

//Este programa permite ao utilizador informar um conjunto de valores a ser usado na projeção de gráficos bidimensionais (2D) e tridimensionais (3D). O programa fornece a possibilidade do utilizador selecionar uma em três funções de projeção 2D, as quais podem ser utilizadas para o desenho de gráficos a serem visualizados de forma individual, ou conjunta. Além disso, o programa ainda fornece a possibilidade do utilizador realizar uma projeção 3D, consoante uma função de projeção que faz a relação conjunta dos valores do eixo X com os valores do eixo Y obtidos pela função de projeção 2D selecionada anteriormente. A função de projeção 3D fornece o cálculo da profundidade (altura) a ser visualizada para cada um dos pares de pontos (x,y).

```
//Esta função representa a função matemática  $y=2x^2 - 5$ 
function y=f(x)
    y = 2*x.^2 - 5
endfunction
```

```
//Esta função representa a função matemática  $y=(3x^2 - 4)/6$ 
function y=g(x)
    y = (3*x.^2 - 4)/6
endfunction
```

```
//Esta função representa a função matemática  $y=\sin(x) + 3$ 
function y=h(x)
    y = sin(x) + 3
endfunction
```

```
//Esta função representa a função matemática  $z=x^2 + y^2$ 
function z=profundidade(x,y)
    z = x.^2 + y.^2
endfunction
```

```
/*
```

Alternativa à utilização conjunta das funções calculaProjecao2DInterna() e calcularFuncaoProjecao2D().

```
//Esta função realiza o cálculo dos valores do eixo y para a função(ões) de projeção 2D selecionada(s). A estratégia de execução adotada utiliza recursividade para executar todas as funções de projeção 2D em uma única invocação [opcaoUnitaria == 4].
function [opcaoUnitaria,varargout]=calcularFuncaoProjecao2D(opcaoUnitaria, x)
```

```
    select opcaoUnitaria

        case 1 then varargout(1) = f(x)

        case 2 then varargout(1) = g(x)

        case 3 then varargout(1) = h(x)

        case 4 then
            for i=1:3

[opcaoUnitariaInterna,varargout(i)]=calcularFuncaoProjecao2D(i,x)
                end
            end
endfunction
```

*/

/*Esta função é uma função utilitária, a qual permite o cálculo dos valores do eixo y para cada uma das funções de projeção 2D suportadas:

$f(x) = 2x^2 - 5$ [opcaoUnitaria == 1]

$g(x) = (3x^2 - 4)/6$ [opcaoUnitaria == 2]

$h(x) = \sin(x) + 3$ [opcaoUnitaria == 3]

A função calculaProjecao2DInterna() assume que o conjunto de valores [i.e. vetor x] utilizado para os cálculos já está previamente definido, a implicar sua invocação por intermédio da função calcularFuncaoProjecao2D().

*/

```
function y=calculaProjecao2DInterna(opcaoUnitaria)
```

```
    select opcaoUnitaria
```

```
        case 1 then y = f(x)
```

```
        case 2 then y = g(x)
```

```
        case 3 then y = h(x)
```

```
    else y = -1
```

```
    end
```

```
endfunction
```

//Esta função realiza o cálculo dos valores do eixo y para a função(ões) de projeção 2D selecionada(s). A estratégia de execução adotada utiliza a função utilitária calculaProjecao2DInterna(), a qual é executada iterativamente para executar todas as funções de projeção 2D em uma única invocação [opcaoUnitaria == 4].

```
function [opcaoUnitaria,varargout]=calcularFuncaoProjecao2D(opcaoUnitaria, x)
```

```
    if opcaoUnitaria > 0 && opcaoUnitaria < 4 then
```

```
        varargout(1) = calculaProjecao2DInterna(opcaoUnitaria)
```

```
    elseif opcaoUnitaria == 4
```

```
        for i = 1:3
```

```
            varargout(i) = calculaProjecao2DInterna(i)
```

```
        end
```

```
    end
```

```
endfunction
```

/*Esta função permite o desenho de gráficos associados às projeções gráficas de uma das funções de projeção 2D a seguir:

$f(x) = 2x^2 - 5$

$g(x) = (3x^2 - 4)/6$

$h(x) = \sin(x) + 3$

A função recebe os seguintes argumentos de entrada:

limpa -> indica se a janela gráfica deve ser limpa antes de realizar a projeção desejada. limpa == %t (verdadeiro) para realizar a limpeza; limpa == %f (falso) caso contrário;

linhas -> indica o número de linhas necessárias para o desenho de diferentes projeções na mesma janela gráfica;

colunas -> indica o número de colunas necessárias para o desenho de diferentes projeções na mesma janela gráfica;

coluna -> indica qual a coluna utilizada para o desenho da projeção desejada;

opcaoDimensional -> indica qual o tipo de projeção a ser desenhada.
 opcaoDimensional == 2 para realizar o desenho de uma projeção 2D (função plot);
 opcaoDimensional = 3 para realizar o desenho de uma projeção 3D (função mesh a utilizar a função de projeção 3D $z=x^2 + y^2$);

varargin -> indica os conjuntos de dados utilizados para o desenho a projeção gráfica desejada, bem como um possível título da projeção a ser desenhada. Os conjuntos de dados devem ser informados na seguinte ordem: x -> y -> z, onde x representa o vetor/matriz de valores a ser projetado no eixo X; y representa o vetor/matriz de valores a ser projetado no eixo Y; e z representa o vetor/matriz de valores a ser projetado no eixo Z. O conjunto de caracteres para definir o título do gráfico a ser desenhado é informado como último argumento de entrada variável da função.

*/

```
function desenhaProjecao(limpa,linhas,colunas,coluna,opcaoDimensional,varargin)
```

```
    if limpa == %t then
```

```
        clf()
```

```
    end
```

```
    subplot(linhas,colunas,coluna)
```

```
    if opcaoDimensional == 2 then
```

```
        plot(varargin(1),varargin(2))
```

```
    elseif opcaoDimensional == 3
```

```
        mesh(varargin(1), varargin(2), varargin(3))
```

```
    end
```

```
    if (opcaoDimensional+1) == length(varargin) then
```

```
        title(string(varargin(opcaoDimensional+1)))
```

```
    end
```

```
endfunction
```

/*Esta função permite o desenho de gráficos associados às projeções gráficas de uma das funções de projeção 2D a seguir:

$$f(x) = 2x^2 - 5$$

$$g(x) = (3x^2 - 4)/6$$

$$h(x) = \sin(x) + 3$$

A função recebe os seguintes argumentos de entrada:

opcaoDimensional -> vetor a indicar qual projeção será deenhada para cada uma das funções. Cada elemento do vetor pode assumir um dos seguintes valores: 2 -> 2D; 3 -> 3D;

linhas -> indica o número de linhas necessárias para o desenho de diferentes projeções na mesma janela gráfica;

pares -> indica se o conjunto de dados utilizado para realizar as projeções é invocado aos pares. %t (verdadeiro) caso o conjunto de dados seja informado aos pares; %f (falso) caso o conjunto de dados utilize um único conjunto de valores no eixo X para as múltiplas projeções realizadas;

titulos -> indica títulos, os quais podem ser incluídos nos gráficos a representar as projeções desenhadas;

varargin -> indica os conjuntos de dados utilizados para o desenho da(s) projeção(ões) gráfica(s) desejada(s), Os conjuntos de dados devem ser informados na seguinte ordem: x -> y, onde x representa o vetor/matriz de valores a ser projetado no eixo X; e y representa o vetor/matriz de valores a ser projetado no eixo Y. O conjunto de valores a ser projetado no eixo Z é calculado com base na função de profundidade (altura) $z = x^2 + y^2$.

*/

```
function desenhaProjecoes(opcaoDimensional,linhas, pares,titulos, varargin)
```

```
    incremento = 1
    inicio = 1
    limpa = %t
    numeroProjecoes = 1
```

```
    if pares == %t then
        incremento = 2
        numeroProjecoes = length(varargin)/2
    elseif pares == %f
        numeroProjecoes = length(varargin) - 1
    end
```

```
    for i =inicio:incremento:length(varargin) - 1
```

```
        indiceProjecao = getIndiceProjecao(pares,i,incremento)
```

```
        if opcaoDimensional(indiceProjecao) == 3 then
            if pares == %t then
                [matrixX,matrixY] = meshgrid(varargin(i),varargin(i+1))
            else
                [matrixX,matrixY] = meshgrid(varargin(1),varargin(i+1))
            end
```

```

        matrixZ = profundidade(matrixX, matrixY)

desenhaProjecao(limpa,linhas,numeroProjecoes,indiceProjecao,opcaoDimensional(ind
iceProjecao),matrixX, matrixY, matrixZ,getTitulo(titulos,indiceProjecao))

elseif opcaoDimensional(indiceProjecao) == 2

    if incremento == 1 then

desenhaProjecao(limpa,linhas,numeroProjecoes,indiceProjecao,opcaoDimensional(ind
iceProjecao),varargin(1), varargin(i+1),getTitulo(titulos,indiceProjecao))
        else

desenhaProjecao(limpa,linhas,numeroProjecoes,indiceProjecao,opcaoDimensional(ind
iceProjecao),varargin(i), varargin(i+1),getTitulo(titulos,indiceProjecao))
            end
        end

        limpa = %f
    end
endfunction

//Esta função retorna o índice da projeção a ser realizada, o qual é utilizado
para a obtenção do tipo de projeção e para definição da posição de desenho da
projeção.
function indiceProjecao=getIndiceProjecao(pares,i,incremento)

    if pares == %t && i > 1 then
        indiceProjecao = i - floor(i/incremento)
    else
        indiceProjecao = i
    end

endfunction

//Esta função retorna o título correspondente ao índice informado. Caso o índice
não existe, o título retornado é um conjunto de caracteres vazio.
function titulo=getTitulo(titulos,indice)

    if length(titulos) >= indice then
        titulo = titulos(indice)
    else
        titulo = ""
    end
endfunction

continuar = 'sim'

while continuar == 'sim'

    x = input("Informe o conjunto de valores [vetor] a ser utilizado para o
desenho do(s) gráfico(s) da(s) função(ões) selecionada(s): ")

```

```
mprintf("Selecione a função de projeção 2D para desenho do(s) gráfico(s): \n
1 -  $y=2x^2 - 5$ \n 2 -  $y=(3x^2 - 4)/6$ \n 3 -  $y = \sin(x) + 3$ \n 4 - todas\n")
opcaoUnitaria=input("Valor Selecionado: ")

opcaoDimensional = input("Deseja realizar a(s) projeção(ões) em 2D ou em 3D
(projeção com  $z = x^2 + y^2$ ) [2 - 2D; 3 - 3D]? ")

titulos = input("Informe um vetor com o(s) título(s) para o(s) gráfico(s)
[Pressione enter no caso de não possuir títulos]: ")

if opcaoUnitaria > 0 && opcaoUnitaria < 4 then

    [opcaoUnitaria,y] = calcularFuncaoProjecao2D(opcaoUnitaria, x)

    desenhaProjecoes(opcaoDimensional,linhas=1,pares=%t,titulos,x,y)

elseif opcaoUnitaria == 4

    [opcaoUnitaria,fY,gY,hY] = calcularFuncaoProjecao2D(opcaoUnitaria, x)

    desenhaProjecoes(opcaoDimensional,linhas=1,pares=%t,titulos,x,fY,gY,hY)

end

continuar = input("Deseja continuar a realizar projeções? [sim/nao]: ", "s")
end

mprintf("Tenha um bom dia :-D")
```

8/13/2020

DocuSigned by:

Jefferson Luiz Rodrigues Souza

AB0CF1C511CA41A...