```
% Spring 2024 AER E 351 Homework 04 Problem 1 (Problem 5.2)
% Matthew Mehrtens
clear, clc, close all;

%% Given
r_Earth = 1; % [au]
r_Jupiter = 5.2; % [au]

theta = 150; % [°]
a = 5; % [au]

cdu = 1.495978e8; % [km]
mu_sun = 1.327e11; % [km^3/s^2]
ctu = sqrt(cdu^3 / mu_sun); % [s]

Earth_siderial = 365.256; % [days]

ctu_to_years = @(t) t * ctu / 86400 / Earth_siderial; % [years]

%% Part a.)
c_fn = @(r_1, r_2, theta) sqrt(r_1^2 + r_2^2 ...
    - 2 * r_1 * r_2 * cos(theta)); % [distance]
a_m_fn = @(r_1, r_2, theta) (r_1 + r_2 + c_fn(r_1, r_2, theta)) ...
    / 4; % [distance]

% a_m_min occurs at theta = 0
% a_m_max occurs at theta = pi
% 0 <= theta < 2pi
a_m_min = a_m_fn(r_Earth, r_Jupiter, 0); % [au]
a_m_max = a_m_fn(r_Earth, r_Jupiter, pi); % [au]

%% Part b.)
a_m = a_m_fn(r_Earth, r_Jupiter, deg2rad(theta)); % [au]

s_fn = @(r_1, r_2, c) (r_1 + r_2 + c) / 2; % [distance]

c = c_fn(r_Earth, r_Jupiter, deg2rad(theta)); % [au]
s = s_fn(r_Earth, r_Jupiter, c); % [au]

% Equation 5.29
% beta_m = beta_m_0 for 0 <= theta <= pi
alpha_m = pi; % [rad]
beta_m = 2 * asin(sqrt((s - c) / s)); % [rad]

% Equation 5.30
t_m = sqrt(s^3 / 8) * (pi - beta_m + sin(beta_m)); % [ctu]

% Equation 5.26 and 5.27
alpha_0_fn = @(s, a) 2 * asin(sqrt(s / (2 * a))); % [rad]
beta_0_fn = @(s, c, a) 2 * asin(sqrt((s - c) / (2 * a))); % [rad]

alpha_0 = alpha_0_fn(s, a); % [rad]
beta_0 = beta_0_fn(s, c, a); % [rad]

alpha = alpha_0; % [rad]
alphasharp = 2 * pi - alpha_0; % [rad]

% Since theta < pi, beta = beta_0
```

```
beta = beta_0; % [rad]

lamberts_eqn = @(mu, a, alpha, beta) a^(3 / 2) ...
    * (alpha - beta - (sin(alpha) - sin(beta))); % [time]

t_F = lamberts_eqn(1, a, alpha, beta); % [ctu]
t_Fstar = lamberts_eqn(1, a, alphasharp, beta); % [ctu]

t_p_fn = @(mu, s, c, theta) sqrt(2) / (3 * mu) * (s^(3 / 2) ...
    - sign(sin(theta)) * (s - c)^(3 / 2)); % [time]

t_p = t_p_fn(1, s, c, deg2rad(theta)); % [ctu]

%% Part c.)
u_fn = @(r) r / norm(r); % [distance]
u_c_fn = @(r_1, r_2, c) (r_2 - r_1) / c; % [distance]
A_fn = @(mu, a, alpha) sqrt(mu / (4 * a)) ...
    * cot(alpha / 2); % [distance/time]
B_fn = @(mu, a, beta) sqrt(mu / (4 * a)) * cot(beta / 2); % [distance/time]
v_1_fn = @(A, B, u_1, u_c) (B + A) * u_c + (B - A) * u_1; % [distance/time]

r_1 = [r_Earth 0]; % [au]
r_2 = r_Jupiter * [cosd(theta) sind(theta)]; % [au]

u_1 = u_fn(r_1); % [au]
u_c = u_c_fn(r_1, r_2, c); % [au]
A = A_fn(1, a, alpha); % [EMOS]
Asharp = A_fn(1, a, alphasharp); % [EMOS]
B = B_fn(1, a, beta); % [EMOS]

v_1 = v_1_fn(A, B, u_1, u_c); % [EMOS]
v_1sharp = v_1_fn(Asharp, B, u_1, u_c); % [EMOS]

%% Part d.)
v_1_mag = norm(v_1); % [EMOS]
v_1sharp_mag = norm(v_1sharp); % [EMOS]

%% Part e.)
p_fn = @(a, s, c, r_1, r_2, alpha, beta) 4 * a * (s - r_1) * (s - r_2) ...
    / c^2 * sin((alpha + beta) / 2)^2; % [distance]

p = p_fn(a, s, c, norm(r_1), norm(r_2), alpha, beta); % [au]
ptilde = p_fn(a, s, c, norm(r_1), norm(r_2), alphasharp, beta); % [au]

e_fn = @(p, a) sqrt(1 - p / a); % []

e = e_fn(p, a); % []
etilde = e_fn(ptilde, a); % []

%% Output
fprintf(...
    "Problem 1.)a.)\n" + ...
    "%g au <= a_m <= %g au\n\n", ...
    a_m_min, a_m_max);
fprintf(...
    "Problem 1.)b.)\n" + ...
    "a_m = %g au\n\n" + ...
    "c = %g au\n" + ...
    "s = %g au\n\n" + ...
    "alpha_m = %g rad\n" + ...
    "beta_m = %g rad\n\n" + ...
    "t_m = %g ctu = %g years\n\n" + ...
    "alpha_0 = %g rad\n" + ...
    "beta_0 = %g rad\n" + ...
    "alpha = %g rad\n" + ...
    "alpha# = %g rad\n" + ...
    "beta = %g rad\n\n" + ...
    "t_F = %g ctu = %g years\n" + ...
    "t_F* = %g ctu = %g years\n\n" + ...
    "t_p = %g ctu = %g years\n\n", ...
    a_m, c, s, alpha_m, beta_m, t_m, ctu_to_years(t_m), alpha_0, ...
    beta_0, alpha, alphasharp, beta, t_F, ctu_to_years(t_F), ...
    t_Fstar, ctu_to_years(t_Fstar), t_p, ctu_to_years(t_p));
fprintf(...
    "Problem 1.)c.)\n" + ...
    "r_1 [au]:\n");
disp(r_1);
fprintf(...
    "r_2 [au]:\n");
disp(r_2);
fprintf(...
    "u_1:\n");
disp(u_1);
fprintf(...
    "u_c:\n");
disp(u_c);
fprintf(...
    "A = %g EMOS\n" + ...
    "A# = %g EMOS\n" + ...
    "B = %g EMOS\n\n" + ...
    "v_1 [EMOS]:\n", ...
    A, Asharp, B);
disp(v_1);
fprintf(...
    "v_1# [EMOS]:\n");
disp(v_1sharp);
fprintf(...
    "Problem 1.)d.)\n" + ...
    "|v_1| = %g EMOS\n" + ...
    "|v_1#| = %g EMOS\n\n", ...
    v_1_mag, v_1sharp_mag);
fprintf(...
    "Problem 1.)e.)\n" + ...
    "p = %g au\n" + ...
    "p~ = %g au\n\n" + ...
    "e = %g\n" + ...
    "e~ = %g\n", ...
    p, ptilde, e, etilde);
```