```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% This code is applicable to problem 1 of Homework 4
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

clear all
close all

% Main routine (driver) for orbital ODE problem
% Specify name of user supplied function M-file with rhs of ode
odefun = 'orbitODE';
% Specify the method to be used
% Options are ExplicitEuler, RK1, RK4, HW2
method = 'RK1';
% Specify if you want automatic time steps, and if so, the tolerance
autostep = true;
TOL = 1.0*(10^-4);
global steps;
steps = 0;
% If automatic time steps are not to be used, specify the number of
 steps
NSTEP=5*(10^5);
% Specify the initial conditions
% Specify initial and final times
t0 = 0; tfinal = 17.1;
TSPAN = [t0,tfinal];
% Specify column vector of initial values
U0 = [0.994;0.0;0.0;-2.00158510637908252240537862224];
mu = 0.012277471;

% Build the Butcher array based on the method selected
if strcmp(method, 'RK4')
    A = [0 0 0 0; 0.5 0 0 0; 0 0.5 0 0; 0 0 1 0];
    b = [1/6;1/3;1/3;1/6];
    c = [0;0.5;0.5;1];
elseif (strcmp(method,'ExplicitEuler') || strcmp(method,'RK1'))
    A = 0;
    b = 1;
    c = 0;
elseif strcmp(method,'HW2')
    A = [0 0; 1 0];
    b = [0.5;0.5];
    c = [0;1];
end
% Series of if statements that will return the order of the RK method
 used
if round(sum(b),4) == 1.0000
    qorder = 1;
    if round(b.'*c,4) == 0.5000
        qorder = 2;
        if round(b.'*(c.^2),4) == round(1/3,4) && ...
           round(sum(b'.*sum(A'.*c)),4) == round(1/6,4)
            qorder = 3;
            if round(b.'*(c.^3),4) == 1/4 && ...
               round(sum(b'.*c'.*sum(A'.*c)),4) == 1/8 && ...
               round(sum(b'.*sum(A'.*c.^2)),4) == round(1/12,4) && ...
               round(sum(A.*b)*sum(A'.*c)',4) == round(1/24,4)
                qorder = 4;
            end
        end
    end
end

% Call the solver, based on if you want automatic time step or not
if autostep == false
    [t,U] = eulerw17d(odefun,TSPAN,U0,NSTEP,method,A,b,c,mu);
else
    [t,U] = RKw17sc(odefun,TSPAN,U0,TOL,A,b,c,qorder,mu);
end

% plot numerical solution;
figure;
hold off;
plot(U(1,:),U(2,:),'b');
xlabel('u_1')
ylabel('u_2')
titlestr = sprintf(['Method = ' method ...
    ', Tolerance = %2.0e, Steps = %i'],TOL,steps);
title(titlestr)
```

Published with MATLAB® R2016b