```
% Ian Woodbury
% 9/15/21
% ECE 202 Fall 2021 MATLAB Exercise M6
% Three carts colliding elastically, find the resulting velocities of each
% collision, and determine where collisions occur and between which carts
% using a while loop. Take input for carts and the cutoff for checks

clear

% ------ Input info and initial conditions------
global m;   % m, array of masses, and M, combined mass are stated as globals
global M;   % for use in the function, "collision" at the bottom of the script
m = input("Enter masses in order, in brackets \n");
% the masses of 1,2,3, repectively, in grams
v = input("Enter velocities in order, in brackets \n");
% the initial velocities in cm/s, not labeled vi, since the array will change
% througout the program
cut = input("Enter the cutoff value for checks of momentum and energy \n");
% gets user input for the value at which the checks fail, and the program
% terminates (cutoff)
disp(" ")
% wanted to skip a line here to make output more legible

% ------ inital momentum and energy calculations ------

pi = sum(v.*m);           % initial momentum of the system in g*cm/s
KEi = sum((1/2)*m.*v.^2); % initial KE of the system in ergs

% ------ loop start ------
count = 0;

while v(1) > v(2) || v(2) > v(3)

    % ----- Check for collision -------

    % varaiable next for the other cart, and marks the cart colliding with cart
    % 2, since cart 2 is always in the collision
    if v(1) > v(2) && v(2) > v(3)
        next = input("Which cart collides with cart 2? \n");
    elseif v(1) > v(2)
        next = 1;
    else
        next = 3;
    end

    % ------ Outputting values and calculations ------

    count = count + 1; % counter for collisions
    M = m(2) + m(next); % combined mass of colliding carts, in grams
    disp("Collision " + count)
    v = collision(v, next)

    % ------ Momentum and Energy Check ------

    dp = abs(pi - sum(v.*m)); % The check for conservation of momentum in the
    % system in g*cm/s, which subracts the momentum of the current array from the
    % initial should be zero if momentum is conserved, uses ABS() to make
    % for cutoff value check
    dKE = abs(KEi - sum((1/2)*m.*v.^2)); % The check for conservation of energy(E)
    % in the system, in ergs, which subtracts the value of KE of the current
    % array from the initial value, and should return zero since E is conserved
    % shockingly, the d in front of both checks means "change in" of KE or p
    % uses ABS() to make for cutoff value check

    % if statements determine which check values to print in the case they are
    % greater than the cutoff value

    if dp > cut
        dp
    end
    if dKE > cut
        dKE
    end

end

if count == 0
    disp("There are no collisions")
else
    disp("There are no more collisions")
end

% function "collision" to calculate the velocity array after each collision
% and update it. Takes next as the variable for the cart colliding with
% cart 2, which is always in the collision
function vf = collision(v0, next)
    global m M;
    vf(next) = (2*m(2)*v0(2) + v0(next)*(m(next)-m(2)))/M;
    vf(2) = (2*m(next)*v0(next) + v0(2)*(m(2)-m(next)))/M;
    vf(4 - next) = v0(4 - next);
end

% The design is successful, and the output matches that of M3 when the same
% values are used

```