor de verdade

Instruções Condicionais

Definem que parte do código será executado durante a execução do programa (`if`, `switch`)

instrução if-else

```
% verificação de porosidade no solo

porosity = 0.2;
poro_min = 1e-4

if porosity <= 0
     disp('porosidade nula..')
     porosity = poro_min;
else
     disp('porosidade adequada..')
end
```

Instruções Condicionais

Definem que parte do código será executado durante a execução do programa (`if`, `switch`)

instrução if - elseif

```
% verificação de porosidade no solo (3)

porosity = 0.2;
poro_max = 0.99;
poro_min = 1e-4;

if porosity < poro_min
     disp('porosidade muito baixa')
     porosity = poro_min;

elseif porosity > poro_max
     disp('porosidade muito alta')
     pososity = poro_max;

else
     disp('porosidade adequada...')
end
```

Instruções Condicionais

Definem que parte do código será executado durante a execução do programa (`if`, `switch`)

instrução switch

```
material = "linear";

switch material

     case "linear"

          stress = material_linear(strain);

     case "nao_linear"

          stress = material_nao_linear(strain);

     case "viscoso"

          stress = material_viscous(strain);

     otherwise

          disp("nenhum material identificado...");
end
```

Dúvidas