```
clc;
clear;

V_r_kV     = input('Receiving end voltage (kV): ');        % kV
P_r_MW     = input('Receiving end power (MW): ');          % MW
pf         = input('Power factor (lagging +ve, leading -ve): ');
len_km     = input('Line length (km): ');                  % km
R_per_km   = input('Resistance per km (ohm/km): ');
X_per_km   = input('Reactance per km (ohm/km): ');         % Ω/km
B_per_km   = input('Susceptance per km (S/km): ');         % S/km

V_r   = V_r_kV * 1e3;         % Convert kV to V
P_r   = P_r_MW * 1e6;         % Convert MW to W
Vr_ph = V_r / sqrt(3);        % Phase voltage (V)

I_r_mag = P_r / (sqrt(3) * V_r * pf);
phi     = acos(pf);
I_r     = I_r_mag * (cos(phi) - 1i*sin(phi));  % Complex (phasor) form

R_total = R_per_km * len_km;      % Ω
X_total = X_per_km * len_km;      % Ω
B_total = B_per_km * len_km;      % S

Z = R_total + 1i * X_total;       % Series impedance (Ω)
Y = 1i * B_total;                 % Shunt admittance (S)

if len_km < 80
    model = 'Short';
elseif len_km <= 250
    model = 'Medium';
else
    model = 'Long';
end
disp(['Selected Model: ', model]);

switch model
    case 'Short'
        A = 1; B = Z; C = 0; D = 1;
        Vs_ph = Vr_ph + I_r * Z;
        I_s   = I_r;

    case 'Medium'
        A = 1 + (Y * Z) / 2;
        B = Z * (1 + (Y * Z) / 4);
        C = Y;
        D = A;
        Vs_ph = A * Vr_ph + B * I_r;
        I_s   = C * Vr_ph + D * I_r;

    case 'Long'
        gamma = sqrt(Z * Y);
        Zc    = sqrt(Z / Y);
```

```
        A = cosh(gamma * len_km * 1e3);
        D = A;
        B = Zc * sinh(gamma * len_km * 1e3);
        C = sinh(gamma * len_km * 1e3) / Zc;
        Vs_ph = A * Vr_ph + B * I_r;
        I_s   = C * Vr_ph + D * I_r;
end

Vs_line = abs(Vs_ph) * sqrt(3);  % Line voltage (V)
Is_line = abs(I_s);              % Line current (A)

P_s        = 3 * abs(Vs_ph) * abs(I_s) * cos(angle(Vs_ph) - angle(I_s)); %
Sending end power
efficiency = (P_r / P_s) * 100;
VR         = ((Vs_line - V_r) / V_r) * 100;

disp('--- RESULTS ---');
fprintf('Line Impedance (Z)   : %.3f + j%.3f Ω\n', real(Z), imag(Z));
fprintf('Receiving End Current: %.2f ∠ %.2f° A\n', abs(I_r),
rad2deg(angle(I_r)));
fprintf('Sending End Voltage  : %.2f kV\n', Vs_line / 1e3);
fprintf('Sending End Current  : %.2f A\n', Is_line);
fprintf('Efficiency           : %.2f %%\n', efficiency);
fprintf('Voltage Regulation   : %.2f %%\n', VR);

Error using input
Cannot call INPUT from EVALC.

Error in TransmissionLine_calc (line 4)
V_r_kV     = input('Receiving end voltage (kV): ');       % kV
             ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
```

Published with MATLAB® R2025a