Boundary value problem

Example 1.

$$\frac{d^2y}{dx^2} + y = 0, y(0) = 0; y\left(\frac{\pi}{2}\right) = 1$$

Solution:

```
function eigen_bvp
  MAXIT = 100;
  TOL = 0.00000000001;
  slope = 1;
  k = 1;
  fp = fopen('data.txt', 'w');
  while k<=MAXIT
      fprintf('%f\n',slope);
      options = odeset('RelTol',1e-4,'AbsTol',[1e-6 1e-6
1e-6 1e-6]);
      [t,Y] = ode45(@ivp,[0 pi/2],[0 slope 0 1],options);
%RKF scheme
      N = length(Y(:,1));
      table=[t';Y(:,1)';Y(:,2)'];
      slope = slope - Y(N,1)/Y(N,3); %NR scheme
      if (Y(N,1)-1)<=TOL
          fprintf(fp,'%f\t%f\t%f\t%f\n',table);
          break;
      end
      k=k+1;
  end
  x=0:0.1:pi/2;
  y=sin(x); %exact solution
  figure(1)
  plot(t,Y(:,1),'-r')
  hold on
  plot(x,y,'o')
  fclose(fp);
% -------------------------------------------------------
----- %
function dy = ivp(t,y) %ivp's definition
dy = zeros(4,1);
dy(1) = y(2);
dy(2) = -y(1);
dy(3) = y(4);
dy(4) = -y(3);
end
end
```

Example 2.

$$\frac{d^2y}{dx^2} + 2y\frac{dy}{dx} = 0, y(0) = 1; y(1) = 0.5$$

Solution:

```
function shooting
  MAXIT = 100;
  TOL = 0.00000000001;
  slope = -1;
  k = 1;
  fp = fopen('data.txt', 'w');
  while k<=MAXIT
      fprintf('%6.10f\n',slope);
      options = odeset('RelTol',1e-4,'AbsTol',[1e-6 1e-6
1e-6 1e-6]);
      [t,Y] = ode45(@ivp,[0 1],[1 slope 0 1],options);
%RKF scheme
      N = length(Y(:,1));
      table=[t';Y(:,1)';Y(:,2)'];
      slope = slope - (Y(N,1)-0.5)/Y(N,3); %NR scheme
      if (Y(N,1)-0.5)<=TOL
          fprintf(fp,'%f\t%f\t%f\t%f\n',table);
          break;
      end
      k=k+1;
  end
  x=0:0.1:1;
  y=1./(1+x); %exact solution
  figure(1)
  plot(t,Y(:,1),'-r')
  hold on
  plot(x,y,'o')
  fclose(fp)
% -------------------------------------------------------
-----
function dy = ivp(t,y) %ivp's definition
dy = zeros(4,1);
dy(1) = y(2);
dy(2) = -2*y(1)*y(2);
dy(3) = y(4);
dy(4) = -2*y(2)*y(3) - 2*y(1)*y(4);
end
end
```

Example3:

$$\frac{d^2y}{dx^2} - 6y^2 = 0, y(0) = 1; y(0.5) = \frac{4}{9}$$

Solution:

```
function shooting
clf;clear;clc
close all
format long;
format compact;
MAXIT = 100;
TOL = 0.00000000001;
slope =-2;
k = 1;
fp = fopen('data.txt', 'w');
while k<=MAXIT
      fprintf('%6.10f\n',slope);
      options = odeset('RelTol',1e-6,'AbsTol',[1e-10 1e-
10 1e-    10 1e-10]);
      [t,Y] = ode45(@ivp,[0 0.5],[1 slope 0 1],options);
%RKF scheme
      N = length(Y(:,1));
      table=[t';Y(:,1)';Y(:,2)'];
      slope = slope - Y(N,1)/Y(N,3); %NR scheme
      if (Y(N,1)-(4/9))<=TOL
          fprintf(fp,'%f\t%f\t%f\t%f\n',table);
          break;
      end
      k=k+1;
  end
  figure(1)
  plot(t,Y(:,1),'-r')
  hold on
  fclose(fp);
% -------------------------------------------------------
-----
function dy = ivp(t,y) %ivp's definition
dy = zeros(4,1);
dy(1) = y(2);
dy(2) =6*y(1)*y(1);
dy(3) = y(4);
dy(4) =12*y(1)*y(3);
end
end
```

Falkaner-scan equation:

$$\frac{d^3y}{dx^3}+y\frac{d^2y}{dx^2}+\beta\left(1-\left(\frac{dy^2}{dx}\right)\right)=0, y(0)=\frac{dy}{dx}(0)=\frac{dy}{dx}(\infty)-1=0$$

Solution:

```
global beta

beta=0
a=fzero(@deriv,0.5)
x0=[0 0 a];
[t,Y]=ode45(@falknerskan,[0 5],x0);
plot(t,Y(:,2),'r')
hold on

beta=0.4
a=fzero(@deriv,0.5)
x0=[0 0 a];
[t,Y]=ode45(@falknerskan,[0 4.5],x0);
plot(t,Y(:,2),'y')

beta=1
a=fzero(@deriv,0.5)
x0=[0 0 a];
[t,Y]=ode45(@falknerskan,[0 3],x0);
plot(t,Y(:,2),'m')

beta=1.6
a=fzero(@deriv,0.5)
x0=[0 0 a];
[t,Y]=ode45(@falknerskan,[0 2.2],x0);
plot(t,Y(:,2),'g')

beta=-0.199
a=fzero(@deriv,0.5)
x0=[0 0 a];
[t,Y]=ode45(@falknerskan,[0 5],x0);
plot(t,Y(:,2),'b')


function f=deriv(a)

global beta
x0=[0 0 a];
```

```
if (beta==0 || beta==-0.199)
[t,Y]=ode45(@falknerskan,[0 5],x0);
end

if beta==0.4
[t,Y]=ode45(@falknerskan,[0 4.5],x0);
end

if beta==1
[t,Y]=ode45(@falknerskan,[0 3],x0);
end

if beta==1.6
[t,Y]=ode45(@falknerskan,[0 2.2],x0);
end

f=Y(end,2)-1;

function xdot=falknerskan(t,x)

global beta

xdot(1)=x(2);
xdot(2)=x(3);
xdot(3)=-x(1)*x(3)-beta*(1-x(2)^2);

xdot=xdot';
```

Initial value problem

Forward-Euler Method:

1. $$\frac{dy}{dx} = f(x,y) = x^2 + y^2, y(0) = 1, [0,1]$$

Solution:

```
clc
close all
f=@(x,y) x^2+y^2;
x(1)=0;
y(1)=1;
```

```
h=0.025;
fprintf('Grid Point\t y(computed)\t');
fprintf('--------------------\n')
for i=1:40
    y(i+1)=y(i)+h*f(x(i),y(i));
    x(i+1)=x(i)+h;
    fprintf('%.2f\t %.3f\n',x(i+1),y(i+1));
end
 hold on
plot(x,y,'-p')
axis([0 1 0 20])
xlabel('$x$','Interpreter','latex','fontsize',20)
ylabel('$y$','Interpreter','latex','fontsize',20)
title('Forward Euler Method')
```

2. $$\frac{dy}{dx} = f(x,y) = -\frac{1}{2}y\,,\; y(0) = 1, [0,5]$$

Solution:

```
clc
close all
f=@(x,y) -0.5*y;
x(1)=0;
y(1)=1;
h=0.25;
fprintf('Grid Point\t y(computed)\t');
fprintf('--------------------\n')
for i=1:40
    y(i+1)=y(i)+h*f(x(i),y(i));
    x(i+1)=x(i)+h;
    fprintf('%.2f\t %.3f\n',x(i+1),y(i+1));
end
t=0:0.25:10;
y1=1./exp(0.5.*t); %exact solution
figure(1)
plot(t,y1,'-p')
hold on
plot(x,y,'-r')
axis([0 8 0 1])
%legend({'$n=0.5$','$n=1$','$n=2$'},'Interpreter','latex'
,'fontsize',18)
xlabel('$x$','Interpreter','latex','fontsize',20)
ylabel('$y$','Interpreter','latex','fontsize',20)
title('Forward Euler Method')
```

Modified- Euler Method:

1. $$\frac{dy}{dx} = f(x,y) = x^2 + y^2, y(0) = 1, [0,1]$$

Solution:

```
clc
close all
f=@(x,y) x^2+y^2;
x(1)=0;
y(1)=1;
h=0.05;
fprintf('Grid Point\t y(computed)\t');
fprintf('--------------------\n')
for i=1:40
    y(i+1)=y(i)+h*f(x(i),y(i));
    x(i+1)=x(i)+h;
    y(i+1)=y(i)+0.5*h*(f(x(i+1),y(i+1))+f(x(i),y(i)));
    fprintf('%.2f\t %.3f\n',x(i+1),y(i+1));

end
plot(x,y,'-p')
axis([0 1 0 20])
hold on
xlabel('x','fontsize',20)
ylabel('y','fontsize',20)
title('Modified Euler method')
```

2. $$\frac{dy}{dx} = f(x,y) = -\frac{1}{2}y, y(0) = 1, [0,5]$$

Solution:

```
clc
close all
f=@(x,y) -0.5*y;
x(1)=0;
y(1)=1;
h=0.5;
fprintf('Grid Point\t y(computed)\t');
fprintf('--------------------\n')
for i=1:20
    y(i+1)=y(i)+h*f(x(i),y(i));
    x(i+1)=x(i)+h;
    y(i+1)=y(i)+0.5*h*(f(x(i+1),y(i+1))+f(x(i),y(i)));
```

```
    fprintf('%.2f\t %.3f\n',x(i+1),y(i+1));
end
t=0:0.25:10;
y1=1./exp(0.5.*t); %exact solution
figure(1)
plot(t,y1,'-p')
hold on
plot(x,y,'r')
axis([0 5 0 1])
%legend({'$n=0.5$','$n=1$','$n=2$'},'Interpreter','latex'
,'fontsize',18)
xlabel('x','fontsize',20)
ylabel('y','fontsize',20)
title('Modified Euler method')
```

Runge-Kutta Method:

1. $\frac{dy}{dx} = f(x,y) = x^2 + y^2, y(0) = 1, [0, 1]$

Solution:

```
clc
close all
f=@(x,y) x^2+y^2;
x(1)=0;
y(1)=1;
h=0.05;
fprintf('Grid Point\t y(computed)\t');
fprintf('-------------------\n')
for i=1:40
    K1=h*f(x(i),y(i));
    K2=h*f(x(i)+0.5*h,y(i)+0.5*K1);
    K3=h*f(x(i)+0.5*h,y(i)+0.5*K2);
    K4=h*f(x(i)+h,y(i)+K3);
    y(i+1)=y(i)+(K1+2*K2+2*K3+K4)/6;
    x(i+1)=x(i)+h;
    fprintf('%.2f\t %.3f\n',x(i+1),y(i+1));
end
hold on
plot(x,y,'-p')
axis([0 1 0 20])
xlabel('x','fontsize',20)
ylabel('y','fontsize',20)
title('Fourth order R-K method')
```

2. $$\frac{dy}{dx} = f(x,y) = -\frac{1}{2}y\,,\, y(0) = 1, [0,5]$$

Solution:

```
clc
close all
f=@(x,y) -0.5*y;
x(1)=0;
y(1)=1;
h=0.5;
fprintf('Grid Point\t y(computed)\t');
fprintf('--------------------\n')
for i=1:20
    K1=h*f(x(i),y(i));
    K2=h*f(x(i)+0.5*h,y(i)+0.5*K1);
    K3=h*f(x(i)+0.5*h,y(i)+0.5*K2);
    K4=h*f(x(i)+h,y(i)+K3);
    y(i+1)=y(i)+(K1+2*K2+2*K3+K4)/6;
    x(i+1)=x(i)+h;
    fprintf('%.2f\t %.3f\n',x(i+1),y(i+1));
end
t=0:0.25:10;
y1=1./exp(0.5.*t); %exact solution
figure(1)
plot(t,y1,'-p')
hold on
plot(x,y,'r')
axis([0 5 0 1])
xlabel('x','fontsize',20)
ylabel('y','fontsize',20)
title('Fourth order R-K method')
```