% Simpson's 3/8 Rule

syms x; % Declare symbolic variable

fun = input('Enter the function f(x) (e.g., x^2+2\*x): ', 's'); % Input function as a string

f = str2func(['@(x) ' fun]); % Convert the input string to a function handle

% Input integration limits and number of partitions

x\_1 = input('Enter the lower limit a: '); % Lower limit of integration

x\_n = input('Enter the upper limit b: '); % Upper limit of integration

n = input('Enter the number of partitions n: '); % Number of partitions

% Verify if the number of intervals is divisible by 3

if mod(n, 3) ~= 0

disp("Simpson's 3/8 Rule is not applicable (requires number of intervals to be a multiple of 3).");

else

% Calculate step size

h = (x\_n - x\_1) / n;

% Generate x and y values

x\_values = x\_1:h:x\_n; % Generate x values

y\_values = arrayfun(f, x\_values); % Evaluate f(x) at each x value

% Apply Simpson's 3/8 Rule

A = y\_values(1) + y\_values(end); % Add first and last terms

for i = 2:n

if mod(i-1, 3) == 0 % Coefficient 2 for every third term

A = A + 2 \* y\_values(i);

else % Coefficient 3 for other terms

A = A + 3 \* y\_values(i);

end

end

A = (3 \* h / 8) \* A; % Final area computation

% Display the result

fprintf('The area under the curve using Simpson''s 3/8 Rule is: %.6f\n', A);

end