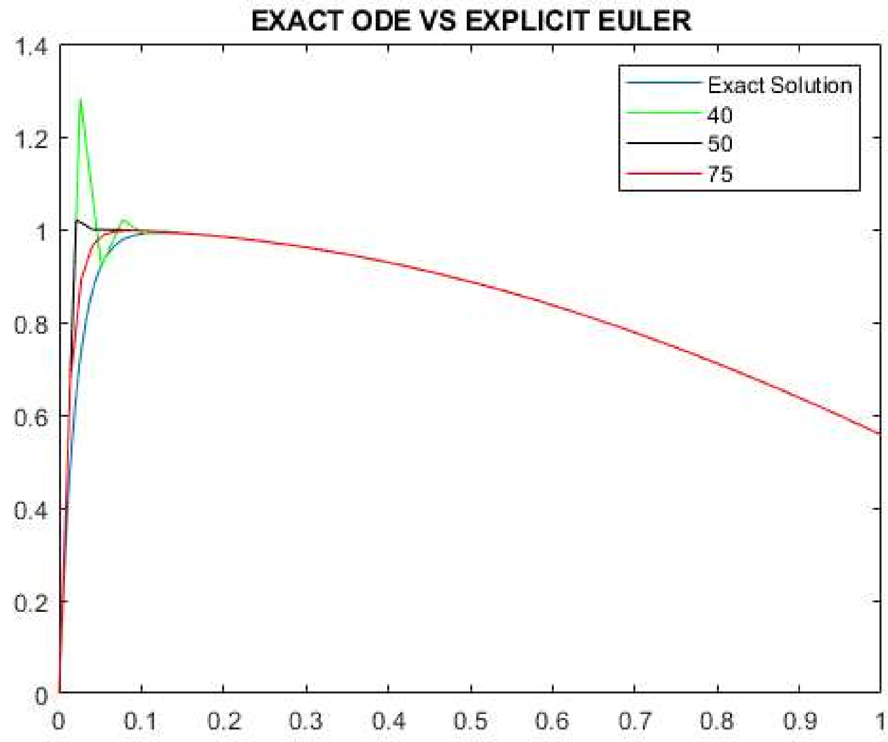
CFD Homework 2 - BRYAN ACOSTA

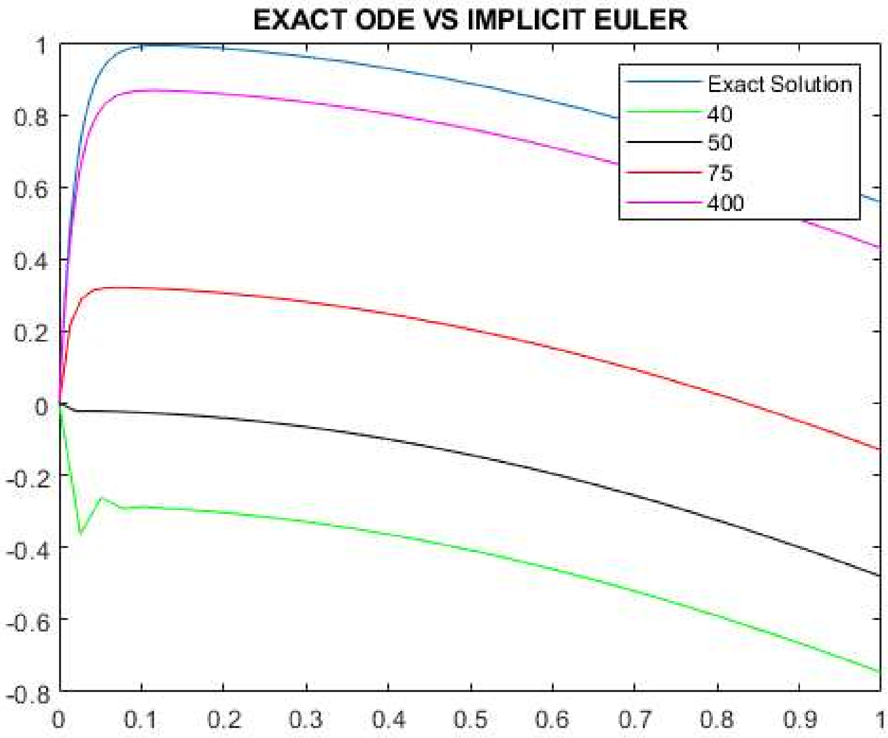
Contents

Solving ODES

```
clear
clc
close all
```

```
clear
clc
close all

x1 = linspace(0,1,30);
x2 = linspace(0,1,40);
x3 = linspace(0,1,50);
x4 = linspace(0,1,75);
x5 = linspace(0,1,150);
x6 = linspace(0,1,400);


diffequation=@(x,y) -50*(y - cos(x));

% EXACT ODE VS EXPLICIT EULER
figure(1)
plot(x6,exact_solution(diffequation,x6))
hold on
title('EXACT ODE VS EXPLICIT EULER')
plot(x2,Explicit_Euler(diffequation,x2),'g')
hold on
plot(x3,Explicit_Euler(diffequation,x3),'k')
hold on
plot(x4,Explicit_Euler(diffequation,x4),'r')
legend('Exact Solution','40','50','75')

% EXACT ODE VS IMPLICIT EULER
figure(2)
plot(x6,exact_solution(diffequation,x6))
hold on
title('EXACT ODE VS IMPLICIT EULER')
plot(x2,Implicit_Euler(diffequation,x2),'g')
hold on
plot(x3,Implicit_Euler(diffequation,x3),'k')
hold on
plot(x4,Implicit_Euler(diffequation,x4),'r')
hold on
plot(x6,Implicit_Euler(diffequation,x6),'m')
legend('Exact Solution','40','50','75','400')
```

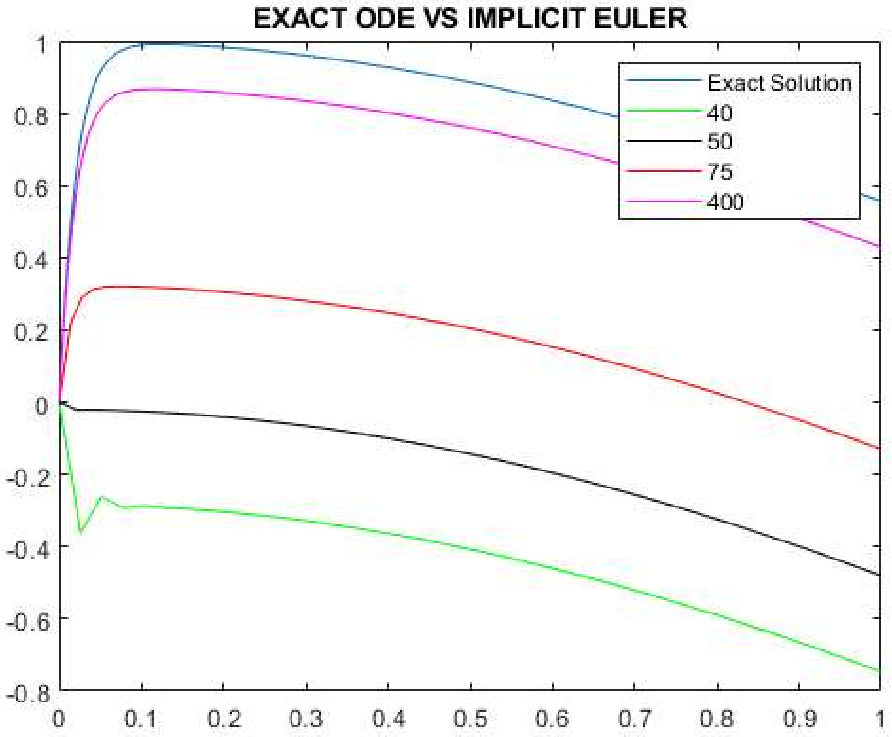


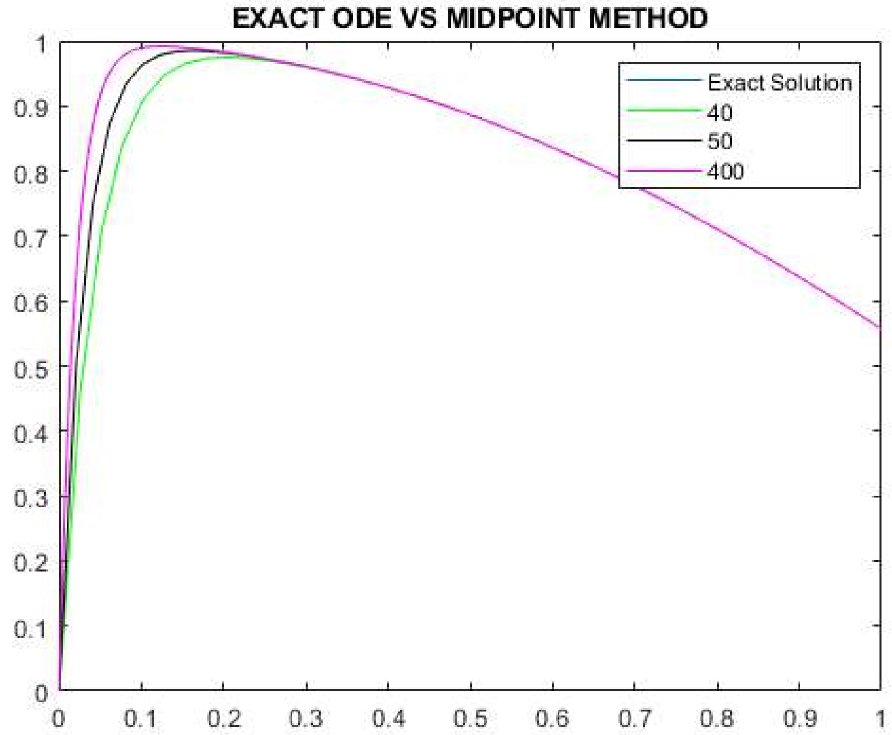


EXACT ODE VS MIDPOINT METHOD

```
figure(3)
plot(x6,exact_solution(diffequation,x6))
hold on
title('EXACT ODE VS MIDPOINT METHOD')
plot(x2,Midpoint(diffequation,x2),'g')
hold on
```

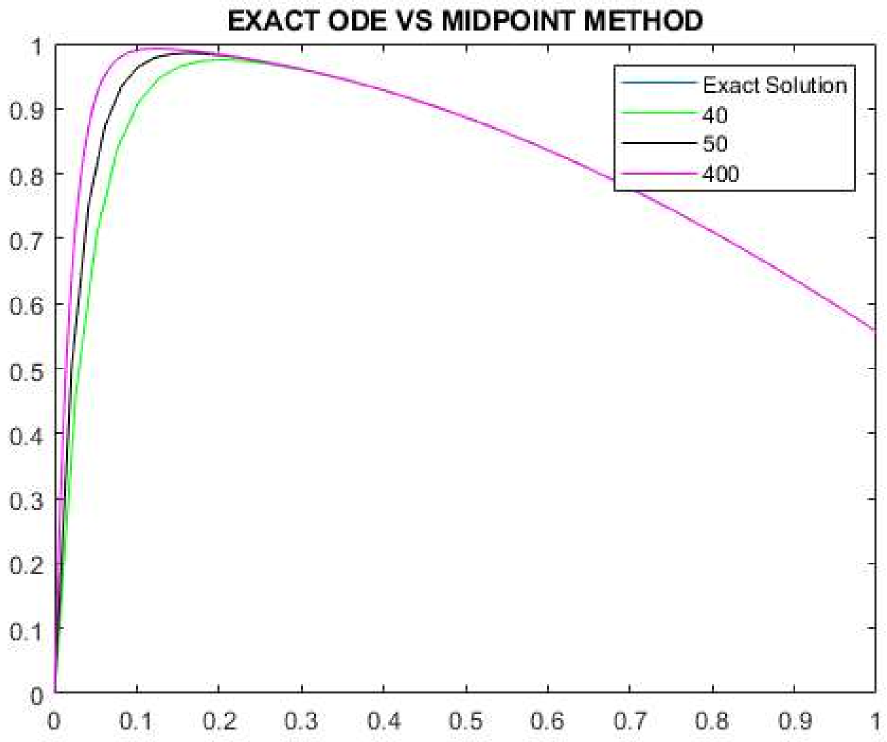
```
plot(x3,Midpoint(diffequation,x3),'k')
hold on
plot(x6,Midpoint(diffequation,x6),'m')
legend('Exact Solution','40','50','400')
```

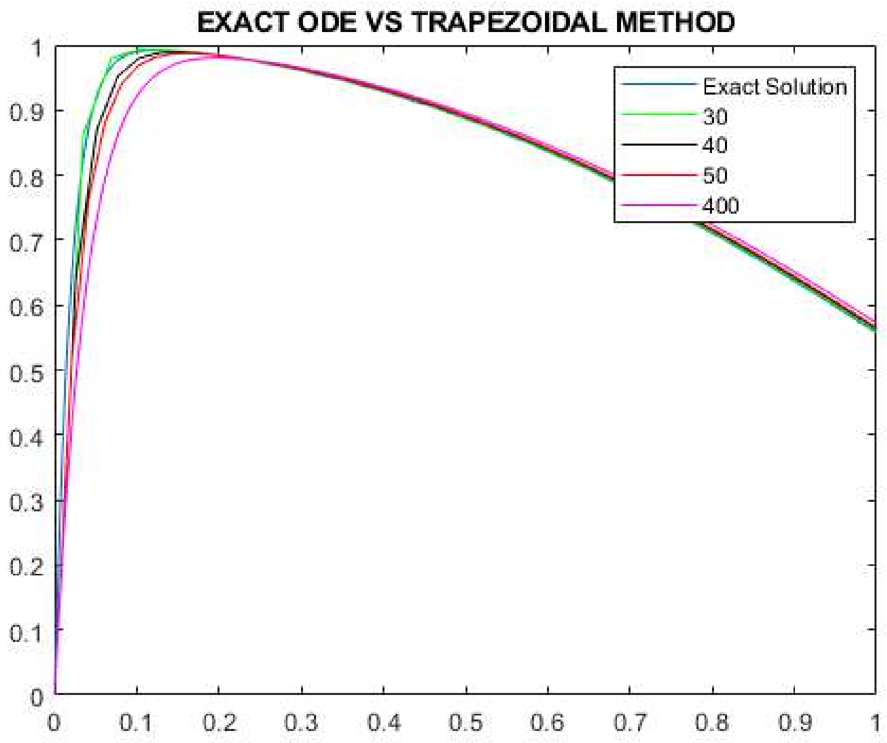




EXACT ODE VS TRAPEZOIDAL METHOD

```
figure(4)
plot(x6,exact_solution(diffequation,x6))
hold on
title('EXACT ODE VS TRAPEZOIDAL METHOD')
plot(x1,0.5.*trapezoidal(diffequation,x1),'g')
hold on
plot(x2,0.5.*trapezoidal(diffequation,x2),'k')
hold on
plot(x3,0.5.*trapezoidal(diffequation,x3),'r')
hold on
plot(x6,0.5.*trapezoidal(diffequation,x6),'m')
legend('Exact Solution','30','40','50','400')
```

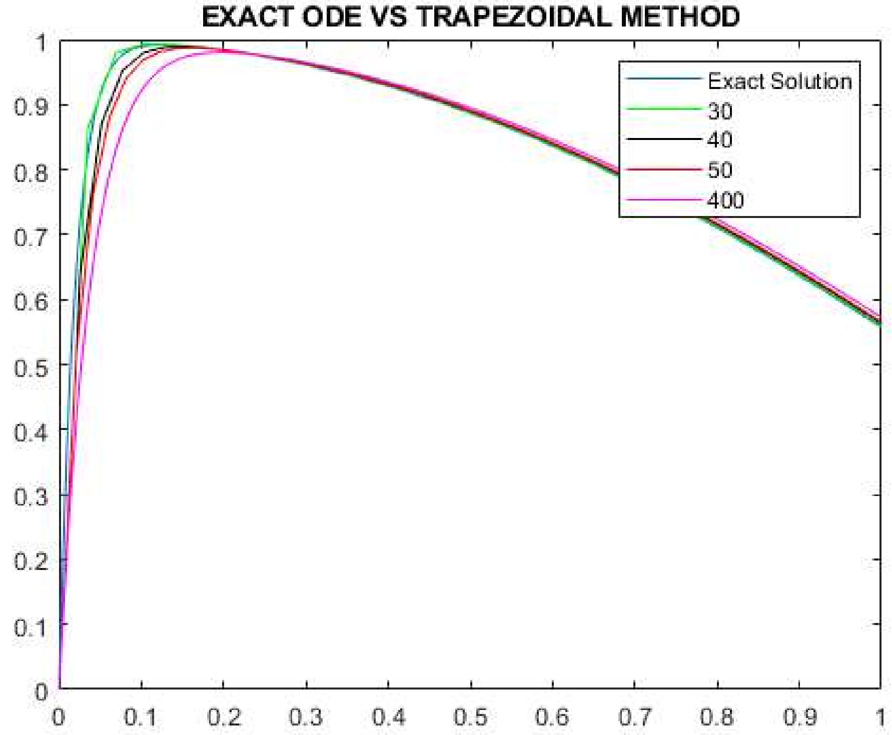


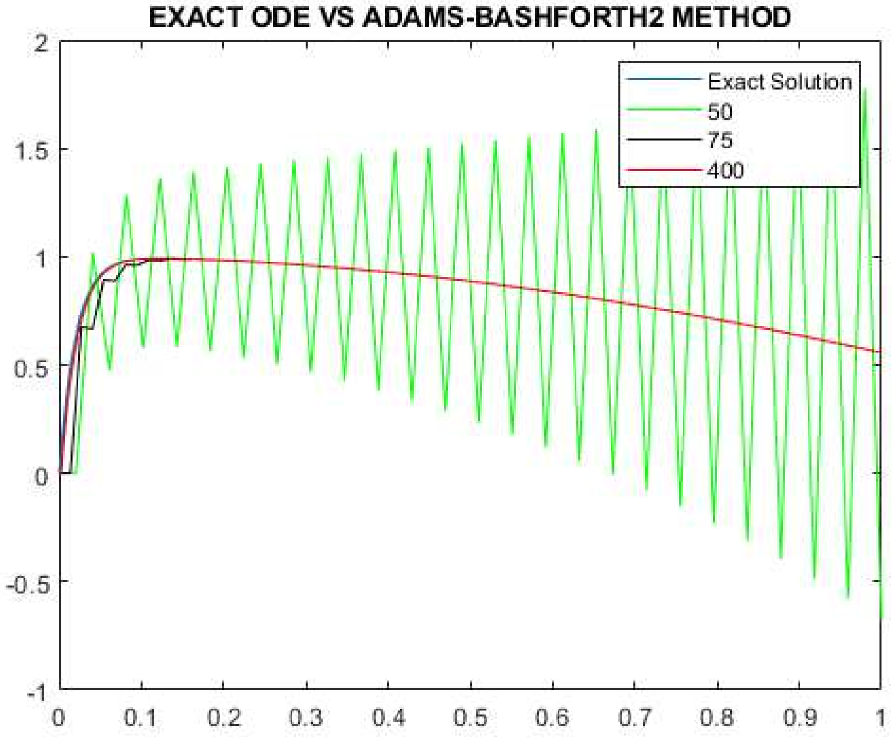


EXACT ODE VS ADAMS-BASHFORTH2 METHOD

```
figure(5)
plot(x6,exact_solution(diffequation,x6))
hold on
title('EXACT ODE VS ADAMS-BASHFORTH2 METHOD')
plot(x3,AdamsB2(diffequation,x3),'g')
hold on
```

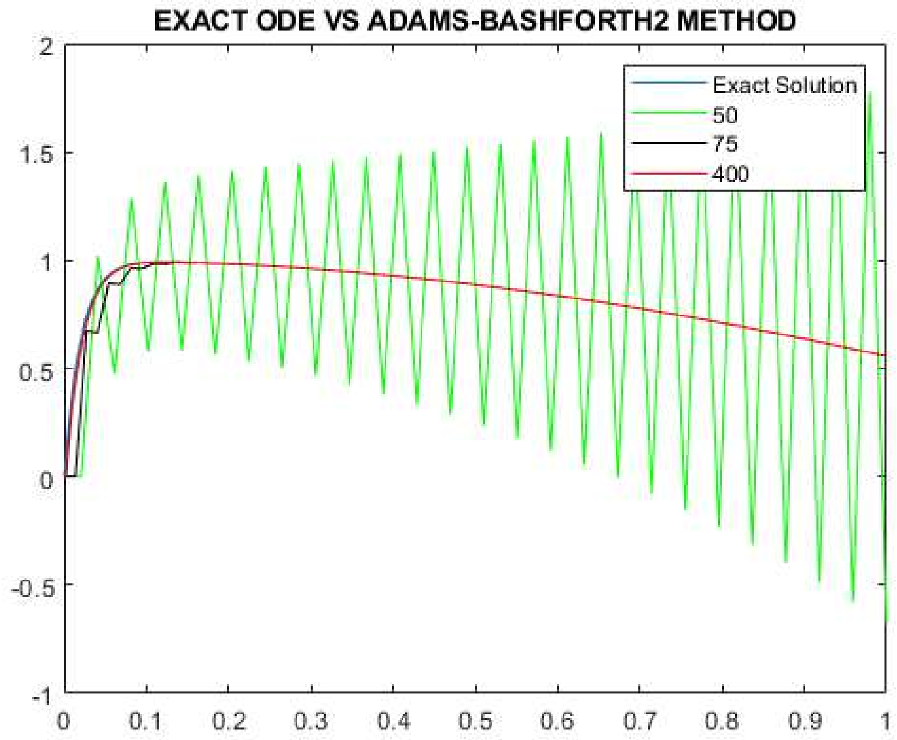
```
plot(x4,AdamsB2(diffequation,x4),'k')
hold on
plot(x6,AdamsB2(diffequation,x6),'r')
legend('Exact Solution','50','75','400')
```





EXACT ODE VS RUNGE-KUTTA 2 METHOD

```
figure(6)
plot(x6,exact_solution(diffequation,x6))
hold on
title('EXACT ODE VS RUNGE-KUTTA 2 METHOD')
plot(x1,RK2(diffequation,x1),'g')
hold on
plot(x2,RK2(diffequation,x2),'k')
hold on
plot(x6,RK2(diffequation,x6),'r')
legend('Exact Solution','30','40','400')

% EXACT ODE VS RUNGE-KUTTA 4 METHOD
figure(7)
plot(x6,exact_solution(diffequation,x6))
hold on
title('EXACT ODE VS RUNGE-KUTTA 4 METHOD')
plot(x1,RK4(diffequation,x1),'g')
hold on
plot(x2,RK4(diffequation,x2),'k')
hold on
plot(x3,RK4(diffequation,x3),'r')
hold on
plot(x6,RK4(diffequation,x6),'m')
legend('Exact Solution','30','40','50','400')
```



```
function [y] = exact_solution(diffeq, xspan)
    [~, y] = ode45(diffeq,xspan, 0);

end
function [output] = Explicit_Euler(diffeq, xspan)
    jump = xspan(2);
    output = xspan.*0;
    for i = 1: (length(xspan)-1)
```

```
           output(i+1) = output(i) + jump*diffeq(xspan(i), output(i));
    end
end
function [output] =Implicit_Euler(diffeq, xspan)
    jump = xspan(2);
    misc = xspan.*0;
    output = xspan.*0;
    for i = 1: (length(xspan)-1)
           misc(i+1) = misc(i) + diffeq(xspan(i), misc(i))*jump;
           output(i+1) = output(i)+ jump* diffeq(xspan(i+1), misc(i+1));
    end
end
function [output] =Midpoint(diffeq, xspan)
    jump = xspan(2);
    halfjump = jump/2;
    output = xspan.*0;
    for i = 1: (length(xspan)-1)
           output(i+1) = output(i) + jump*diffeq(xspan(i)+halfjump,output(i)+halfjump*diffeq(xspan(i),output(i)));
    end
end
function [output] =trapezoidal(diffeq, xspan)
    jump = xspan(2);
    halfjump = jump/2;
    output = xspan.*0;
    for i = 1: (length(xspan)-1)
           output(i+1) = output(i)+ halfjump*(diffeq(xspan(i), output(i))+ diffeq(xspan(i+1), output(i+1)));
    end
end
function [output] =AdamsB2(diffeq, xspan)
    jump = xspan(2);
    output = xspan.*0;
    for i = 1: (length(xspan)-2)
           output(i+2) = output(i+1) + 0.5*jump*(3*diffeq(xspan(i+1), output(i+1))-diffeq(xspan(i),output(i)));
    end
end
function [output] = RK2(diffeq, xspan)
    jump = xspan(2);
    output = xspan.*0;
    for i = 1: (length(xspan)-1)
           input1 = xspan(i)+ 0.5*jump;
           input2 = output(i)+(0.5*jump*diffeq(xspan(i),output(i)));
           output(i+1) = output(i) + jump*diffeq(input1,input2);
    end
end
function [output] = RK4(diffeq, xspan)
    jump = xspan(2);
    halfjump = jump/2;
    output = xspan.*0;
    for i = 1: (length(xspan)-1)
           input1 = diffeq(xspan(i),output(i));
           input2 = diffeq(xspan(i)+ halfjump,output(i)+halfjump*input1);
           input3 = diffeq(xspan(i)+ halfjump,output(i)+halfjump*input2);
           input4 = diffeq(xspan(i)+ jump,output(i)+jump*input3);
           output(i+1) = output(i) + (1/6)*jump*(input1+input2+input3+input4);
    end
end
```

Published with MATLAB® R2021a