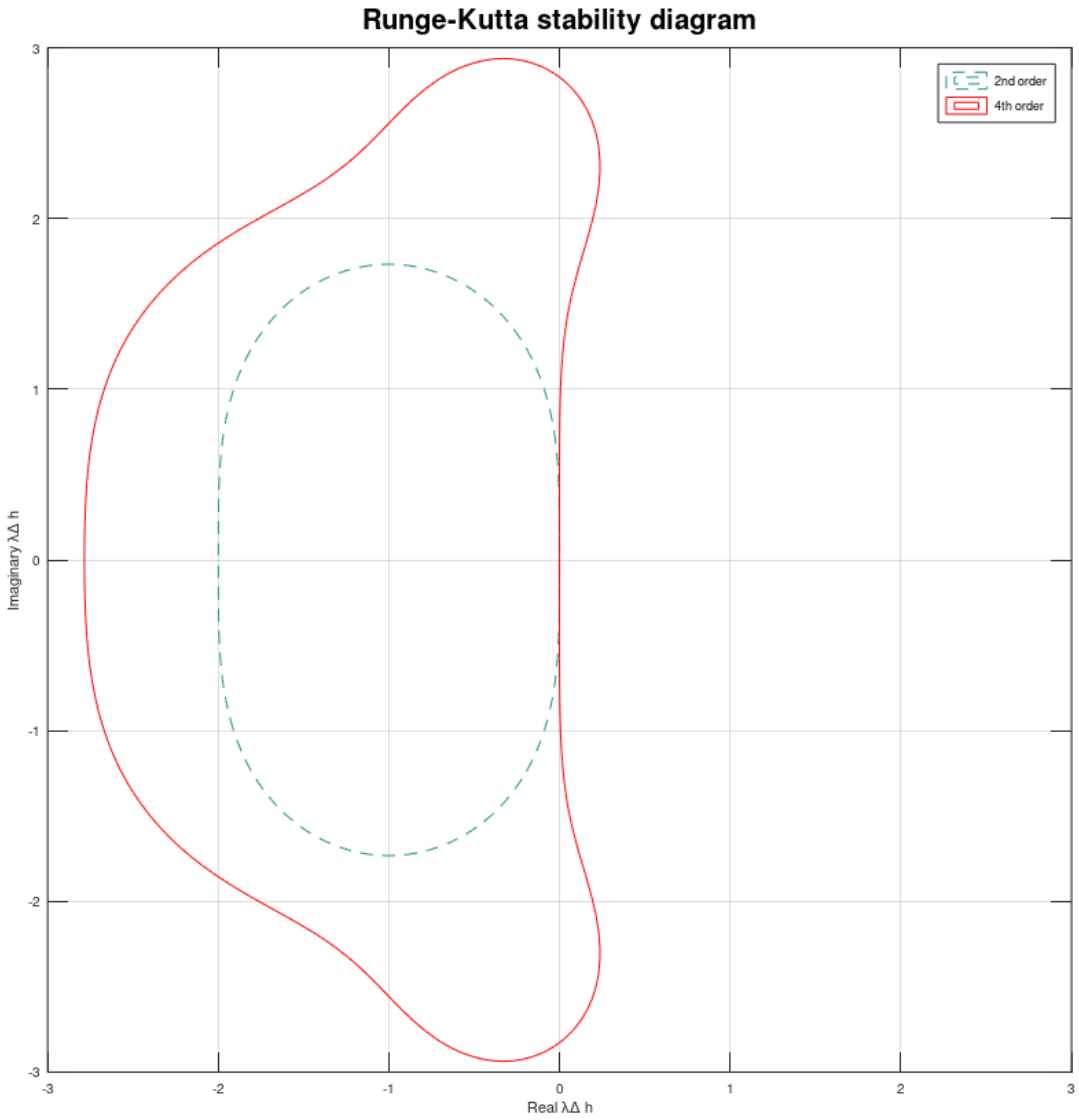
hw5 B09505021 Numerical Analysis

B09505021 工海二 張景華

A.

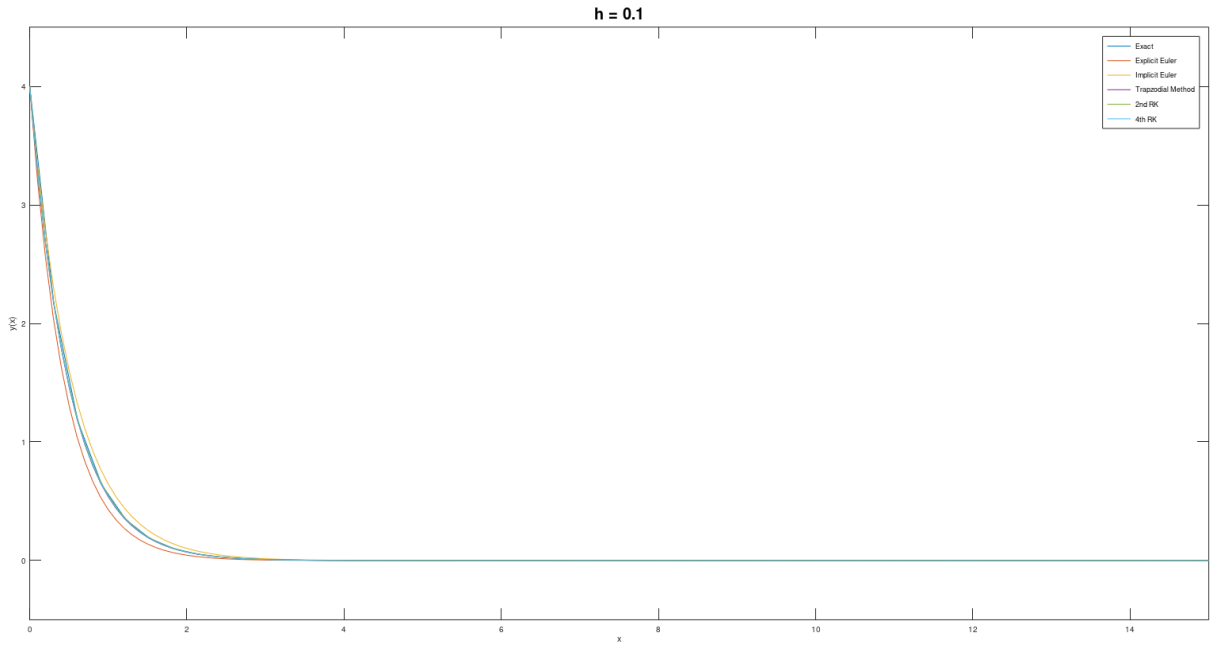
```
x = linspace(-3,3);
y = linspace(-3,3);
[x,y] = meshgrid(x,y);
z = x + i*y;
g1 = 1 + z + 0.5*z.^2;
g2= 1 + z + 0.5*z.^2 + 1/6*z.^3 + 1/24*z.^4;
g1mag  = abs(g1);
g2mag = abs(g2);
contour(x,y,g1mag,[1 1],'--');
hold on;
contour(x,y,g2mag,[1 1],'r-');
axis('square');
xlabel('Real \lambda\Delta h');
ylabel('Imaginary \lambda\Delta h');
legend( '2nd order','4th order');
title('Runge-Kutta stability diagram','FontSize',20);
grid on;

inter2 = @(x)(abs(1 +x +0.5*x.^2 )-1);
a=fsolve(inter2,0);
b=fsolve(inter2,-3);
inter4 = @(x)(abs(1+ x +0.5*x.^2 +1/6*x.^3 +1/24*x.^4 )-1);
c=fsolve(inter4,0);
d=fsolve(inter4,-3);
inter4 = @(y)(abs(1+ i*y +0.5*i*y.^2 +1/6*i*y.^3 +1/24*i*y.^4 )-1);
e=fsolve(inter4,0);
f=fsolve(inter4,-3);
g=fsolve(inter4,3);
```

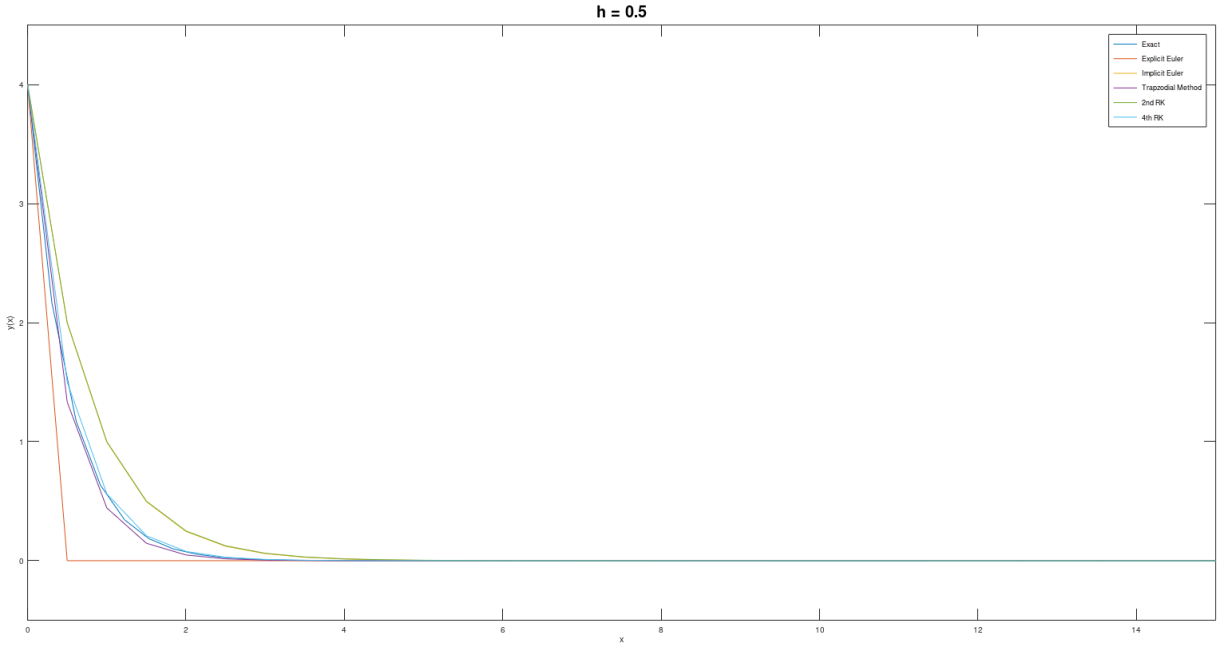


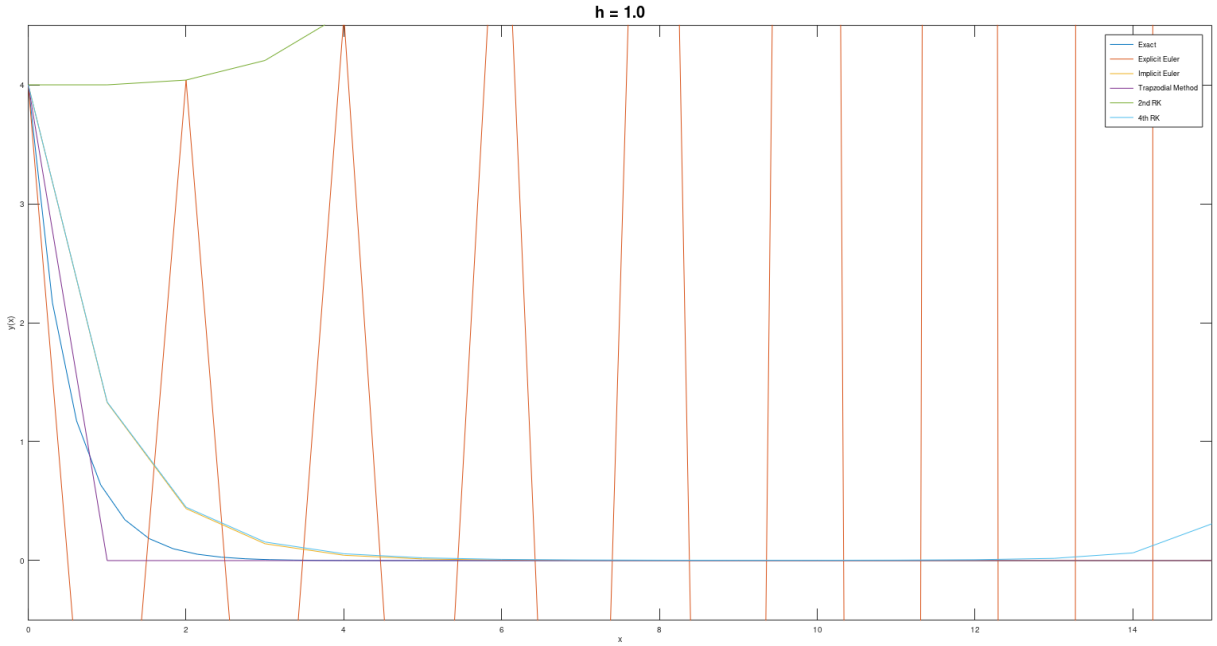
B.1

```
f = @(x) 4*exp(-0.01/3*x.^3-2*x);
x_true = linspace(0,15,50);
y_true = f(x_true);
lamda = @(x) (-2-0.01*x.^2)

h = 1.0;
x = (0:h:15);
n = length(x);
y1 = zeros(1,n);  #(Euler )
y2 = zeros(1,n);  #(Backward Euler )
y3 = zeros(1,n);  #(Trapezoidal Method )
y4 = zeros(1,n);  #(2nd Runge-Kutta )
y5 = zeros(1,n);  #(4th Runge-Kutta )
```

```
y1(1) = 4;
y2(1) = 4;
y3(1) = 4;
y4(1) = 4;
y5(1) = 4;

for i = 1:n-1;
  y1(i+1) = y1(i) .* (1+lamda(x(i)).*h);
  y2(i+1) = y2(i) ./ (1-lamda(x(i+1)).*h);
  y3(i+1) = y3(i) .* (1+lamda(x(i)).*h/2)./(1-lamda(x(i+1)).*h/2);
  y4(i+1) = y4(i) .* (1+lamda(x(i)).*h .+((lamda(x(i)).^2)*h^2)/2);
  y5(i+1) = y5(i) .* (1+lamda(x(i)).*h .+((lamda(x(i)).^2)*h^2)/2 .+((lamda(x(i)).^3)*h^3)/6 .+
((lamda(x(i)).^4)*h^4)/24);
endfor;

plot(x_true,y_true)
hold on
plot(x,y1)
plot(x,y2)
plot(x,y3)
plot(x,y4)
plot(x,y5)
axis([0 15 -0.5 4.5])
xlabel('x')
ylabel('y(x)')
legend('Exact','Explicit Euler','Implicit Euler','Trapzodial Method','2nd RK','4th RK')
title('h = 1.0','Fontsize',20)
```



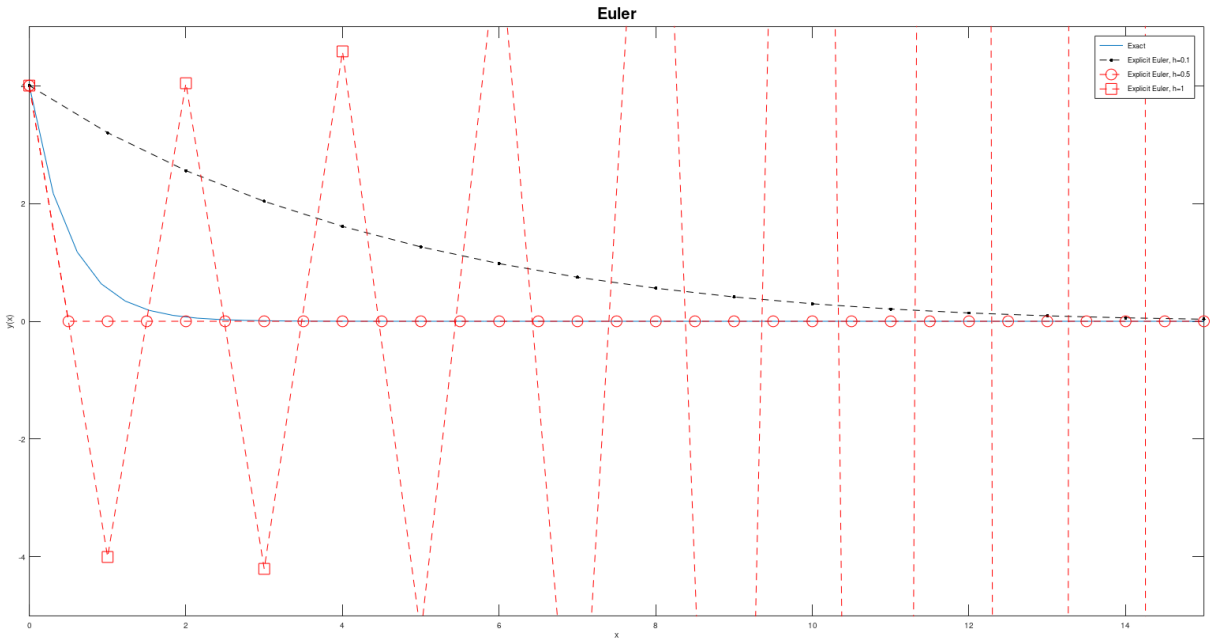


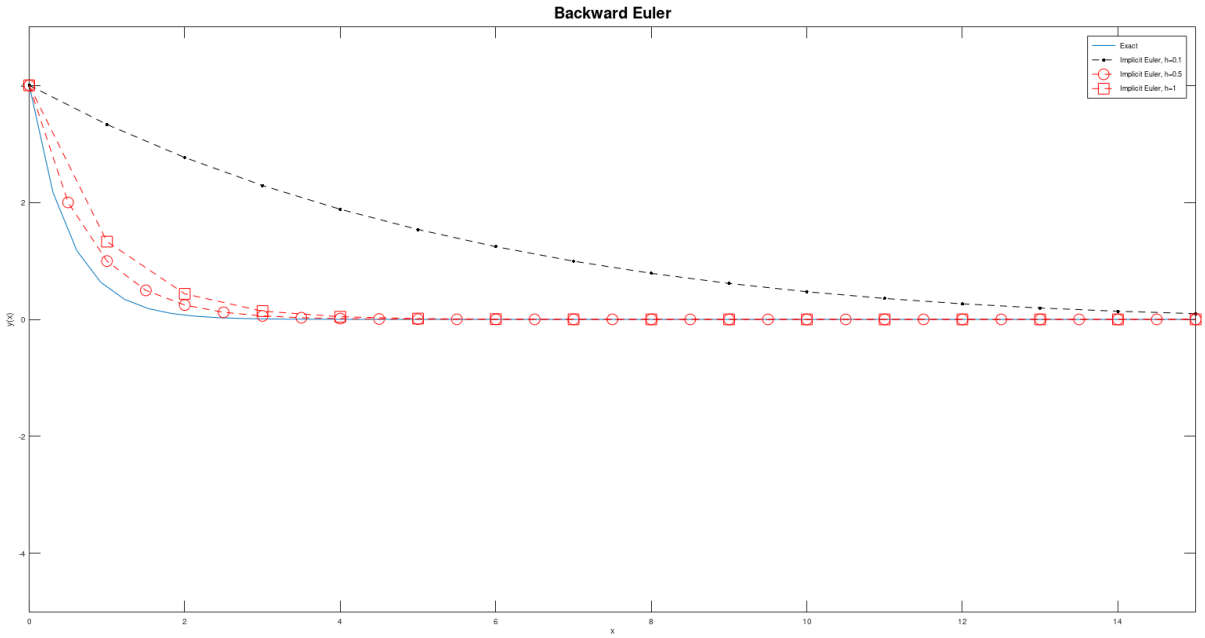


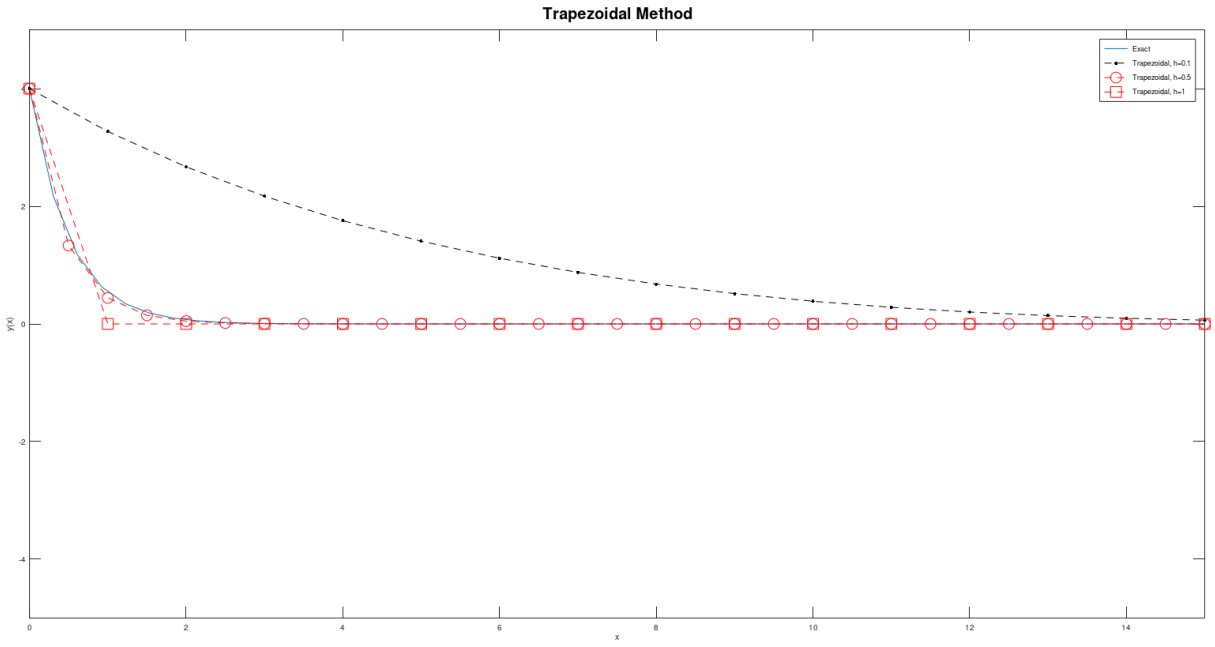
B.2

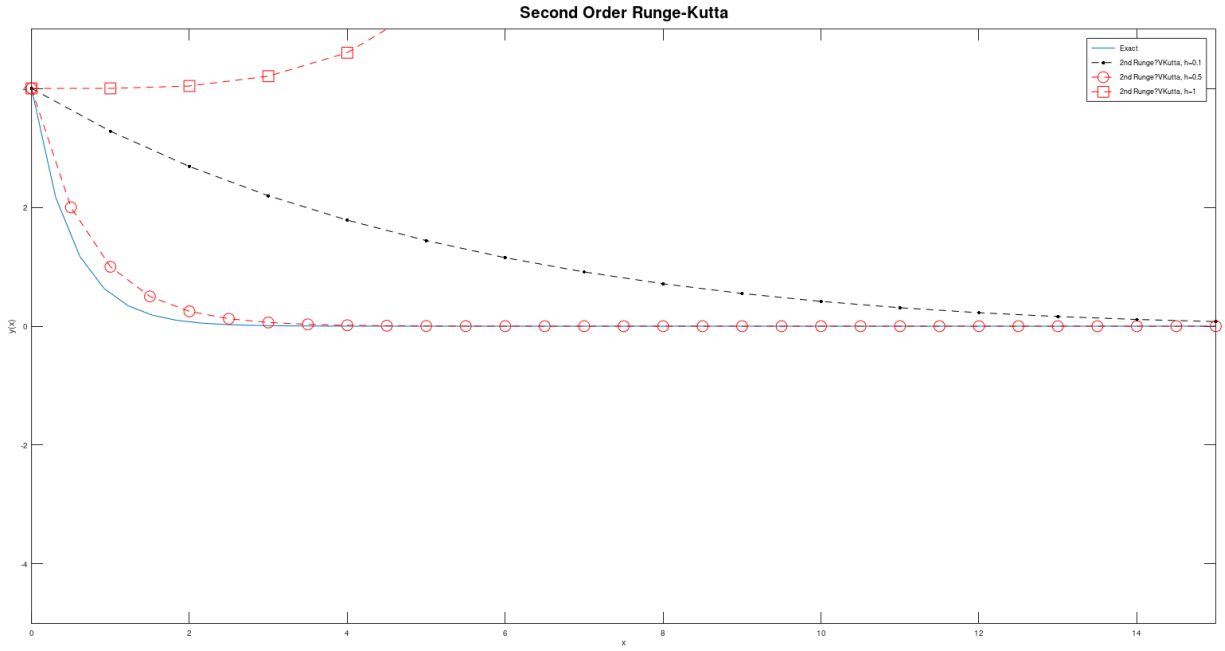
```
f = @(x) 4*exp(-0.01/3*x.^3-2*x);
x_true = linspace(0,15,50);
y_true = f(x_true);
lamda = @(x1) (-2-0.01*x1.^2)
lamda = @(x2) (-2-0.01*x2.^2)
lamda = @(x3) (-2-0.01*x3.^2)
```

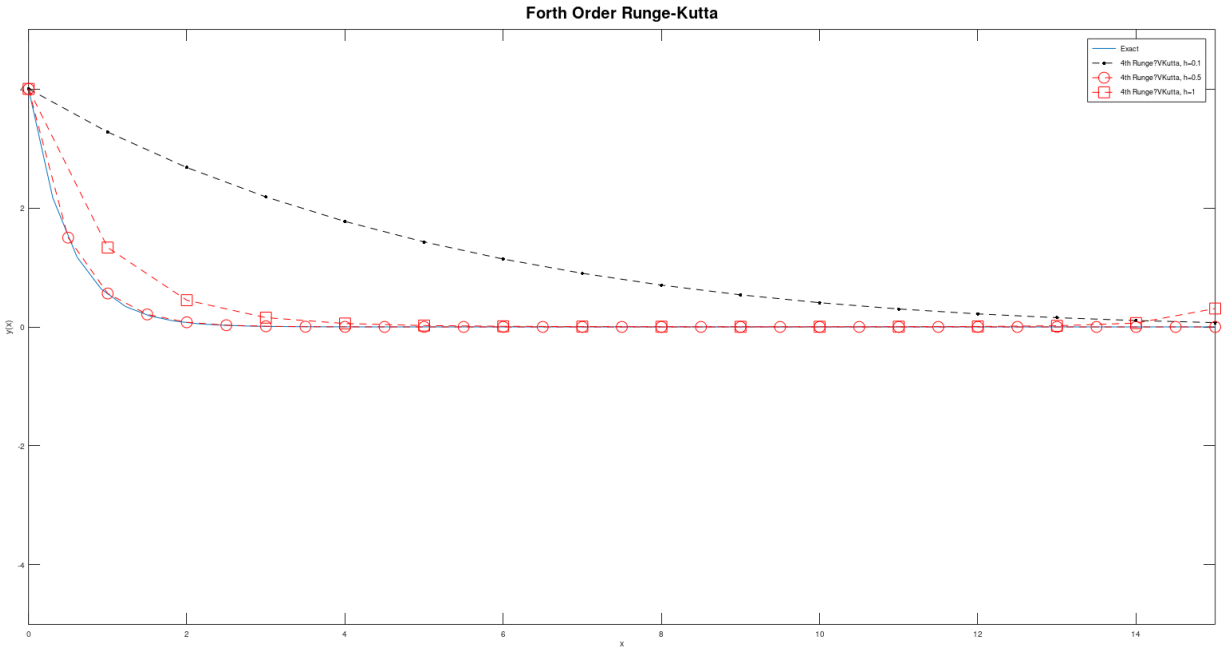
```
h1 = 0.1;
x1 = (0:h:15);
n1 = length(x1);
y1 = zeros(1,n1);
h2 = 0.5;
x2 = (0:h2:15);
n2 = length(x2);
y2 = zeros(1,n2);
h3 = 1;
x3 = (0:h3:15);
n3 = length(x3);
y3 = zeros(1,n3);
y1(1) = 4;
y2(1) = 4;
y3(1) = 4;
for i = 1:n1-1;
  y1(i+1) = y1(i) .* (1+lamda(x1(i)).*h1);
endfor;
for i = 1:n2-1;
  y2(i+1) = y2(i) .* (1+lamda(x2(i)).*h2);
endfor;
for i = 1:n3-1;
  y3(i+1) = y3(i) .* (1+lamda(x3(i)).*h3);
endfor;

plot(x_true,y_true)
hold on
plot(x1,y1,'k.--','MarkerSize',7)
plot(x2,y2,'ro--','MarkerSize',7)
plot(x3,y3,'rs--','MarkerSize',7)
axis([0 15 -5 5])
xlabel('x')
ylabel('y(x)')
legend('Exact','Explicit Euler, h=0.1','Explicit Euler, h=0.5','Explicit Euler, h=1','Location','NorthEast')
title('Euler','Fontsize',20)
```




Backward Euler
Exact
Implicit Euler, h=0.1
Implicit Euler, h=0.5
Implicit Euler, h=1
y(x)
x


Trapezoidal Method
Exact
Trapezoidal, h=0.1
Trapezoidal, h=0.5
Trapezoidal, h=1
y(x)
x


Second Order Runge-Kutta
Exact
2nd Runge?VKutta, h=0.1
2nd Runge?VKutta, h=0.5
2nd Runge?VKutta, h=1
y(x)
x


Forth Order Runge-Kutta
Exact
4th Runge?VKutta, h=0.1
4th Runge?VKutta, h=0.5
4th Runge?VKutta, h=1
y(x)
x