

```
function Q2()
    % Define the function and interval
    f = @(x) 1 ./ (1 + x);
    a = 0;
    b = 4;
    exact = log(5); % Exact integral value ln(5)

    % Different values of n (number of subintervals)
    n_values = [2, 4, 8];
    num_n = length(n_values);

    % Initialize error arrays
    errors_trap = zeros(num_n, 1);
    errors_simp = zeros(num_n, 1);

    for i = 1:num_n
        n = n_values(i);
        h = (b - a) / n;

        % Composite Trapezoidal Rule
        x = linspace(a, b, n + 1);
        y = f(x);
        trap = (h / 2) * (y(1) + 2 * sum(y(2:end-1)) + y(end));
        errors_trap(i) = abs(trap - exact);

        % Composite Simpson's Rule
        if mod(n, 2) ~= 0
            error('Simpson's rule requires an even number of subintervals.');
        end
        sum_odds = sum(y(2:2:end-1));
        sum_evens = sum(y(3:2:end-2));
        simp = (h / 3) * (y(1) + 4 * sum_odds + 2 * sum_evens + y(end));
        errors_simp(i) = abs(simp - exact);
    end

    % Display results
    fprintf('Subintervals\tTrapezoidal Error\tSimpson Error\n');
    for i = 1:num_n
        fprintf('%d\t\t\t%.4e\t\t%.4e\n', n_values(i), errors_trap(i), errors_simp(i));
    end
end
```