

```

% Spring 2024 AER E 351 Homework 07 Problem 1.)9.2.)
% Matthew Mehrstens
clear; clc; close all;

%% Equations
emos2kmpsec = @(v) v * 1.495978e8 / sqrt(1.495978e8^3 / 1.327e11); % [km/s]
kmpsec2emos = @(v) v * sqrt(1.495978e8^3 / 1.327e11) / 1.495978e8; % [EMOS]
au2km = @(d) d * 1.495978e8; % [km]
km2au = @(d) d / 1.495978e8; % [km]

% Velocity in a circular orbit with radius r.
v_c_fn = @(mu,r) sqrt(mu ./ r); % [velocity]

% mu is the mu of the sun, r_1 is the radius of the inner orbit, and r_2 is
% the radius of the outer orbit. Returns [v_periapse,v_apoapse].
v_Hohmann_fn = @(mu,r_1,r_2) [...
    sqrt(mu .* (2 ./ r_1 - 1 ./ ((r_1 + r_2) ./ 2))), ...
    sqrt(mu .* (2 ./ r_2 - 1 ./ ((r_1 + r_2) ./ 2)))]; % [velocity]

% Assumes massless, circular orbits. mu is relative to the sun, r_1 is the
% radius of the inner orbit, and r_2 is the radius of the outer orbit.
% Returns [deltav_1,deltav_2] where deltav_1 is the deltav required at the
% inner orbit and deltav_2 is the deltav required at the outer orbit.
deltav_Hohmann_fn = @(mu,r_1,r_2) ...
    abs(v_Hohmann_fn(mu,r_1,r_2) - v_c_fn(mu,[r_1,r_2])); % [velocity]

% Periapse velocity of a hyperbola. Mu is relative to the planet.
v_p_hyper_fn = @(mu,v_inf,r_park) ...
    sqrt(v_inf.^2 + 2 .* mu ./ r_park); % [velocity]

% deltav assuming the planets have mass.
deltav_fn = @(v_p_hyper,v_park) abs(v_p_hyper - v_park); % [velocity]

% mu is relative to the planet, v_inf is the excess escape speed for the
% hyperbolic orbit, and r_p_hyper is the periapse radius of the hyperbolic
% trajectory.
psi_fn = @(mu,v_inf,r_p_hyper) v_inf.^2 .* r_p_hyper ./ mu; % []

% Point of departure on a circular orbit for a hyperbolic escape
% trajectory. psi is calculated relative to the planet.
deltaby2_fn = @(psi) asin(1 / (1 + psi)); % [rad]

% Aiming radius in units of planetary radii. r_p_hyper is the periapse of
% the incoming hyperbolic trajectory, r_planet is the radius of the planet,
% mu is relative to the planet, and v_inf is the excess escape speed for
% the hyperbolic orbit. Give all distances in the same units.
Deltabyplanet_radius_fn = @(psi,r_p_hyper,planet_radius) ...
    r_p_hyper ./ planet_radius .* sqrt(1 + 2 ./ psi); % [planet radii]

%% Given
G = 6.67259e-20; % [km^3/(kg*s^2)]

m_Earth = 5.974e24; % [kg]
m_Mars = 0.1074 * m_Earth; % [kg]

mu_Sun = 1; % [cdu^3/ctu^2]
mu_Earth = G * m_Earth; % [km^3/s^2]
mu_Mars = G * m_Mars; % [km^3/s^2]

```

```

Earth_radius = 6.37812e3; % [km]
Mars_radius = 0.532 * Earth_radius; % [km]

r_Earth = 1; % [au]
r_Mars = 1.5237; % [au]
r_park_Earth = 1.1 * Earth_radius; % [km]
r_park_Mars = 1.3 * Mars_radius; % [km]

%% (a)
deltav_Sun = deltav_Hohmann_fn(mu_Sun,r_Earth,r_Mars); % [EMOS]
v_inf_Earth = deltav_Sun(1); % [EMOS]
v_inf_Mars = deltav_Sun(2); % [EMOS]

v_p_hyper = v_p_hyper_fn([mu_Earth,mu_Mars], ...
    emos2kmpsec([v_inf_Earth,v_inf_Mars]), ...
    [r_park_Earth,r_park_Mars]); % [km/s]

deltav = deltav_fn(v_p_hyper,...
    v_c_fn([mu_Earth,mu_Mars],[r_park_Earth,r_park_Mars])); % [km/s]

%% (b)
psi = psi_fn([mu_Earth,mu_Mars],emos2kmpsec([v_inf_Earth,v_inf_Mars]), ...
    [r_park_Earth,r_park_Mars]); % []

deltaby2_Earth = deltaby2_fn(psi(1)); % [rad]

%% (c)
Deltabyr_Mars = Deltabyplanet_radius_fn(psi(2),r_park_Mars,Mars_radius); % [Mars radii]

%% Output
fprintf( ...
    "Problem 1.)9.2.)\n" + ...
    "Part a.)\n" + ...
    "v_inf_Earth = %g EMOS = %g km/s\n" + ...
    "v_inf_Mars = %g EMOS = %g km/s\n" + ...
    "v_p_hyper_Earth = %g km/s\n" + ...
    "v_p_hyper_Mars = %g km/s\n" + ...
    "deltav_Earth = %g EMOS = %g km/s\n" + ...
    "deltav_Mars = %g EMOS = %g km/s\n" + ...
    "deltav_total = %g EMOS = %g km/s\n\n", ...
    v_inf_Earth,emos2kmpsec(v_inf_Earth),v_inf_Mars, ...
    emos2kmpsec(v_inf_Mars),v_p_hyper,kmpsec2emos(deltav(1)),deltav(1), ...
    kmpsec2emos(deltav(2)),deltav(2),kmpsec2emos(sum(deltav)),sum(deltav));
fprintf( ...
    "Part b.)\n" + ...
    "(Earth) delta / 2 = %g rad = %g°\n\n", ...
    deltaby2_Earth,rad2deg(deltaby2_Earth));
fprintf( ...
    "Part c.)\n" + ...
    "(Mars) Delta / Mars_radius = %g Mars radii\n", ...
    Deltabyr_Mars);

```