Homework 1 Results

Using the equations in the Class 1 notes, compute the Keplerian elements and orbit period for the four position/velocity state vectors.

Given: $\mu = 3.986004418 \times 10^{14}\ m^3/s^2$

Vector 1

```
r:  ( -464836.97860600, -6191644.71680500, -2961635.48103900) m
rd: (    7322.77235464,     406.01896116,   -1910.89281450) m/sec
```

Keplerian Elements for Vector 1

```
       Keplerian Elements for Vector 1
       a:     6819999.99903084 m
       e:           0.01000000
       inc:        30.00000000 deg
       raan:       30.00000000 deg
       wp:         29.99999941 deg
       nu:        209.43319063 deg
       TP:       5605.15391192 sec
       rp:    6751799.99904516 m
       ra:    6888199.99901653 m
```

Vector 2

```
r:  (  572461.71122800, -1015437.19439600, 7707337.87130200) m
rd: (   -6195.26294500,   -3575.88965000,      -5.42328300) m/sec
```

Keplerian Elements for Vector 2

```
       a:     7800000.00120126 m
       e:           0.00100000
       inc:        98.60000000 deg
       raan:       30.00000000 deg
       wp:         40.00000696 deg
       nu:         50.08784582 deg
       TP:       6855.71704370 sec
       rp:    7792200.00046198 m
       ra:    7807800.00194054 m
```

Vector 3

```
r:  ( -5142754.61711500, 16130814.76756600, 20434322.22979000) m
rd: (   -2924.28712800,   -2303.32626400,    1084.79883400) m/sec
```

Keplerian Elements for Vector 3

```
       a:     26560000.00601707 m
       e:           0.00100000
       inc:        55.00000000 deg
       raan:       50.00000000 deg
       wp:         40.00000535 deg
       nu:         30.05735251 deg
       TP:      43077.75745550 sec
       rp:    26533440.00049993 m
       ra:    26586560.01153421 m
```

Vector 4

```
r:  ( -21100299.89402400, 36462486.12050000,   69117.55512600) m
rd: (   -2664.26812500,   -1539.99665900,       1.83444200) m/sec
```

Keplerian Elements for Vector 4

```
        a:     42164171.68690219 m
        e:           0.00100000
        inc:         0.10000000 deg
        raan:       49.99999572 deg
        wp:         40.00000208 deg
        nu:         30.05736006 deg
        TP:      86164.09682317 sec
        rp:    42122007.51785447 m
        ra:    42206335.85594991 m
```

Main Function

```
%%MATLAB for Homework 1

% Output file
fid = fopen('c:\temp\Homework1_2025.txt','wt');

%%Given and known values

%Assign value of mu (GM, Standard Gravitational parameter of earth)
mu=3.986004418e14;

fprintf(fid,'mu:      %16.8f\n', mu);
% First vector

%Input Given Vector Values
%Position and velocity vectors
r = [-464836.978606; -6191644.716805; -2961635.481039];
rd = [7322.77235464; 406.01896116; -1910.89281450];

%Dump vector information
fprintf(fid, 'Vector 1\n');
fprintf(fid,'r:  ( %16.8f, %16.8f, %16.8f) m\n', r(1), r(2), r(3));
fprintf(fid,'rd: ( %16.8f, %16.8f, %16.8f) m/sec\n\n', rd(1), rd(2), rd(3));

% Compute the Keplerian elements
[a, e, inc, raan, wp, nu] = KeplerianElements(mu, r, rd);

WriteKeplerianElements(fid, 1, a, e, inc, raan, wp, nu);

% Compute and output the orbit period
TP = Period(mu, a);
fprintf(fid,'TP:    %16.8f sec\n', TP);

% Compute and output the perigee and apogee radii
rp = a * (1 - e);
ra = a * (1 + e);
fprintf(fid,'rp:    %16.8f m\n', rp);
fprintf(fid,'ra:    %16.8f m\n', ra);

% Second vector
r = [572461.711228; -1015437.194396; 7707337.871302];
rd = [-6195.262945; -3575.889650; -5.423283];

%Dump vector information
fprintf(fid, '\n\nVector 2\n');
fprintf(fid,'r:  ( %16.8f, %16.8f, %16.8f) m\n', r(1), r(2), r(3));
fprintf(fid,'rd: ( %16.8f, %16.8f, %16.8f) m/sec\n\n', rd(1), rd(2), rd(3));

% Compute the Keplerian elements
[a, e, inc, raan, wp, nu] = KeplerianElements(mu, r, rd);

WriteKeplerianElements(fid, 2, a, e, inc, raan, wp, nu);

% Compute and output the orbit period
TP = Period(mu, a);
fprintf(fid,'TP:    %16.8f sec\n', TP);

% Compute and output the perigee and apogee radii
rp = a * (1 - e);
ra = a * (1 + e);
fprintf(fid,'rp:    %16.8f m\n', rp);
fprintf(fid,'ra:    %16.8f m\n', ra);
% Third vector

r = [-5142754.617115; 16130814.767566; 20434322.229790];
rd = [-2924.287128; -2303.326264; 1084.798834];

%Dump vector information
fprintf(fid, '\n\nVector 3\n');
```

```
fprintf(fid,'r:  ( %16.8f, %16.8f, %16.8f) m\n', r(1), r(2), r(3));
fprintf(fid,'rd: ( %16.8f, %16.8f, %16.8f) m/sec\n\n', rd(1), rd(2), rd(3));

% Compute the Keplerian elements
[a, e, inc, raan, wp, nu] = KeplerianElements(mu, r, rd);

WriteKeplerianElements(fid, 3, a, e, inc, raan, wp, nu);

% Compute and output the orbit period
TP = Period(mu, a);
fprintf(fid,'TP:    %16.8f sec\n', TP);

% Compute and output the perigee and apogee radii
rp = a * (1 - e);
ra = a * (1 + e);
fprintf(fid,'rp:    %16.8f m\n', rp);
fprintf(fid,'ra:    %16.8f m\n', ra);
% Fourth vector
r = [-21100299.894024; 36462486.120500; 69117.555126];
rd = [-2664.268125; -1539.996659; 1.834442];

%Dump vector information
fprintf(fid, '\n\nVector 4\n');
fprintf(fid,'r:  ( %16.8f, %16.8f, %16.8f) m\n', r(1), r(2), r(3));
fprintf(fid,'rd: ( %16.8f, %16.8f, %16.8f) m/sec\n\n', rd(1), rd(2), rd(3));

% Compute the Keplerian elements
[a, e, inc, raan, wp, nu] = KeplerianElements(mu, r, rd);

WriteKeplerianElements(fid, 4, a, e, inc, raan, wp, nu);

% Compute and output the orbit period
TP = Period(mu, a);
fprintf(fid,'TP:    %16.8f sec\n', TP);

% Compute and output the perigee and apogee radii
rp = a * (1 - e);
ra = a * (1 + e);
fprintf(fid,'rp:    %16.8f m\n', rp);
fprintf(fid,'ra:    %16.8f m\n', ra);
% Close the file
fclose(fid);

%% End of script
```

Keplerian Element Computation

```
function [a, e, inc, raan, wp, nu] = KeplerianElements(mu, r, v)

% Compute the Keplerian Elements as described in Class 1 Notes
% mu        Gravitational parameter (units consistent with r and v)
% r         Position vector in units consistent with mu
% v         Velocity vector in units consistent with mu

% Define the Cartesian Z axis
    Zhat = [0; 0; 1];

%     Compute the angular momentum
    h = cross(r, v);
%     Get the inclination
    inc = acos(dot(h, Zhat)/(norm(h) * norm(Zhat)));

%     Compute the node vector
    N = cross(Zhat, h);

%     Compute right ascension of ascending node
    raan = atan2(N(2), N(1));

%     Ensure RAAN is in range [0, 2pi]
    if (raan < 0)
        raan = raan + 2*pi;
    end

%     Get the B vector
    B = cross(v, h) - mu * (r/norm(r));

%     Argument of perigee computations
    cosWp = dot(B, N)/(norm(B) * norm(N));

%     The arguments in the cross product are unit vectors of N and B
    NxB = cross((N/norm(N)), (B/norm(B)));

%     Dot product with unit vector of h yields a cosine that's
%     either +1 or -1 since NxB is parallel or anti-parallel to h
    sinWp = dot((h/norm(h)), NxB);

%     Compute the argument of perigee in radians
    wp = atan2(sinWp, cosWp);

%     Ensure wp is in range [0, 2pi]
    if (wp < 0)
        wp = wp + 2*pi;
    end

%     Eccentricity = B/mu
    e = norm(B) / mu;

%     Get the orbit energy
    energy = (dot(v, v))/2 - mu/norm(r);

%     Compute the semi-major axis
    a = -mu/(2*energy);

% Compute the true anomaly
    nu = acos(dot(r, B)/(norm(r) * norm(B)));

%     Quadrant check per Vallado 2-86
    if(dot(r, v) < 0)
        nu = 2*pi - nu;
    end

end
```

Period Computation

```
function [TP] = Period(mu, a)

% Compute the Keplerian orbit period
     TP = 2 * pi * sqrt(a^3/mu);

end
```

Keplerian Element Output

```
function WriteKeplerianElements(fid, number, a, e, inc, raan, wp, nu)

% Formatted output of the input Keplerian elements.
% fid       File handle from file opened in main
% Number    Numerical value to insert in header
% a         Semi-major axis (assumed meters)
% e         Eccentricity
% inc       Inclination (assumed radians)
% raan      Right ascension of ascending node (assumed radians)
% wp        Argument of periapsis (assumed radians)
% nu        True anomaly(assumed radians)

    fprintf(fid, 'Keplerian Elements for Vector %d\n', number);
    fprintf(fid,'a:     %16.8f m\n', a);
    fprintf(fid,'e:     %16.8f\n', e);
    fprintf(fid,'inc:   %16.8f deg\n', rad2deg(inc));
    fprintf(fid,'raan:  %16.8f deg\n', rad2deg(raan));
    fprintf(fid,'wp:    %16.8f deg\n', rad2deg(wp));
    fprintf(fid,'nu:    %16.8f deg\n', rad2deg(nu));

end
```