```
% Aidan Carey
%12/09/23
%project 2 phase 5
%description - in phase 5, and the drag functions work, we can explore the
%results by searching for the range of C that will give a range of 463
%feet. We can also find the max height, flight time, and energy lost from
%this drag coefficient.

% Constants
g = 32.2;  % acceleration due to gravity in ft/s^2
launch_angle = deg2rad(28);  % launch angle in radians
exit_velocity = 116 * 5280 / 3600;  % exit velocity in ft/s
% (converted from mph)
analytic_time_of_flight = 5.3;  % time of flight in seconds

% Additional constants
mass_of_baseball = 0.145;  % mass of a baseball in kg
rho_air = 0.00238;  % air density in slugs/ft^3

% User input for drag coefficient
C = input('Enter a value for drag coefficient (e.g., 0.38): ');

% Cross-sectional area of a baseball (approximated as a sphere)
radius_of_baseball = 0.06035;  % in feet
A = pi * radius_of_baseball^2;  % cross-sectional area

% Initial conditions
x0 = 0; y0 = 0;  % initial position
vx0 = exit_velocity * cos(launch_angle);  % initial x-component of velocity
vy0 = exit_velocity * sin(launch_angle);  % initial y-component of velocity

% Time settings
dt = 0.01;  % time step
t_max = analytic_time_of_flight;  % maximum time
t_values = 0:dt:t_max;

% Initialize arrays to store results
x_values_with_drag = zeros(size(t_values));
y_values_with_drag = zeros(size(t_values));

% Initial conditions
x_values_with_drag(1) = x0;
y_values_with_drag(1) = y0;
vx_with_drag = vx0;
vy_with_drag = vy0;

% Initialize variables to store information
time_of_flight = 0;
max_height = 0;
range_ft = 0;
final_speed = 0;

% Numerical computation using Euler's method with drag
for i = 2:length(t_values)
    % With drag
    v_with_drag = sqrt(vx_with_drag^2 + vy_with_drag^2);
    ax_with_drag = -0.5 * C * rho_air * A * v_with_drag * vx_with_drag ...
        / mass_of_baseball;
    ay_with_drag = -g - 0.5 * C * rho_air * A * v_with_drag * ...
```

```
        vy_with_drag / mass_of_baseball;

    % Update velocities and positions using Euler's method
    vx_with_drag = vx_with_drag + ax_with_drag * dt;
    vy_with_drag = vy_with_drag + ay_with_drag * dt;
    x_values_with_drag(i) = x_values_with_drag(i - 1) + vx_with_drag * dt;
    y_values_with_drag(i) = y_values_with_drag(i - 1) + vy_with_drag * dt;

    % Check for the end of the trajectory
    if y_values_with_drag(i) < 0
        % Grab information when the ball hits the ground
        time_of_flight = t_values(i);
        max_height = max(y_values_with_drag);
        range_ft = x_values_with_drag(i);
        final_speed = sqrt(vx_with_drag^2 + vy_with_drag^2);
        break;
    end
end

% Convert positions to feet
x_values_with_drag = x_values_with_drag * 3.28084;
y_values_with_drag = y_values_with_drag * 3.28084;

% Display results
disp(['Time of Flight: ', num2str(time_of_flight), ' seconds']);
disp(['Maximum Height: ', num2str(max_height), ' feet']);
disp(['Range: ', num2str(range_ft), ' feet']);
disp(['Final Speed: ', num2str(final_speed), ' ft/s']);

% Compute percent errors
known_range = 463;  % known range from mlb.statcast
percent_error_range = ((range_ft - known_range) / known_range) * 100;

known_max_height = 100;  % known max height from mlb.statcast
percent_error_max_height = ((max_height - known_max_height) /...
    known_max_height) * 100;

disp(['Percent Error in Range: ', num2str(percent_error_range), '%']);
disp(['Percent Error in Max Height: ', ...
    num2str(percent_error_max_height), '%']);

% Calculate initial kinetic energy
initial_speed = exit_velocity;  % Assuming the initial speed is the
% exit velocity
KE_initial = 0.5 * mass_of_baseball * initial_speed^2;

% Calculate final kinetic energy (after the FOR loop)
final_speed = sqrt(vx_with_drag(end)^2 + vy_with_drag(end)^2);
KE_final = 0.5 * mass_of_baseball * final_speed^2;

% Calculate energy lost
delta_KE = KE_initial - KE_final;

% Display the results
disp(['Initial Kinetic Energy: ', num2str(KE_initial), ' J']);
disp(['Final Kinetic Energy: ', num2str(KE_final), ' J']);
disp(['Energy Lost: ', num2str(delta_KE), ' J']);
```

```
>> project2phase5
Enter a value for drag coefficient (e.g., 0.38): 13.8
Time of Flight: 4.31 seconds
Maximum Height: 75.2674 feet
Range: 463.0216 feet
Final Speed: 101.3334 ft/s
Percent Error in Range: 0.0046723%
Percent Error in Max Height: -24.7326%
Initial Kinetic Energy: 2098.538 J
Final Kinetic Energy: 744.4628 J
Energy Lost: 1354.0752 J
>>
```