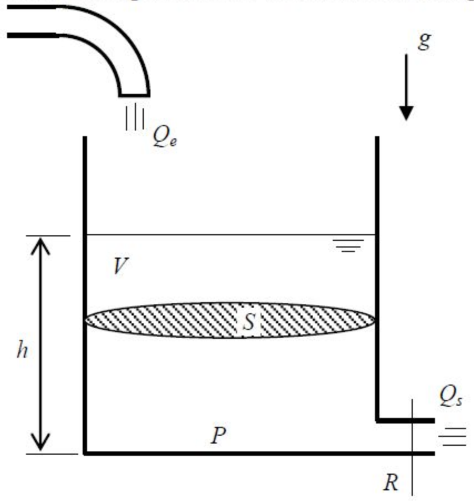
Bruno Nogueira Lucas
10772668

Lista B

Exercício:

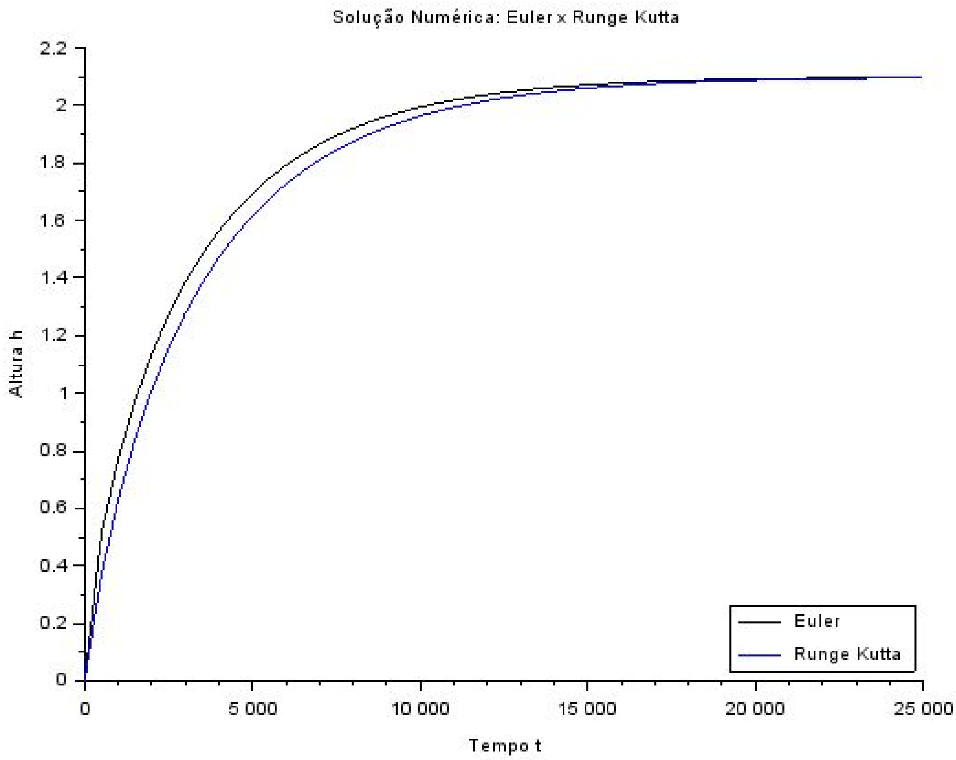
Implemente um programa no Scilab que resolva numericamente a equação diferencial que modela o sistema abaixo, tanto pelo método de Euler como Runge Kutta.



Reservatório com água
Parâmetros:
S = 10 m^2 - área da seção transversal (constante)
$R = 2\times10^8$ Pa/(m^3/s)^2 - parâmetro que relaciona vazão com queda de pressão (perda de carga)
ρ = 1000 kg/m^3 - massa específica da água
G = 10 m/s^2 - aceleração da gravidade na superfície da terra

Variáveis:
Q_e = 0,010247 m^3/s - vazão de entrada
h: nível do reservatório [m]
V: volume de água no reservatório [m^3]
P: pressão relativa à atmosférica, no fundo do reservatório [Pa]
Q_S: vazão de saída [m^3/s]

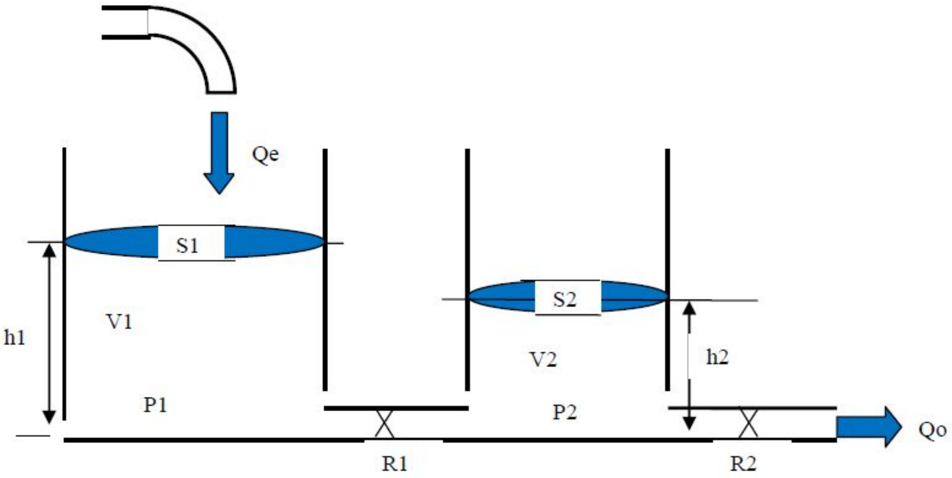
Admite-se que a água seja incompressível.

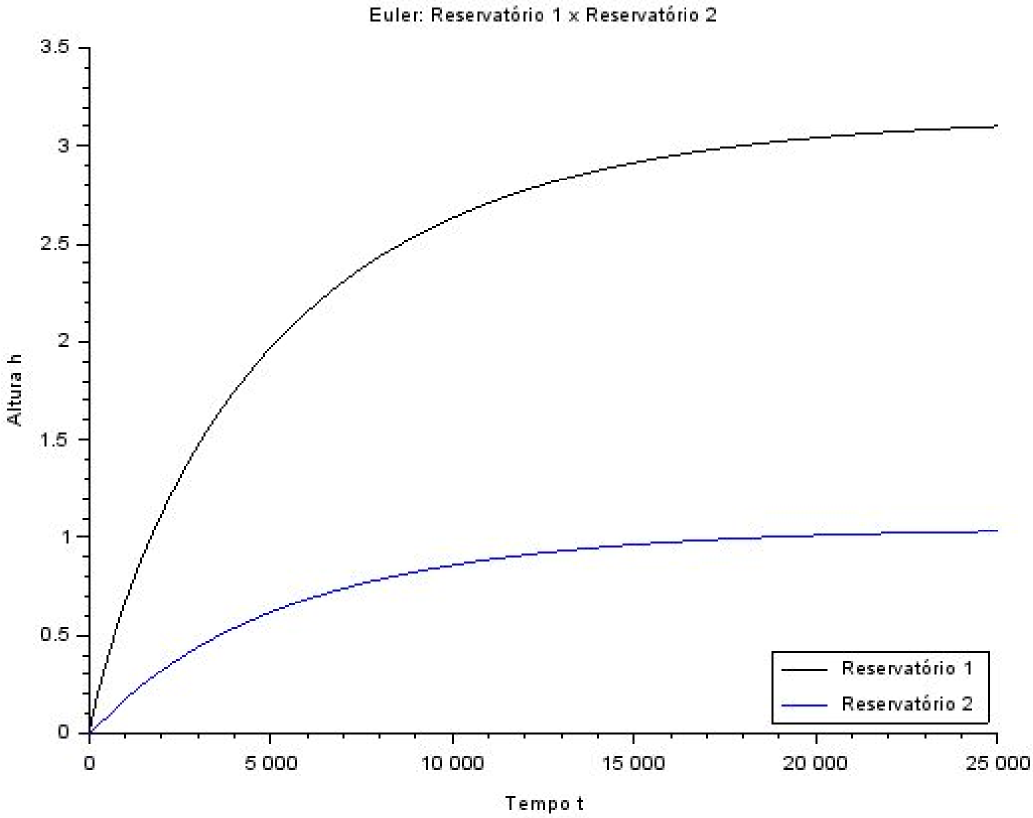


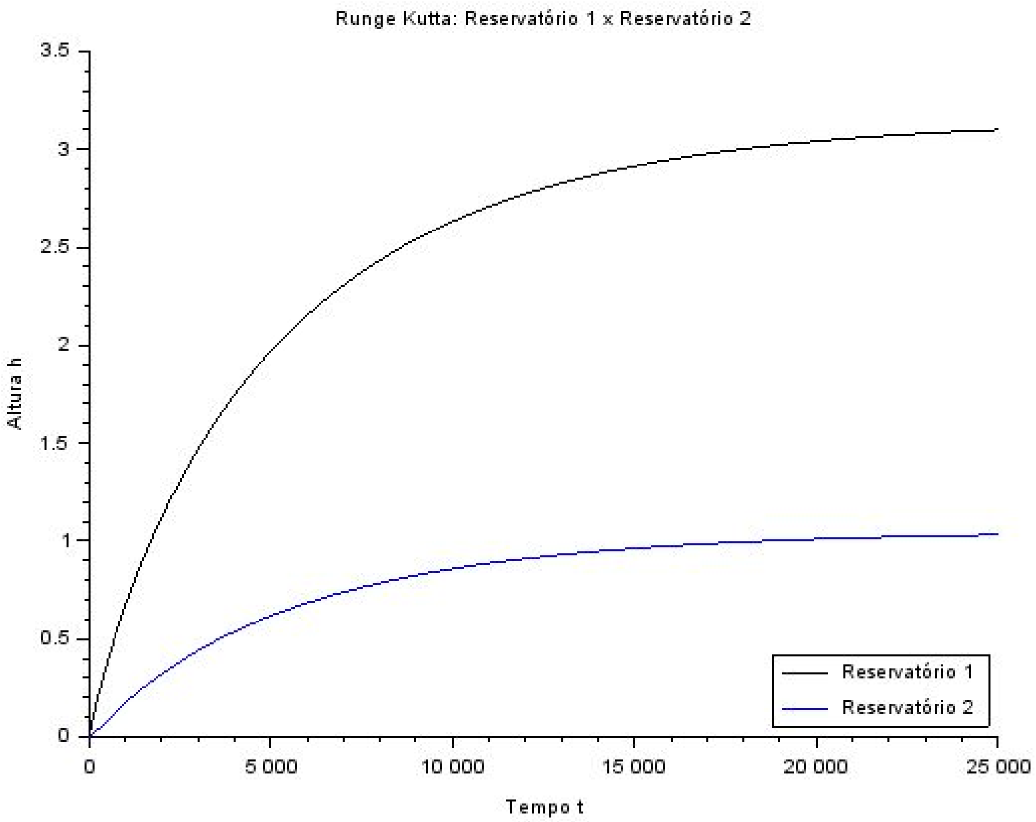

```
clear

// Definindo os parâmetros do sistema
```

```
S = 10;
R = 2*10^8;
ro = 1000;
G = 10;

// Definindo as variáveis do sistema
Qe = 0.010247;

// Equação Diferencial
function hdot=funcao(h)
   hdot = (-sqrt(ro*G*h/R)+Qe)/S;
endfunction

t(1) = 0;
tf = 25000;
h(1) = 0;
w(1) = 0;
x = 500;
n = round(tf/x);

// Aplicação dos Métodos

for i=1:n
   t(i+1) = t(i)+x;
   h(i+1) = h(i)+x*funcao(h(i));
   k1 = x*(-sqrt(ro*G*w(i)/R)+Qe)/S;
   k2 = x*(-sqrt(ro*G*(w(i)+k1/2)/R)+Qe)/S;
   k3 = x*(-sqrt(ro*G*(w(i)+k2/2)/R)+Qe)/S;
   k4 = x*(-sqrt(ro*G*(w(i)+k3)/R)+Qe)/S;
   w(i+1) = w(i)+((k1+2*k2+2*k3+k4)/6);
end

// Plotagem dos dados
plot2d([t,t],[h,w]);
legends(["Euler","Runge Kutta"],[1,2],4)
xtitle("Solução Numérica: Euler x Runge Kutta","Tempo t","Altura h")
```

Desenvolva um programa em Scilab que resolva numericamente o sistema de equações diferencias que modela o sistema com dois reservatórios, usando tanto Euler como Runge Kutta. Dica: raciocine com vetores.








```
clear

// Definindo os parâmetros do sistema
S1 = 10;
S2 = S1/2;
R1 = 2*10^8;
R2 = R1/2;
ro = 1000;
G = 10;

// Definindo as variáveis do sistema
Qe = 0.010247;

// Equação Diferencial
function h1dot=funcao1(h1, h2)
  h1dot = (-sqrt(ro*G*(h1-h2)/R1)+Qe)/S1;
endfunction

function h2dot=funcao2(h1, h2)
  h2dot = (-sqrt(ro*G*h2/R2)+sqrt(ro*G*(h1-h2)/R1))/S2;
endfunction

t(1) = 0;
tf = 25000;
h1(1) = 0;
h2(1) = 0;
w1(1) = 0;
```

```
w2(1) = 0;
x = 5
n = round(tf/x);

// Aplicação dos Métodos

for i=1:n
    t(i+1) = t(i)+x;

    h1(i+1) = h1(i)+x*funcao1(h1(i),h2(i));
    h2(i+1) = h2(i)+x*funcao2(h1(i),h2(i));

    k11 = x*funcao1(w1(i),w2(i));
    k21 = x*funcao1(w1(i)+k11/2,w2(i)+k11/2);
    k31 = x*funcao1(w1(i)+k21/2,w2(i)+k21/2);
    k41 = x*funcao1(w1(i)+k31,w2(i)+k31);
    k12 = x*funcao2(w1(i),w2(i));
    k22 = x*funcao2(w1(i)+k12/2,w2(i)+k12/2);
    k32 = x*funcao2(w1(i)+k22/2,w2(i)+k22/2);
    k42 = x*funcao2(w1(i)+k32,w2(i)+k32);
    w1(i+1) = w1(i)+((k11+2*k21+2*k31+k41)/6);
    w2(i+1) = w2(i)+((k12+2*k22+2*k32+k42)/6);
end

// Plotagem dos dados
plot2d([t,t],[h1,h2]);
legends(["Reservatório 1","Reservatório 2"],[1,2],4)
xtitle("Euler: Reservatório 1 x Reservatório 2","Tempo t","Altura h")
set("current_figure",1)
plot2d([t,t],[w1,w2]);
legends(["Reservatório 1","Reservatório 2"],[1,2],4)
xtitle("Runge Kutta: Reservatório 1 x Reservatório 2","Tempo t","Altura h")
```