```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% 1 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

I = 2e-6;
R = 1e3;
C = 3e-6;
T = 20e-3;

h_vec = T./(2.^ [0:10]');

N = size(h_vec,1);

err_vec = zeros(N,1);

for i = 1:N
    h    = h_vec(i);
    tn   = [0:h:T]';
    vn   = R*I * (1 - exp(-tn/(R*C)));
    iCn  = I * exp(-tn/(R*C));
    dvdt = [((vn(2:end)) - vn(1:end-1)) ./ (tn(2:end) - tn(1:end-1)) ;
            (vn(end) - vn(end-1)) ./ (tn(end) - tn(end-1))];
    iCh  = C*dvdt;
    en   = iCn - iCh;
    err_vec(i) = max(abs(en));
end

setfonts();
[p,C2] = order_estimate(h_vec, err_vec)

val  = 1e-3;
fun  = @(t) R*I * (1 - exp(-t/(R*C))) - val;
dfun = @(t) I/C * exp(-t/(R*C));

toll  = 1e-10;
t_0   = T/2;
n_max = 1000;
[t_star_vec, k_new] = newton(t_0, n_max, toll, fun, dfun);

format short e;
t_star = t_star_vec(end)
k_new

% sanity check
R*I .* (1 - exp(-t_star/(R*C)))

% figure(3);
% x = linspace(0, 0.01, 1000);
% y = fun(x);
% plot(x, y, '-b')
% % y2 = dfun(x) * 1e-3;
% % plot(x, y, '-b', x, y2, '-g')
% grid on
% hold on % mi permette di disegnare un'altra figura
% plot(x, zeros(length(x),1), '-r'); % '-r' = red
% xlabel('x');
% ylabel('y');
% title('y = f(x)');

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% 2 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

a = 0;
```

```
b = T;
nvec = [1:10]';
xx = [a: (b-a)/10000 :b]';
fun = @(t) I * exp(-t/(R*C));
iCex = fun(xx);

N = size(nvec,1);
err_vec = zeros(N,1);

for i = 1:N
    n = nvec(i);
    h = (b-a)/n;
    xi = [a:h:b]';
    fi = fun(xi);
    Pn = polyfit(xi,fi,n);
    i_c = polyval(Pn, xx);
    er = iCex - i_c;
    err_vec(i) = max(abs(er));
end

figure(1)
semilogy([1:N], err_vec, 'b*')
xlabel('N')
ylabel('Errore')
title('Errore interpolazione')

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% 3 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

N = 10;
r = 2;
Mh_vec  = 2.^ [1:N]';
h_vec   = zeros(N,1);
err_vec = zeros(N,1);

for i = 1:N
    h_vec(i) = (b-a) / Mh_vec(i);
    xi = [a:h_vec(i):b]';

    % fi = fun(xi);
    % i_c = interp1(xi, fi, xx);
    % er = iCex - i_c;
    % err_vec(i) = max(abs(er));

    i_c = zeros(size(xx,1),1);
    index = 1;
    H = h_vec(i);
    for j = 1:Mh_vec(i)
        xi  = [a + H*(j-1): H/r : a + H*j];
        fi  = fun(xi);
        Pn  = polyfit(xi,fi,r);
        while xx(index) < xi(end)
            i_c(index) = polyval(Pn, xx(index));
            index = index + 1;
        end
        if j == Mh_vec(i) & index <= size(xx,1)
            i_c(index) = polyval(Pn, xx(index));
            index = index + 1;
        end
        if j == Mh_vec(i)
            assert(index-1 == size(i_c,1));
        end
    end
    er = iCex - i_c;
    err_vec(i) = max(abs(er));

    % if i == 1
    %     figure(4);
    %     plot(xx, iCex, '-b', xx, i_c, '-g')
    %     grid on
    % end
end

figure(2)
loglog(h_vec, err_vec, '-x');
xlabel('h');
ylabel('Error');
title('Errore interpolazione')
grid on;

format short e;
h_vec
err_vec
[p,C2] = order_estimate(h_vec, err_vec)
```