```matlab
% Sounak Ghosh
% 9/19/19
% ECE 202 - Fall 2019 - MATLAB Exercise M3
% Equation source: http://www.convertalot.com/elastic_collision_calculator.html
% MATLAB script to determine the number of collisions in a system of carts.

clear % clears all variables in the workplace; avoids common errors
clc   % clears all previous outputs in the command window

% -------- given information --------

M = [240 120 360];  % mass of the carts in g
Vi = [30 15 -45];  % initial velocity of carts in cm8/s
m12 = M(1) + M(2);   % mass of the carts in cart#1 & cart#2 in g
m23 = M(2) + M(3);   % sum of the masses of cart#1 & cart#2 in g

% --------- calculations ----------

Momentum_i = sum(M.*Vi); % Initial Momentum of the total system.
Energy_i = sum(0.5*M.*Vi.^2); % Initial Energy of the total system.

% From the diagram M2 & M3 are going to collide first.
% After Collision A

VfA(2) = (M(2)*Vi(2) - M(3)*Vi(2) + 2*M(3)*Vi(3))/ m23;

VfA(3) = (2*M(2)*Vi(2) - M(2)*Vi(3) + M(3)*Vi(3))/ m23;

VfA(1) = Vi(1)           % final velocities of Carts after Collision A.

CheckMomentum_A = sum(M.*VfA) - Momentum_i   % The change in total energy
                                             % momentum of the system before
CheckEnergy_A = sum(0.5*M.*VfA.^2) - Energy_i    % & after the collision should
                                                       % be zero.

% The velocities after collision A indicate carts #1 & #2 are going to collide.
% After Collision B.

VfB(1) = (M(1)*VfA(1) - M(2)*VfA(1) + 2*M(2)*VfA(2))/ m12;

VfB(2) = (2*M(1)*VfA(1) - M(1)*VfA(2) + M(2)*VfA(2))/ m12;

VfB(3) = VfA(3)          % Energy of the total system after collision B.

CheckMomentum_B = sum(M.*VfB) - Momentum_i   % The change in total energy
                                              % momentum of the system before
CheckEnergy_B = sum(0.5*M.*VfB.^2) - Energy_i   % & after the collision should
                                              % be zero.

```

```matlab
% The velocities after collision B indicate carts #2 & #3 are going to collide.
% After Collision C.

VfC(2) = (M(2)*VfB(2) - M(3)*VfB(2) + 2*M(3)*VfA(3))/ m23;

VfC(3) = (2*M(2)*VfB(2) - M(2)*VfA(3) + M(3)*VfA(3))/ m23;

VfC(1) = VfB(1)      % Energy of the total system after collision C.

CheckMomentum_C = sum(M.*VfC) - Momentum_i   % The change in total energy
                                              % momentum of the system before
CheckEnergy_C = sum(0.5*M.*VfC.^2) - Energy_i     % & after the collision should
                                              % be zero

% The velocities after collision C indicate carts #1 & #2 are going to collide.
% After Collision D.

VfD(1) = (M(1)*VfC(1) - M(2)*VfC(1) + 2*M(2)*VfC(2))/ m12;

VfD(2) = (2*M(1)*VfC(1) - M(1)*VfC(2) + M(2)*VfC(2))/ m12;

VfD(3) = VfC(3)    % Energy of the total system after collision D.

CheckMomentum_D = sum(M.*VfD) - Momentum_i   % The change in total energy
                                              % momentum of the system before
CheckEnergy_D = sum(0.5*M.*VfD.^2) - Energy_i    % & after the collision should
                                              % be zero

Total_P = sum(M .* VfD) -  Momentum_i        % Total momentum of system after all collisions
Total_E = sum( 0.5.*M.* VfD.^2) - Energy_i    % Total energy of system after all collisions

% The final velocities of the carts 1, 2 and 3 after collision D are -50
% cm/s, -35 cm/s and 25 cm/s respectively, which indicate that cart#1 adn
% cart#2 are moving in the same direction with cart#1 moving faster than
% cart 2, hence carts #1 and #2 will never collide, and cart#3 is moving in
% the opposite direction of cart#1 and cart#2, therefore cart#3 won't have
% a collision with either of the other carts. In  conclusion,
% we can determine that the carts will not have any further collisions.
```