```
function [R_per_unit , X_per_unit , B_per_unit]= e230512_p2(text_path,
 library_path)  %resistance (R), reactance (X) and susceptance (B)

format longg;

[S_base, V_base, N_circuit, N_bundle, d_bundle, length,conductor_name,
 outside_diameter, R_AC, GMR_conductor] = e230512_p1(text_path, library_path);
Z_base = V_base ^ 2 / S_base;

%DISTANCE calculations
[phase_A , phase_B , phase_C]=phase_locations(text_path, N_circuit);
[phase_A_image , phase_B_image , phase_C_image]=phase_locations(text_path,
 N_circuit);
phase_A_image(2)=-phase_A(2);
phase_A_image(4)=-phase_A(4);
phase_B_image(2)=-phase_B(2);
phase_B_image(4)=-phase_B(4);
phase_C_image(2)=-phase_C(2);
phase_C_image(4)=-phase_C(4);
%distances between phases
distance_a_a=distance(phase_A,phase_A);
distance_b_b=distance(phase_B,phase_B);
distance_c_c=distance(phase_C,phase_C);
distance_a_b=distance(phase_A,phase_B);
distance_a_c=distance(phase_A,phase_C);
distance_b_c=distance(phase_B,phase_C);


%adjusting according to bundle geometry
switch N_bundle
    case 1
        GMR_bundle=  GMR_conductor;
        r_eq_bundle=  outside_diameter/2;
    case 2
        GMR_bundle= nthroot( GMR_conductor * d_bundle , 2 );
        r_eq_bundle=nthroot(outside_diameter/2 * d_bundle , 2);
    case 3
        GMR_bundle= nthroot( GMR_conductor * d_bundle^2 , 3 );
        r_eq_bundle=nthroot(outside_diameter/2 * d_bundle^2 , 3);
    case 4
        GMR_bundle=
 nthroot( GMR_conductor*d_bundle*d_bundle*d_bundle*sqrt(2) , 4);
        r_eq_bundle=
 nthroot( outside_diameter/2*d_bundle*d_bundle*d_bundle*sqrt(2) , 4);
    case 5
        GMR_bundle= nthroot( GMR_conductor * d_bundle^2 *
 (2*d_bundle*cos(36))^2 , 5 );
        r_eq_bundle=nthroot(outside_diameter/2 * d_bundle^2 *
 (2*d_bundle*cos(36))^2 , 5);
    case 6
        GMR_bundle= nthroot( GMR_conductor * d_bundle^2 * (d_bundle*sqrt(3))^2
 * 2*d_bundle, 6 );
```

```
        r_eq_bundle=nthroot(outside_diameter/2 * d_bundle^2 *
 (d_bundle*sqrt(3))^2 * 2*d_bundle, 6 );
    case 7
        GMR_bundle= nthroot( GMR_conductor * d_bundle^2 *
 (d_bundle*1.801937736)^2 * (d_bundle*2.246979604)^2 , 7 );
        r_eq_bundle=nthroot(outside_diameter/2 * d_bundle^2 *
 (d_bundle*1.801937736)^2 * (d_bundle*2.246979604)^2 , 7 );
    case 8
        GMR_bundle= nthroot( GMR_conductor * d_bundle^2 *
 (d_bundle*sin(135/2)*2)^2 * (d_bundle+d_bundle*sqrt(2))^2 *
 d_bundle*2.61312593 , 8 );
        r_eq_bundle=nthroot(outside_diameter/2 * d_bundle^2 *
 (d_bundle*sin(135/2)*2)^2 * (d_bundle+d_bundle*sqrt(2))^2 *
 d_bundle*2.61312593 , 8 );
end

switch N_circuit
    case 1
        %GMD's between phases
        GMD_a_b=  distance_a_b(1);
        GMD_b_c = distance_b_c(1);
        GMD_a_c=  distance_a_c(1);
        GMD_eq=  nthroot( GMD_a_b*GMD_b_c*GMD_a_c , 3);

        %GMR calculations
        GMR_eq = GMR_bundle;

        %r_eq calculations
        r_eq = r_eq_bundle;

        %earth effect calculations
        H_1=distance(phase_A,phase_A_image);
        H_2=distance(phase_B,phase_B_image);
        H_3=distance(phase_C,phase_C_image);

        H_1_2= distance(phase_A_image,phase_B);
        H_2_3= distance(phase_B_image,phase_C);
        H_3_1= distance(phase_C_image,phase_A);

        nom=nthroot( H_1_2(1)  *  H_2_3(1)  *  H_3_1(1)  ,3);
        den=nthroot( H_1(1)  *  H_2(1)  *  H_3(1),3);
        earth_effect =log(  nom/den   );
    case 2
        %GMD's between phases
        GMD_a_b=  nthroot( (distance_a_b(1)*distance_a_b(2))^2 , 4);
        GMD_b_c = GMD_a_b; %same symmetry
        GMD_a_c=  nthroot( ( distance_a_c(1)*distance_a_c(2) )^2 , 4);
        GMD_eq=  nthroot( GMD_a_b*GMD_b_c*GMD_a_c , 3);

        %GMR calculations
        GMR_c1_c2 = nthroot( GMR_bundle * distance_a_a(2) , 2);
        GMR_a1_a2 = GMR_c1_c2;
        GMR_b1_b2 = nthroot( GMR_bundle*distance_b_b(2) , 2);
        GMR_eq = nthroot( GMR_a1_a2*GMR_c1_c2*GMR_b1_b2 , 3);
```

```
        %r_eq calculations
        r_eq_a= nthroot( r_eq_bundle * distance_a_a(2) , 2);
        r_eq_b= nthroot( r_eq_bundle * distance_b_b(2) , 2);
        r_eq_c= nthroot( r_eq_bundle * distance_c_c(2) , 2);
        r_eq= nthroot( r_eq_a * r_eq_b * r_eq_c , 3);

        %earth effect calculations
        H_1=distance(phase_A,phase_A_image);
        H_2=distance(phase_B,phase_B_image);
        H_3=distance(phase_C,phase_C_image);

        H_1_2= distance(phase_A_image,phase_B);
        H_2_3= distance(phase_B_image,phase_C);
        H_3_1= distance(phase_C_image,phase_A);

        nom=nthroot( H_1_2(1)*H_1_2(2)*H_1_2(3)*H_1_2(4)   *
 H_2_3(1)*H_2_3(2)*H_2_3(3)*H_2_3(4)   *
 H_3_1(1)*H_3_1(2)*H_3_1(3)*H_3_1(4),12);
        den=nthroot( H_1(1)*H_1(2)*H_1(3)*H_1(4)   *
 H_2(1)*H_2(2)*H_2(3)*H_2(4)   *   H_3(1)*H_3(2)*H_3(3)*H_3(4),12);
        earth_effect =log(   nom/den    );

end


%RESISTANCE calculation
R_per_unit = (   R_AC * length   ) / (   N_bundle * N_circuit   )   / Z_base

%INDUCTANCE calculation
L=2*10^-7*log(GMD_eq/GMR_eq);
X_per_unit = 2*pi*50*L*length /   Z_base

%CAPACITANCE calculation
C= 2*pi* 8.854187817620 *10^-12 / ( log(GMD_eq/r_eq)-earth_effect );
B_per_unit= 2*pi *50*C*length * Z_base

end

Not enough input arguments.

Error in e230512_p2 (line 5)
[S_base, V_base, N_circuit, N_bundle, d_bundle, length,conductor_name,
 outside_diameter, R_AC, GMR_conductor] = e230512_p1(text_path, library_path);
```

Published with MATLAB® R2021b

```
function d= distance(A, B)

    d(1)= sqrt(  ( A(1)-B(1) )^2 + ( A(2)-B(2) )^2  );
    d(2)= sqrt(  ( A(1)-B(3) )^2 + ( A(2)-B(4) )^2  );
    d(3)= sqrt(  ( A(3)-B(1) )^2 + ( A(4)-B(2) )^2  );
    d(4)= sqrt(  ( A(3)-B(3) )^2 + ( A(4)-B(4) )^2  );

end
```

Published with MATLAB® R2021b

```
function [phase_A , phase_B , phase_C]= phase_locations(text_path, N_circuit)
  %resistance (R), reactance (X) and susceptance (B)

input_file= fopen(text_path);

for i = 1:14
    fgetl(input_file);
end  %skips the initial info and arrives at phase locations


fgetl(input_file); %15

phase_C(1)=str2double(fgetl(input_file));
phase_C(2)=str2double(fgetl(input_file));

fgetl(input_file); %18

phase_A(1)=str2double(fgetl(input_file));
phase_A(2)=str2double(fgetl(input_file));

fgetl(input_file); %21

phase_B(1)=str2double(fgetl(input_file));
phase_B(2)=str2double(fgetl(input_file));


if(N_circuit==2)

    fgetl(input_file); %24

    phase_C(3)=str2double(fgetl(input_file));
    phase_C(4)=str2double(fgetl(input_file));

    fgetl(input_file); %27

    phase_A(3)=str2double(fgetl(input_file));
    phase_A(4)=str2double(fgetl(input_file));

    fgetl(input_file); %30

    phase_B(3)=str2double(fgetl(input_file));
    phase_B(4)=str2double(fgetl(input_file));

else
    phase_C(3)=0;
    phase_C(4)=0;

    phase_A(3)=0;
    phase_A(4)=0;

    phase_B(3)=0;
    phase_B(4)=0;
```

```
end
```

```
end
```

```
Not enough input arguments.

Error in phase_locations (line 4)
input_file= fopen(text_path);
```

Published with MATLAB® R2021b