MÉTODOS COMPUTACIONAIS PARA ENGENHARIA CIVIL

LISTA DE EXERCÍCIOS – RAÍZES DE FUNÇÕES NÃO-LINEARES

**QUESTÃO 1:**

function [fy] = profcritica(y)

%Calculo da profundidade critica para um canal trapezoidal

%Valido para Q = 20m³/s e g = 9.81m/s²

B = 3 + y;

Ac = 3\*y + 0.5\*y^2;

fy = 1 - 20^2\*B/(9.81\*Ac^3);

end

(a) Pelo gráfico, observa-se que o intervalo que convém, i.e., para valores positivos de y, tem como y = 0 e y = 2 como limites inferior e superior, respectivamente.

>> ezplot('profcritica')

>> grid on



(b) Pelo método da Bisseção:

>> bissec('profcritica',.5,2.5,1,15)

ii xi F(xi) xu F(xu) xr F(xr) ea

1 1.500000 -32.258215 2.500000 0.813032 1.500000 -0.030946 66.666667

2 1.500000 -0.030946 2.000000 0.813032 2.000000 0.601809 25.000000

3 1.500000 -0.030946 1.750000 0.601809 1.750000 0.378909 14.285714

4 1.500000 -0.030946 1.625000 0.378909 1.625000 0.206927 7.692308

5 1.500000 -0.030946 1.562500 0.206927 1.562500 0.097956 4.000000

6 1.500000 -0.030946 1.531250 0.097956 1.531250 0.036261 2.040816

7 1.500000 -0.030946 1.515625 0.036261 1.515625 0.003383 1.030928

8 1.507813 -0.030946 1.515625 0.003383 1.507813 -0.013595 0.518135

**ans =**

**1.5078**

(c) Pelo método da Falsa Posição:

>> falsapos('profcritica',.5,2.5,1,15)

ii xi F(xi) xu F(xu) xr F(xr) ea

1 0.500000 -32.258215 2.450831 0.813032 2.450831 0.799873 79.598760

2 0.500000 -32.258215 2.403629 0.799873 2.403629 0.786123 1.963793

3 0.500000 -32.258215 2.358342 0.786123 2.358342 0.771792 1.920300

4 0.500000 -32.258215 2.314919 0.771792 2.314919 0.756894 1.875778

5 0.500000 -32.258215 2.273311 0.756894 2.273311 0.741447 1.830293

6 0.500000 -32.258215 2.233468 0.741447 2.233468 0.725474 1.783922

7 0.500000 -32.258215 2.195340 0.725474 2.195340 0.709003 1.736747

8 0.500000 -32.258215 2.158880 0.709003 2.158880 0.692065 1.688862

9 0.500000 -32.258215 2.124038 0.692065 2.124038 0.674695 1.640364

10 0.500000 -32.258215 2.090766 0.674695 2.090766 0.656933 1.591358

11 0.500000 -32.258215 2.059017 0.656933 2.059017 0.638822 1.541955

12 0.500000 -32.258215 2.028743 0.638822 2.028743 0.620408 1.492269

13 0.500000 -32.258215 1.999896 0.620408 1.999896 0.601740 1.442417

14 0.500000 -32.258215 1.972429 0.601740 1.972429 0.582868 1.392520

15 0.500000 -32.258215 1.946296 0.582868 1.946296 0.563846 1.342698

**ans =**

**1.9463**

Observa-se que, neste caso, o método da bisseção convergiu mais rapidamente para a solução do que o método da falsa posição, para um mesmo erro a.

**QUESTÃO 2:**

O gráfico da derivada da função que representa a curva elástica resultante de uma viga uniforme sujeita a uma carga distribuída de forma linearmente crescente é dado por:



function [dy] = ddeflex2(x)

%Derivada da equacao da curva elastica resultante, dados:

L = 600; %L: comprimento da viga (cm)

wo = 2.5; %wo: carregamento linearmente crescente (kN/cm)

E = 50000; %E: modulo de deformacao longitudinal (kN/cm²)

I = 30000; %I: momento de inercia (cm4)

dy = 2.5/(120\*E\*I\*L)\*(-5\*x.^4 + 6\*L^2\*x.^2 - L^4);

end

Diante disso, pode-se observar que a raiz da função correspondente ao ponto de deflexão máxima está compreendida no intervalo entre x = 200cm e x = 300cm.

>> raiz = bissec('ddeflex2',200,300,0.01,1000)

ii xi F(xi) xu F(xu) xr F(xr) ea

1 250.000000 -0.001185 300.000000 0.000562 250.000000 -0.000327 20.000000

2 250.000000 -0.000327 275.000000 0.000562 275.000000 0.000119 9.090909

3 262.500000 -0.000327 275.000000 0.000119 262.500000 -0.000104 4.761905

4 262.500000 -0.000104 268.750000 0.000119 268.750000 0.000008 2.325581

5 265.625000 -0.000104 268.750000 0.000008 265.625000 -0.000048 1.176471

6 267.187500 -0.000048 268.750000 0.000008 267.187500 -0.000020 0.584795

7 267.968750 -0.000020 268.750000 0.000008 267.968750 -0.000006 0.291545

8 267.968750 -0.000006 268.359375 0.000008 268.359375 0.000001 0.145560

9 268.164063 -0.000006 268.359375 0.000001 268.164063 -0.000003 0.072833

10 268.261719 -0.000003 268.359375 0.000001 268.261719 -0.000001 0.036403

11 268.310547 -0.000001 268.359375 0.000001 268.310547 -0.000000 0.018198

12 268.310547 -0.000000 268.334961 0.000001 268.334961 0.000000 0.009098

**raiz =**

**268.3350**

Substituindo este valor na equação da curva elástica (código e gráfico abaixo) a fim de determinar o valor da deflexão máxima, tem-se que:

>> defmax = deflex2(raiz)

**defmax =**

**-0.5152**

function [y] = deflex2(x)

%Equacao da curva elastica resultante, dados:

L = 600; %L: comprimento da viga (cm)

wo = 2.5; %wo: carregamento linearmente crescente (kN/cm)

E = 50000; %E: modulo de deformacao longitudinal (kN/cm²)

I = 30000; %I: momento de inercia (cm4)

y = (2.5/(120\*E\*I\*L))\*(-x.^5 + 2\*L^2\*x.^3 - L^4\*x);

%y: deflexao da viga (cm) para uma posicao x (cm)

end



**QUESTÃO 3:**

Para a determinação da profundidade do tanque para um armazenamento de 30m³, escolheram-se os métodos, seja intervalar ou aberto, que convergiu para um menor erro para o mesmo número de iterações.

function [fx] = tanque(h)

%Calculo da profundidade h de agua no tanque (m)

V = 30; %V: volume do tanque (m³)

R = 3; %R: raio do tanque (m)

fx = V - pi\*h.^2\*(3\*R - h)/3;

end

****

(a) Pelo método da Falsa Posição:

>> raiz = falsapos('tanque',2,2.5,0.01,3)

ii xi F(xi) xu F(xu) xr F(xr) ea

1 2.025659 0.678469 2.500000 -12.542401 2.025659 0.031537 1.266699

2 2.026849 0.031537 2.500000 -12.542401 2.026849 0.001443 0.058697

3 2.026903 0.001443 2.500000 -12.542401 2.026903 0.000066 0.002685

**raiz =**

**2.0269**

(b) Pelo método de Newton-Raphson:

function [dfx] = dtanque(h)

%Derivada da função tanque(h)

V = 30; %V: volume do tanque (m³)

R = 3; %R: raio do tanque (m)

dfx = pi\*h.^2/3 - 2\*pi\*h\*(3\*R - h)/3;

end

>> raiz = newton\_raph('tanque','dtanque',3,0.001,3)

iter xr fun(xr) dfun(xr) ea

1 2.0610 -0.8669 -25.5045 45.5581

2 2.0270 -0.0034 -25.3004 1.6769

3 2.0269 -0.0000 -25.2995 0.0067

**raiz =**

**2.0269**

**QUESTÃO 4:**

O gráfico abaixo corresponde a função para uma deflexão de 9mm da linha central de uma viga em balanço submetida a uma carga distribuída ao longo de seu comprimento.

function [y] = deflex4(x)

%Calculo da deflexao y da linha central da viga em funcao da posicao x

L = 3; %L: comprimento da viga (m)

wo = 15; %wo: carregamento linearmente crescente (kN/m)

E = 70e6; %E: modulo de deformacao longitudinal (kN/m²)

I = 52.9e-6; %I: momento de inercia (m4)

y = 9\*10^-3 - (wo\*L/(3\*pi^4\*E\*I))\*((48\*L^3\*cos(pi\*x/(2\*L)) - 48\*L^3 + 3\*pi^3\*L\*x^2 - pi^3\*x^3));

end



Para a utilização do método de Newton-Raphson, é necessário a derivada da função que é dada pelo código abaixo.

function [dy] = ddeflex4(x)

%Derivada da deflexao y da linha central da viga em funcao da posicao x

L = 3; %L: comprimento da viga (m)

wo = 15; %wo: carregamento linearmente crescente (kN/m)

E = 70e6; %E: modulo de deformacao longitudinal (kN/m²)

I = 52.9e-6; %I: momento de inercia (m4)

dy = (wo\*L/(3\*pi^4\*E\*I))\*(-(pi\*x/(2\*L))\*48\*L^3\*sin(pi\*x/(2\*L)) + 6\*pi^3\*L\*x - 3\*pi^3\*x^2);

end

(a) Pelo método de Newton-Raphson:

>> raiz = newton\_raph('deflex4','ddeflex4',1.8,0.001,1000)

iter xr fun(xr) dfun(xr) ea

1 1.8959 0.0005 -0.0147 5.0605

2 1.9312 0.0003 -0.0158 1.8269

3 1.9496 0.0002 -0.0164 0.9440

4 1.9601 0.0001 -0.0167 0.5341

5 1.9663 0.0001 -0.0169 0.3149

6 1.9700 0.0000 -0.0170 0.1898

7 1.9723 0.0000 -0.0171 0.1158

8 1.9737 0.0000 -0.0171 0.0712

9 1.9746 0.0000 -0.0171 0.0439

10 1.9751 0.0000 -0.0172 0.0272

11 1.9755 0.0000 -0.0172 0.0169

12 1.9757 0.0000 -0.0172 0.0105

13 1.9758 0.0000 -0.0172 0.0065

14 1.9759 0.0000 -0.0172 0.0040

15 1.9759 0.0000 -0.0172 0.0025

16 1.9760 0.0000 -0.0172 0.0016

**raiz =**

**1.9760**

(b) Pelo método da secante:

>> raiz = msec('deflex4',1.5,2.5,0.001,1000)

iter xr fun(xr) ea

1 1.9683 0.0001 40.0000

2 1.9759 0.0000 27.0132

3 1.9760 -0.0000 0.3864

4 1.9760 0.0000 0.0036

**raiz =**

**1.9760**

(c) Usando a função fzero do MATLAB:

>> raizc = fzero('deflex4',1)

**raizc =**

**1.9760**

**QUESTÃO 5:**

O gráfico a seguir corresponde a função para o cálculo da profundidade correspondente às condições descritas a seguir.

function [fx] = profund(h)

%Calculo da profundidade da agua no tanque, dados:

L = 10; %L: comprimento do tanque (m)

r = 1; %r: raio da secao transversal (m)

V = 12.4; %V: volume de agua (m³)

fx = V - L\*abs((0.5\*pi\*r^2 - r^2\*asin(h/r) - h.\*sqrt(r^2-h.^2)));

end



Embora, após a plotagem do gráfico, poderiam ter sido adotados valores que resultariam menor número de iterações e, consequentemente, menor tempo computacional de execução, adotou-se como o intervalo inicial as condições limites para o raio do semicírculo.

>> h = bissec('profund',0,1,1,100)

ii xi F(xi) xu F(xu) xr F(xr) ea

1 0.000000 -3.307963 0.500000 12.400000 0.500000 6.258152 100.000000

2 0.000000 -3.307963 0.250000 6.258152 0.250000 1.639454 100.000000

3 0.125000 -3.307963 0.250000 1.639454 0.125000 -0.814489 100.000000

4 0.125000 -0.814489 0.187500 1.639454 0.187500 0.419947 33.333333

5 0.156250 -0.814489 0.187500 0.419947 0.156250 -0.195726 20.000000

6 0.156250 -0.195726 0.171875 0.419947 0.171875 0.112536 9.090909

7 0.164063 -0.195726 0.171875 0.112536 0.164063 -0.041493 4.761905

8 0.164063 -0.041493 0.167969 0.112536 0.167969 0.035548 2.325581

9 0.166016 -0.041493 0.167969 0.035548 0.166016 -0.002966 1.176471

10 0.166016 -0.002966 0.166992 0.035548 0.166992 0.016292 0.584795

**h =**

**0.1670**

Logo a profundidade da água no tanque (m) é dada por:

>> profd = 1 - h

**profd =**

**0.8330**

**QUESTÃO 6:**

Para a determinação da quantidade mínima de material a ser utilizado na confecção do recipiente em função do raio r, deve-se obter o valor para o qual a derivada da função corresponde à área total do cilindro seja nula.

function [fx] = areatotal\_cil(r)

%Calculo da quantidade minima de material a ser utilizado correspondente a

%area total do recipiente (cilindro), dado:

V = 1000; %V: volume do recipiente (cm³)

fx = 2\*pi\*(r+0.25).^2 + (2\*pi\*r+0.25)\*V/(pi\*r.^2);

end

O gráfico e o código da derivada desta função está representado abaixo, onde pode-se observar que o valor em que esta condição é atendida está compreendida no intervalo entre r = 4 e r = 6cm.

function [dfx] = dareatotal\_cil(r)

%Derivada da area total do recipiente (cilindro), dado:

V = 1000; %V: volume do recipiente (cm³)

dfx = 4\*pi\*(r+0.25) + (2\*pi\*r+0.25)\*(-2)\*V/(pi\*r.^3) + 2\*V/r.^2;

end



>> raiz = bissec('dareatotal\_cil',4,6,0.04,10)

ii xi F(xi) xu F(xu) xr F(xr) ea

1 5.000000 -74.079721 6.000000 22.247432 5.000000 -15.299794 20.000000

2 5.000000 -15.299794 5.500000 22.247432 5.500000 5.184325 9.090909

3 5.250000 -15.299794 5.500000 5.184325 5.250000 -4.547192 4.761905

4 5.250000 -4.547192 5.375000 5.184325 5.375000 0.434318 2.325581

5 5.312500 -4.547192 5.375000 0.434318 5.312500 -2.026123 1.176471

6 5.343750 -2.026123 5.375000 0.434318 5.343750 -0.788503 0.584795

7 5.359375 -0.788503 5.375000 0.434318 5.359375 -0.175264 0.291545

8 5.359375 -0.175264 5.367188 0.434318 5.367188 0.129981 0.145560

9 5.363281 -0.175264 5.367188 0.129981 5.363281 -0.022527 0.072833

10 5.363281 -0.022527 5.365234 0.129981 5.365234 0.053756 0.036403

**raiz =**

**5.3652**