

Contents

- El programa resuelve un circuito eléctrico aplicando el análisis de nodos.
- Valores de los elementos pasivos.
- Valores de la fuentes de excitación.
- Construcción de las matrices de datos del circuito.
- Construcción de la matriz R.
- Construcción del vector de fuentes de excitación.
- Cálculo de los voltajes de nodos.
- Cálculo de los voltajes para cada elemento pasivo del circuito.
- Cálculo de las corrientes para cada elemento pasivo del circuito.
- Cálculo de las potencias para cada elemento pasivo del circuito.
- Tabla con un resumen total de las variables del circuito.

El programa resuelve un circuito eléctrico aplicando el análisis de nodos.

```
clc  
clear
```

Valores de los elementos pasivos.

Datos que se introducen de forma manual.

```
R1 = 1;  
R2 = 2;  
R3 = 3;  
R4 = 4;  
R5 = 5;  
R6 = 6;  
R7 = 7;  
R8 = 8;
```

Valores de la fuentes de excitación.

Datos que se introducen de forma manual.

```
If1 = 1;  
If2 = 2;
```

Construcción de las matrices de datos del circuito.

Datos que se introducen de forma manual. Valor Nodo_envío Nodo_recepción

```
Datos = [ R1      1      2  
          R2      2      3  
          R3      2      4  
          R4      3      5  
          R5      6      7  
          R6      4      5  
          R7      5      8
```

```

R8      7      8];

%      Valor  Nodo_envío  Nodo_recepción
fuentes = [ If1      6      1
            If2      6      4];

[R_d,C_d] = size(Datos);
[R_f,C_f] = size(fuentes);

```

Construcción de la matriz R.

```

Ref = max(max(Datos(:,2:3)));
Matriz_R = zeros(Ref);

for k=1:R_d
    Matriz_R(Datos(k,2),Datos(k,3))=-1/Datos(k,1);
    Matriz_R(Datos(k,3),Datos(k,2))=-1/Datos(k,1);
end
clear k

DD=-sum(Matriz_R,2);

Matriz_R = Matriz_R + diag(DD);

Matriz_R(:,end)=[];
Matriz_R(end,:)=[];

```

Construcción del vector de fuentes de excitación.

```

Matriz_f = zeros(Ref,1);

for k=1:R_f
    Matriz_f(fuentes(k,2),1)=-fuentes(k,1)+Matriz_f(fuentes(k,2),1);
    Matriz_f(fuentes(k,3),1)=+fuentes(k,1)+Matriz_f(fuentes(k,3),1);
end
Matriz_f(end,:)=[];

```

Cálculo de los voltajes de nodos.

```

disp('Voltajes nodales con respecto al nodo de referencia (V)')
Vnodos = [ Matriz_R\Matriz_f
           0      ]

```

Voltajes nodales con respecto al nodo de referencia (V)

Vnodos =

```

30.4000
29.4000
26.6000
30.6000
21.0000
-39.0000
-24.0000
0

```

Cálculo de los voltajes para cada elemento pasivo del circuito.

```
disp('Voltajes en los elementos pasivos (V)')
V_R = Vnodos(Datos(:,2))-Vnodos(Datos(:,3))
```

Voltajes en los elementos pasivos (V)

V_R =

```
1.0000
2.8000
-1.2000
5.6000
-15.0000
9.6000
21.0000
-24.0000
```

Cálculo de las corrientes para cada elemento pasivo del circuito.

```
disp('Corriente en los elementos pasivos (A)')
I_R = V_R./Datos(:,1)
```

Corriente en los elementos pasivos (A)

I_R =

```
1.0000
1.4000
-0.4000
1.4000
-3.0000
1.6000
3.0000
-3.0000
```

Cálculo de las potencias para cada elemento pasivo del circuito.

```
disp('Potencia en los elementos pasivos (W)')
P_R = V_R.*I_R
```

Potencia en los elementos pasivos (W)

P_R =

```
1.0000
3.9200
0.4800
```

7.8400
45.0000
15.3600
63.0000
72.0000

Tabla con un resumen total de las variables del circuito.

Solo funciona en Matlab. Si se va a usar Matlab, borre los simbolos % de los siguientes renglones.

```
% Valor_de_la_resistencia = Datos(:,1);  
% Nodo_Positivo = Datos(:,2);  
% Nodo_Negativo = Datos(:,3);  
% Nodo = [1:Ref]';  
% Voltaje_Nodo_y_Ref = Vnodos;  
% Voltaje_R = V_R;  
% Corriente_R = I_R;  
% Potencia_R = P_R;  
% Resumen1 = table(Valor_de_la_resistencia,Nodo_Positivo,Nodo_Negativo,Voltaje_R,Corriente  
_R,Potencia_R)  
% Resumen2 = table(Nodo,Voltaje_Nodo_y_Ref)
```