Use a método da bisseção para estimar a raízes de

$$f(x) = \cos(x) - x \cdot e^{x}$$

no intervalo $-2 \leq x \leq 0$, utilizando com critério de parada $\varepsilon_s < 1\ \%$.

A partir da solução gráfica do Exemplo 1.1, percebe-se que a função muda de sinal entre -2 e 0. Assim, o primeiro passo é calcular a aproximação inicial da raiz x_r .Com $x_l^1 = -2$ e $x_u^1 = 0$, x_r vale

$$x_r^1 = \frac{x_u^1 + x_l^1}{2} = \frac{0-2}{2} = -1$$

O próximo passo é calcular o erro aproximado relativo percentual:

$$\varepsilon_a^1 = \frac{1}{2}\left|\frac{x_u^1 - x_l^1}{x_r^1}\right| \times 100\% = \frac{1}{2}\left|\frac{0+2}{-1}\right| \times 100 = 100\%$$

Como, $\varepsilon_a^1 > 5\ \%$, iniciam-se os preparativos para a próxima iteração calculando-se o valor da função no extremo inferior e no ponto médio:

$$f\left(x_l^1\right) = \cos(-2) + 2 \cdot e^{-2} = -0{,}1454763$$

$$f(x_r^1) = \cos(-1) - 1 \cdot e^{-1} = 0{,}9081817$$

Como o produto entre os dois valores é menor que zero, a troca de sinal ocorre no subintervalo entre o extremo inferior e o ponto médio, assim redefine-se a extremidade superior com o valor do ponto médio. Assim,

$$x_u^2 = x_r^1 = -1$$

e

$$x_l^2 = x_l^1 = -2$$

A partir do novo intervalo, uma segunda estimativa da raiz é calculada:

$$x_r^2 = \frac{x_u^2 + x_l^2}{2} = \frac{-1-2}{2} = -1{,}5$$

que representa um erro aproximado relativo de 33,33%:

$$\varepsilon_a^2 = \frac{1}{2}\left|\frac{x_u^2 - x_l^2}{x_r^2}\right| \times 100\% = \frac{1}{2}\left|\frac{-1+2}{-1{,}5}\right| \times 100 = 33{,}33\%$$

Como a extremidade inferior não se alterou

$$f\left(x_l^2\right) = f\left(x_l^1\right) = -0{,}1454763$$

e o novo valor da função no ponto médio é

$$f(x_r^2) = \cos(-1{,}5) - 1{,}5 \cdot e^{-1{,}5} = 0{,}4054324$$

Assim, como $f\left(x_l^2\right) f(x_r^2) < 0$ tem-se

$$x_u^3 = x_r^2 = -1{,}5$$

e

$$x_l^3 = x_l^2 = -2$$

A partir do novo intervalo, a terceira estimativa da raiz é calculada:

$$x_r^3 = \frac{x_u^3 + x_l^3}{2} = \frac{-1{,}5-2}{2} = -1{,}75$$

que representa erro aproximado relativo de 14,29%.

O método é repetido até a sétima iteração, quando se obtém a raiz

$$x = -1{,}8125$$

que apresenta a precisão desejada. A Tabela 1.1, contém todos os valores calculados para a determinação da raiz.

Tabela 1.1 – Processo de obtenção da raiz para a função do Exemplo 1.2

Iteração i	x_l^i	x_u^i	x_r^i	$f(x_l^i)$	$f(x_r^i)$	$f(x_l^i) \cdot f(x_r^i)$	ε_a^i **%**
1	-2	0	-1	-0,145476	0,908182	-0,132119	100
2	-2	-1	-1,5	-0,145476	0,405432	-0,058981	33,33
3	-2	-1,5	-1,75	-0,145476	0,125858	-0.018309	14,29
4	-2	-1,75	-1,875	-0,145476	-0.011993	0.001745	6,67
5	-1,875	-1,75	-1,8125	-0,0119929	0,056525	-0.000678	3,45
6	-1,875	-1,8125	-1,84375	-0,0119929	0.022147	-0.000265	1,69
7	-1,875	-1,84375	-1,859375	-0,0119929	0.005045	-0.000061	0,84

```
--> f = 'cos(x)-x*exp(x)'

 f  =

  "cos(x)-x*exp(x)"

--> [raiz, i]=bissecao(f, -2, 0, 1)

Iter    Erro aprox.%    Raiz         xl           xu

1       100.000000      -1.000000    -2.000000    0.000000

2       33.333333       -1.500000    -2.000000    -1.000000

3       14.285714       -1.750000    -2.000000    -1.500000

4       6.666667        -1.875000    -2.000000    -1.750000

5       3.448276        -1.812500    -1.875000    -1.750000

6       1.694915        -1.843750    -1.875000    -1.812500

7       0.840336        -1.859375    -1.875000    -1.843750

 raiz  =

  -1.859375

 i  =

   7.
```

Exercício 3

Uma opção é carregar os dados pelo console, outra opção é utilizar um script

OPÇÃO 1 - CONSOLE

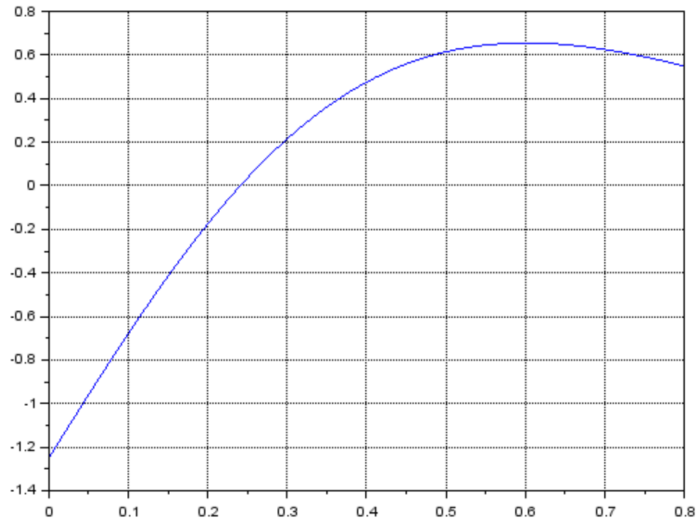
```
--> exec('D:\Users\Marco\Documents\2020_1\ccientifica\bissecaoslide.sci', -1)

--> eo = 8.9d-12; F = 1.25; q = 2d-5; r = 0.85;

--> x = 0:0.01:0.8;

--> f = (1/(4*%pi*eo))*(q^2*x)./((x.^2+r^2)^(3/2))-F;

--> plot(x,f);xgrid;
```




```
--> f = '(1/(4*%pi*eo))*(q^2*x)./((x.^2+r^2)^(3/2))-F';


--> [raiz, i]=bissecao(f, 0, 0.4)
```

Iter	Erro aprox.%	Raiz	xl	xu
1	100.000000	0.200000	0.000000	0.400000
2	33.333333	0.300000	0.200000	0.400000
3	20.000000	0.250000	0.200000	0.300000
4	11.111111	0.225000	0.200000	0.250000
5	5.263158	0.237500	0.225000	0.250000
6	2.564103	0.243750	0.237500	0.250000
7	1.298701	0.240625	0.237500	0.243750
8	0.645161	0.242188	0.240625	0.243750
9	0.323625	0.241406	0.240625	0.242188

10	0.162075	0.241016	0.240625	0.241406
11	0.080972	0.241211	0.241016	0.241406
12	0.040502	0.241113	0.241016	0.241211
13	0.020255	0.241064	0.241016	0.241113
14	0.010129	0.241040	0.241016	0.241064
15	0.005064	0.241052	0.241040	0.241064
16	0.002532	0.241046	0.241040	0.241052
17	0.001266	0.241043	0.241040	0.241046
18	0.000633	0.241042	0.241040	0.241043
19	0.000317	0.241042	0.241042	0.241043
20	0.000158	0.241043	0.241042	0.241043
21	0.000079	0.241043	0.241043	0.241043

```
 raiz  =


   0.2410429

 i  =


   21.
```

OPÇÃO 2 - SCRIPT

```
// CARREGAR ANTES A FUNÇÃO DA BISSECÇÃO
eo = 8.9d-12; F = 1.25; q = 2d-5; r = 0.85
x = 0:0.01:0.8
fun = '(1/(4*%pi*eo))*(q^2*x)./((x.^2+r^2)^(3/2))-F'
f = evstr(fun)
plot(x,f);xgrid;
xl = input("Entre com xl: ")
xu = input("Entre com xu: ")
[raiz, i]=bissecao(fun, xl, xu)
printf("Raiz x = %.6f m encontrada após %d iterações",raiz,i)
```

Após ser executado, o script apresenta o gráfico e solicita pelo console xl e xu.

```
function [raiz, iter]=falsapos(funcao, xl, xu, es)
  // Cálculo das raizes pelo processo da falsa posição
  // function [raiz,iter]=falsapos(funcao, xl, xu, es)
  // onde raiz é a raiz procurada de funcao
  // iter é o num de iterações para o erro especificado
  // funcao é a função de entrada literal em x
  // xl é o limite inferior do intervalo de busca
  // xu é o limite superior do intervalo de busca
  // es é o criterio de parada que é opcional
  x = xu;    fu = evstr(funcao);
  x = xl;    fl = evstr(funcao);
  if (fu*fl >= 0) then
    error("Nenhuma raiz no intervalo dado.")
  end
  i = 0; ea=100; xr_novo = xl;
  // se es nao foi estabelecido usa 0.0001%
  if argn(2) < 4 then
    es = 0.0001;
  end
  // inicio do processo iterativo
  printf("Iter\tErro aprox.%%\tRaiz\t\txl\t\txu\n");
  while ea > es do
    xr_velho = xr_novo;
    xr_novo = xu + fu*(xl-xu)/(fu-fl);
    if xr_novo ~=0 then // xr_novo não pode ser zero
      ea = abs((xr_novo - xr_velho)/xr_novo)*100;
    end
    i=i+1;
    printf("%d\t%f\t%f\t%f\t%f\n",i,ea,xr_novo,xl,xu);
    x=xr_novo;      fr = evstr(funcao);
    x = xl;
    fl = evstr(funcao);
    if(fl*fr < 0) then
      xu = xr_novo;
      x = xu;
      fu = evstr(funcao);
    elseif(fl*fr > 0) then
      xl = xr_novo;
      x = xl;
      fl = evstr(funcao);
    else
      break;
    end
  end
  raiz = xr_novo;
  iter = i;
endfunction
```

```
-> eo = 8.9d-12; F = 1.25; q = 2d-5; r = 0.85;

--> f = '(1/(4*%pi*eo))*(q^2*x)./((x.^2+r^2)^(3/2))-F';

--> [raiz, iter]=falsapos(f, 0, 0.4)

Iter    Erro aprox.%    Raiz        xl          xu
1       100.000000      0.289749    0.000000    0.400000
2       14.472792       0.253116    0.000000    0.289749
3       3.816314        0.243811    0.000000    0.253116
4       0.888275        0.241665    0.000000    0.243811
5       0.200248        0.241182    0.000000    0.241665
6       0.044810        0.241074    0.000000    0.241182
7       0.010011        0.241050    0.000000    0.241074
8       0.002236        0.241044    0.000000    0.241050
9       0.000499        0.241043    0.000000    0.241044
10      0.000111        0.241043    0.000000    0.241043
11      0.000025        0.241043    0.000000    0.241043
 raiz  =


   0.2410428
 iter  =


   11.
```