```
clc;
clear;
% ODE definition and exact solution
f_dash = @(t, y) -1000 * (y - (t + 2)) + 1;
exact_sol = @(t) -exp(-1000*t) + t + 2;
% Step sizes to try
h_values = [5e-4, 2e-3, 2.5e-3];
t_final = 0.01;

for k = 1:length(h_values)
    h = h_values(k);
    t = 0:h:t_final;
    n = length(t);

    y_euler = zeros(size(t));
    y_euler(1) = 1;

    y_rk4 = zeros(size(t));
    y_rk4(1) = 1;

    y_exact = exact_sol(t);

    % Euler Method
    for i = 1:n-1
        y_euler(i+1) = y_euler(i) + h * f_dash(t(i), y_euler(i));
    end

    % RK4 Method
    for i = 1:n-1
        k1 = f_dash(t(i), y_rk4(i));
        k2 = f_dash(t(i) + 0.5*h, y_rk4(i) + 0.5*h*k1);
        k3 = f_dash(t(i) + 0.5*h, y_rk4(i) + 0.5*h*k2);
        k4 = f_dash(t(i) + h, y_rk4(i) + h*k3);
        y_rk4(i+1) = y_rk4(i) + (h/6)*(k1 + 2*k2 + 2*k3 + k4);
    end

    % Error calculation
    err_euler = abs(y_euler - y_exact);
    err_rk4 = abs(y_rk4 - y_exact);

    % Print results at final t
    fprintf("Step size h = %.4g\n", h);
    fprintf("  Euler: y(0.01) = %.8f | Error = %.2e\n", y_euler(end),
err_euler(end));
    fprintf("  RK4:   y(0.01) = %.8f | Error = %.2e\n", y_rk4(end),
err_rk4(end));
    fprintf("  Exact: y(0.01) = %.8f\n\n", y_exact(end));
end
```

```
Step size h = 0.0005
  Euler: y(0.01) = 2.00999905 | Error = 4.44e-05
  RK4:   y(0.01) = 2.00995424 | Error = 3.61e-07
  Exact: y(0.01) = 2.00995460
```

```
Step size h = 0.002
  Euler: y(0.01) = 3.01000000 | Error = 1.00e+00
  RK4:   y(0.01) = 2.00588477 | Error = 4.07e-03
  Exact: y(0.01) = 2.00995460

Step size h = 0.0025
  Euler: y(0.01) = -3.05250000 | Error = 5.06e+00
  RK4:   y(0.01) = 1.83320398 | Error = 1.77e-01
  Exact: y(0.01) = 2.00995460
```

Published with MATLAB® R2025a