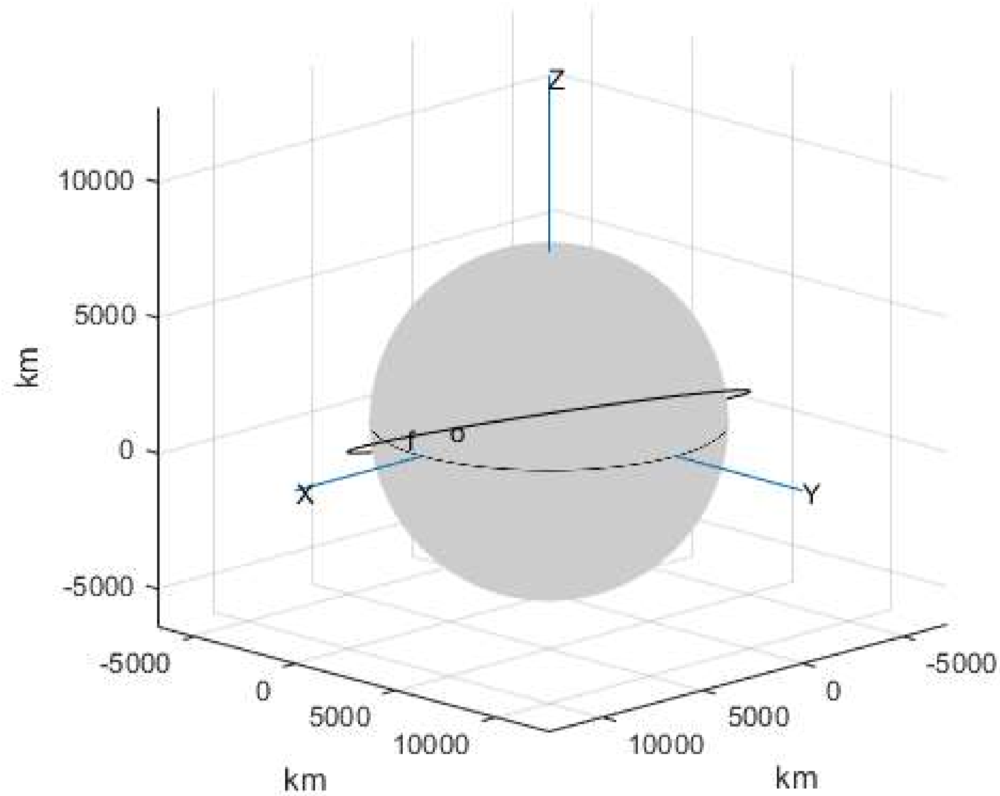
```
function orbit

%{
This function computes the orbit of a spacecraft by using rkf45 to
numerically integrate Equation 2.22.
It also plots the orbit and computes the times at which the maximum
and minimum radii occur and the speeds at those times.
hours - converts hours to seconds
G - universal gravitational constant (km^3/kg/s^2)
m1 - planet mass (kg)
m2 - spacecraft mass (kg)
mu - gravitational parameter (km^3/s^2)
R - planet radius (km)
r0 - initial position vector (km)
v0 - initial velocity vector (km/s)
t0,tf - initial and final times (s)
y0 - column vector containing r0 and v0
t - column vector of the times at which the solution is found
y - a matrix whose columns are:
columns 1, 2 and 3:
The solution for the x, y and z components of the
position vector r at the times in t
columns 4, 5 and 6:
The solution for the x, y and z components of the
velocity vector v at the times in t
r - magnitude of the position vector at the times in t
imax - component of r with the largest value
rmax - largest value of r
imin - component of r with the smallest value
rmin - smallest value of r
v_at_rmax - speed where r = rmax
v_at_rmin - speed where r = rmin
User M-function required: rkf45
User subfunctions required: rates, output
%}
% --------------------------------------------------------------------
clc; close all; clear all
hours = 3600;
G = 6.6742e-20;

%...Input data:
% Earth:
m1 = 5.974e24 %  7.3483e+22 %moon %5.974e24 - earth;
R = 6378.137%1737;
m2 = 1000;
r0 = [6837.432552 1868.795099 1455.480629];
v0 = [-2.294079 6.758849 2.049468];
t0 = 0;
tf = 200*60;%200*hours;
%...End input data

%...Numerical integration:
mu = G*(m1 + m2);
y0 = [r0 v0]';
[t,y] = rkf45(@rates, [t0 tf], y0);
```

```
%...Output the results:
output
return
% ~~~~~~~~~~~~~~~~~~~~~~~~
function dydt = rates(t,f)
% ~~~~~~~~~~~~~~~~~~~~~~~~
%{
This function calculates the acceleration vector using Equation 2.22
t - time
f - column vector containing the position vector and the
velocity vector at time t
x, y, z - components of the position vector r
r - the magnitude of the the position vector
vx, vy, vz - components of the velocity vector v
ax, ay, az - components of the acceleration vector a
dydt - column vector containing the velocity and acceleration
components
%}
% ------------------------
x = f(1);
y = f(2);
z = f(3);
vx = f(4);
vy = f(5);
vz = f(6);
r = norm([x y z]);
ax = -mu*x/r^3;
ay = -mu*y/r^3;
az = -mu*z/r^3;
dydt = [vx vy vz ax ay az]';
end %rates
% ~~~~~~~~~~~~~
function output
% ~~~~~~~~~~~~~
%{
This function computes the maximum and minimum radii, the times they
occur and and the speed at those times. It prints those results to
the command window and plots the orbit.
r - magnitude of the position vector at the times in t
imax - the component of r with the largest value
rmax - the largest value of r

imin - the component of r with the smallest value
rmin - the smallest value of r
v_at_rmax - the speed where r = rmax
v_at_rmin - the speed where r = rmin
User subfunction required: light_gray
%}
% -------------
for i = 1:length(t)
r(i) = norm([y(i,1) y(i,2) y(i,3)]);
end
[rmax imax] = max(r);
[rmin imin] = min(r);
v_at_rmax = norm([y(imax,4) y(imax,5) y(imax,6)]);
v_at_rmin = norm([y(imin,4) y(imin,5) y(imin,6)]);
%...Output to the command window:
```

```
fprintf('\n\n--------------------------------------------------------\n')
fprintf('\n Earth Orbit\n')
fprintf(' %s\n', datestr(now))
fprintf('\n The initial position is [%g, %g, %g] (km).',...
r0(1), r0(2), r0(3))
fprintf('\n Magnitude = %g km\n', norm(r0))
fprintf('\n The initial velocity is [%g, %g, %g] (km/s).',...
v0(1), v0(2), v0(3))
fprintf('\n Magnitude = %g km/s\n', norm(v0))
fprintf('\n Initial time = %g h.\n Final time = %g h.\n',0,tf/hours)
fprintf('\n The minimum altitude is %g km at time = %g h.',...
rmin-R, t(imin)/hours)
fprintf('\n The speed at that point is %g km/s.\n', v_at_rmin)
fprintf('\n The maximum altitude is %g km at time = %g h.',...
rmax-R, t(imax)/hours)
fprintf('\n The speed at that point is %g km/s\n', v_at_rmax)
fprintf('\n--------------------------------------------------------\n\n')
%...Plot the results:
% Draw the planet
[xx, yy, zz] = sphere(100);
surf(R*xx, R*yy, R*zz)
colormap(light_gray)
caxis([-R/100 R/100])
shading interp
% Draw and label the X, Y and Z axes
line([0 2*R], [0 0], [0 0]); text(2*R, 0, 0, 'X')
line( [0 0], [0 2*R], [0 0]); text( 0, 2*R, 0, 'Y')
line( [0 0], [0 0], [0 2*R]); text( 0, 0, 2*R, 'Z')
% Plot the orbit, draw a radial to the starting point
% and label the starting point (o) and the final point (f)
hold on
plot3( y(:,1), y(:,2), y(:,3),'k')
line([0 r0(1)], [0 r0(2)], [0 r0(3)])
text( y(1,1), y(1,2), y(1,3), 'o')
text( y(end,1), y(end,2), y(end,3), 'f')
% Select a view direction (a vector directed outward from the origin)
view([1,1,.4])
% Specify some properties of the graph
grid on
axis equal
xlabel('km')
ylabel('km')
zlabel('km')
% ~~~~~~~~~~~~~~~~~~~~~~~
function map = light_gray
% ~~~~~~~~~~~~~~~~~~~~~~~
%{
This function creates a color map for displaying the planet as light
gray with a black equator.
r - fraction of red
g - fraction of green
b - fraction of blue
%}
% -----------------------
r = 0.8; g = r; b = r;
map = [r g b
0 0 0
```

```
r g b];
end %light_gray
end %output
end %orbit
% ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
```

```
m1 =

   5.9740e+24


R =

   6.3781e+03



--------------------------------------------------------

 Earth Orbit
 28-Jan-2017 17:30:51

 The initial position is [6837.43, 1868.8, 1455.48] (km).
 Magnitude = 7236.11 km

 The initial velocity is [-2.29408, 6.75885, 2.04947] (km/s).
 Magnitude = 7.42598 km/s

 Initial time = 0 h.
 Final time = 3.33333 h.

 The minimum altitude is 852.503 km at time = 0.280533 h.
 The speed at that point is 7.43159 km/s.

 The maximum altitude is 875.06 km at time = 1.12901 h.
 The speed at that point is 7.40848 km/s

--------------------------------------------------------
```



Published with MATLAB® R2016a