

```

%Méthode Des Eléments Finis Lineaires&1D
%Problème De Départ : -((1+t).y'(t))' = t ,Omega=]0,1[
%Avec des conditions Aux Limites : y(1)+y'(1)=0 , y(0)=0
%a(y,v)=int((1+t).y'.v') + 2y(1).v(1)
%l(v)=int(t.v)
clc;
clear all;
close all;
% discrétisation
a=0;
b=1;
h=1/4;
interval = a:h:b; %Le Maillage
n=length(interval); %nbr Des Nœuds
N=n-1; %Dimension De L'espace V_h
syms t % Définition Des Variables Et Fonctions
f = @(t) t; %Fonction f(t)
g = @(t) 1+t; %Fonction g(t)
% Définition Les Fonctions De Base
phi_i = (t-h*(i-1))/h; %ph(i) si t(i-1) <= t <= t(i)
phi_ii = (h*(i+1)-t)/h; %ph(i) si t(i) <= t <= t(i+1)
phi_i_minus_1 = (h*i-t)/h; %ph(i-1) si t(i-1) <= t <= t(i)
phi_i_plus_1 = (t-h*i)/h; %ph(i+1) si t(i) <= t <= t(i+1)
phi_0 = (h-t)/h; %ph(0) si t(0) <= t <= t(1)
phi_n_plus_1 = (t-(1-h))/h; %ph(n+1) si t(n) <= t <= t(n+1)
%Creation De La Matrice "A" Et Le Vecteur "B"
A = zeros(N,N);
B = zeros(N,1);
A(N,N)=int(g*diff(phi_n_plus_1,t)^2,[1-h 1])+2;
A_i_ii=int(diff(phi_i,t)^2,[h*(i-1) h*i])+int(diff(phi_ii,t)^2,[h*i
h*(i+1)]);
A_i_minus_1=int(diff(phi_i,t)*diff(phi_i_minus_1,t),[h*(i-1) h*i]);
%calcul diagonal Principale De La Matrice "A"
for i=1:N-1
t(1)=0;
t(i+1)=t(i)+h;
A(i,i) = h*(g(t(i+1))+g(h*(i-1)))*A_i_ii;
end
%calcul diagonal Inf De La Matrice "A"
for i=2:N
A(i,i-1) = (h/2*(g(t(i))+g(h*(i-1))))*A_i_minus_1;
end
%calcul diagonal Supp De La Matrice "A"
for i=1:N-1
A(i,i+1) = A(i+1,i); %Puisque A est symétrique
end
%calcul Vecteur B
B_i = int(phi_i,[h*(i-1) h*i]) + int(phi_ii,[h*i h*(i+1)]);
for i=1:N
t(1)=0;
t(i+1)=t(i)+h;
B(i) = (h*(f(t(i+1))+f(h*(i-1))))*B_i;
end
%Affichage La Matrice A Et Le Vecteur B
A=A
B=B
%Résoudre Le Système D'Equations A.Y=B
Y=A\B

```

```

%Méthode Des Eléments Finis Lineaires&1D
%Problème d'EDO : -y''(t) + alpha*y(t) = f(t) ,Omega=]0,1[
%Avec Les Conditions Aux Limites : y'(0)=y(0) , y'(1)=0
%a(y,v) = int(y'.v') + alpha*int(y.v) + y(0).v(0)
%l(v) = int(f.v)
clc;
clear all;
close all;
% discrétisation
a=0;
b=1;
h=1/3;
interval = a:h:b; %Le Maillage
N=length(interval); %nbr De Nœuds = Dimension De L'espace V_h
syms t % Définition Des Variables
alpha = 1; %constant
f = @(t) 1; %Fonction f(t)
% Définition Les Fonctions De Base
phi_i = (t-h*(i-1))/h;%ph(i) si t(i-1) <= t <= t(i)
phi_ii = (h*(i+1)-t)/h;%ph(i) si t(i) <= t <= t(i+1)
phi_i_minus_1 = (h*i-t)/h; %ph(i-1) si t(i-1) <= t <= t(i)
phi_i_plus_1 = (t-h*i)/h; %ph(i+1) si t(i) <= t <= t(i+1)
phi_0 = (h-t)/h; %ph(0) si t(0) <= t <= t(1)
phi_n_plus_1 = (t-(1-h))/h; %ph(n+1) si t(n) <= t <= t(n+1)
%Creation De La Matrice "A" Et Le Vecteur "B"
A = zeros(N,N);
B = zeros(N,1);
A(1,1)=int(diff(phi_0,t)^2+alpha*(phi_0)^2,[0 h])+1;
A(N,N)=int(diff(phi_n_plus_1,t)^2+alpha*(phi_n_plus_1)^2,[1-h 1]);
A_i_i=int(diff(phi_i,t)^2+alpha*(phi_i)^2,[h*(i-1)
h*i])+int(diff(phi_ii,t)^2+alpha*(phi_ii)^2,[h*i h*(i+1)]);
A_i_minus_1=int(diff(phi_i,t)*diff(phi_i_minus_1,t)+alpha*(phi_i)*(phi_i_minu
s_1),
[h*(i-1) h*i]);
A_i_plus_1 = A_i_minus_1;%Puisque A est symétrique
%calcul diagonal Principale De La Matrice "A"
for i=2:N-1
A(i,i) = A_i_i;
end
%calcul diagonal Inf De La Matrice "A"
for i=2:N
A(i,i-1) = A_i_minus_1;
end
%calcul diagonal Supp De La Matrice "A"
for i=1:N-1
A(i,i+1) = A_i_plus_1;
end
%calcul Vecteur B
B_i = int(phi_i,[h*(i-1) h*i]) + int(phi_ii,[h*i h*(i+1)]);
for i=1:N
t(1)=0;
t(i+1)=t(i)+h;
B(i) = (h*(f(t(i+1))+f(h*(i-1))))*B_i;
end
%Affichage La Matrice A Et Le Vecteur B
A=A
B=B
Y=A\B %Résoudre Le Système D'Equations A.Y=B

```