

---

```

function yp_solution()
    % Define the differential equation as a system of first order equations
    function dy = system(t, y)
        y1 = y(1);
        y1_prime = y(2);
        y2 = y(3);
        y2_prime = y(4);
        W = y1*y2_prime - y1_prime*y2;

        % Define the nonhomogeneous part
        f = 9*t^2 + 4*t;

        u1_prime = -y2*f/W;
        u2_prime = y1*f/W;

        % The equations derived from the given ODE
        y1_double_prime = (y1 - t*y1_prime)/t^2;
        y2_double_prime = (y2 - t*y2_prime)/t^2;

        dy = [y1_prime; y1_double_prime; y2_prime; y2_double_prime];
    end

    % Initial conditions: Assuming y1 and y2 are solutions of the homogeneous
    equation
    y1_0 = 1;
    y1_prime_0 = 0;
    y2_0 = 0;
    y2_prime_0 = 1;

    [x, sol] = ode45(@system, [1, 5], [y1_0, y1_prime_0, y2_0, y2_prime_0]);

    y1 = sol(:,1);
    y1_prime = sol(:,2);
    y2 = sol(:,3);
    y2_prime = sol(:,4);

    % Calculate Wronskian
    W = y1.*y2_prime - y1_prime.*y2;

    % Calculate the particular solution using Cramer's rule
    f = 9*x.^2 + 4*x;

    % Integrate using Simpson's rule
    u1 = zeros(size(x));
    u2 = zeros(size(x));

    for i = 2:length(x)
        u1(i) = u1(i-1) + (x(i) - x(i-1)) * (-y2(i) * f(i) / W(i));
        u2(i) = u2(i-1) + (x(i) - x(i-1)) * (y1(i) * f(i) / W(i));
    end

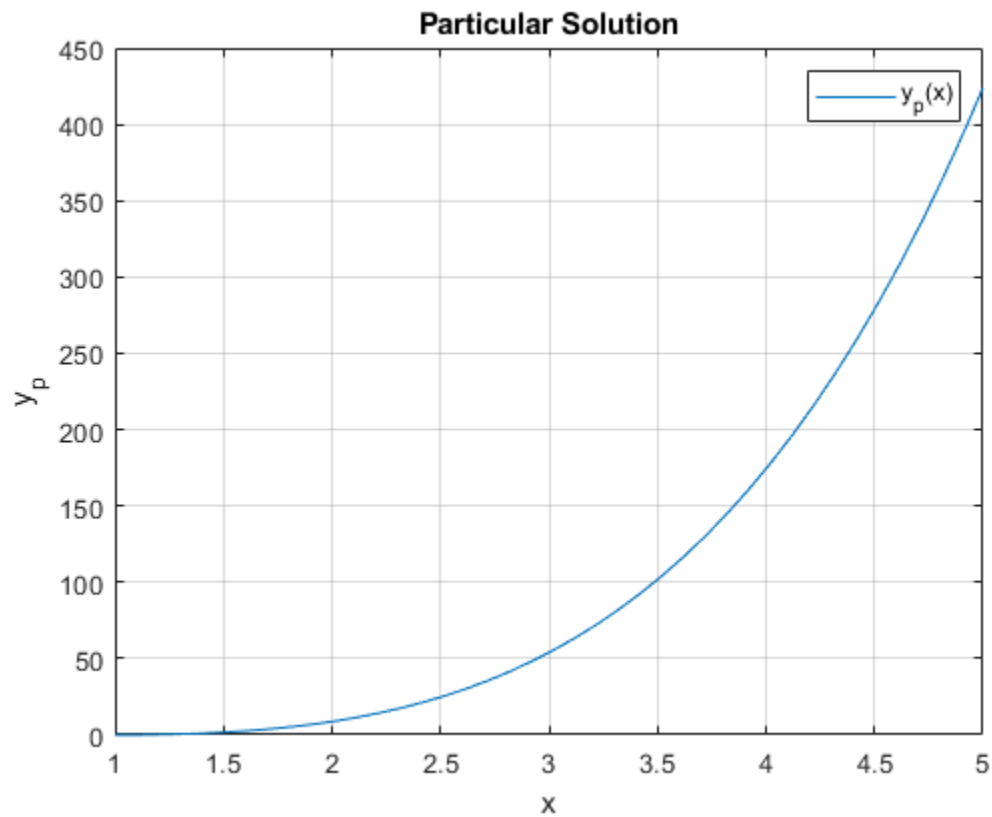
    y_p = y1.*u1 + y2.*u2;

```

---

---

```
% Plotting the particular solution
plot(x, y_p, 'DisplayName', 'y_p(x)');
title('Particular Solution');
xlabel('x');
ylabel('y_p');
legend();
grid on;
end
```



*Published with MATLAB® R2023a*