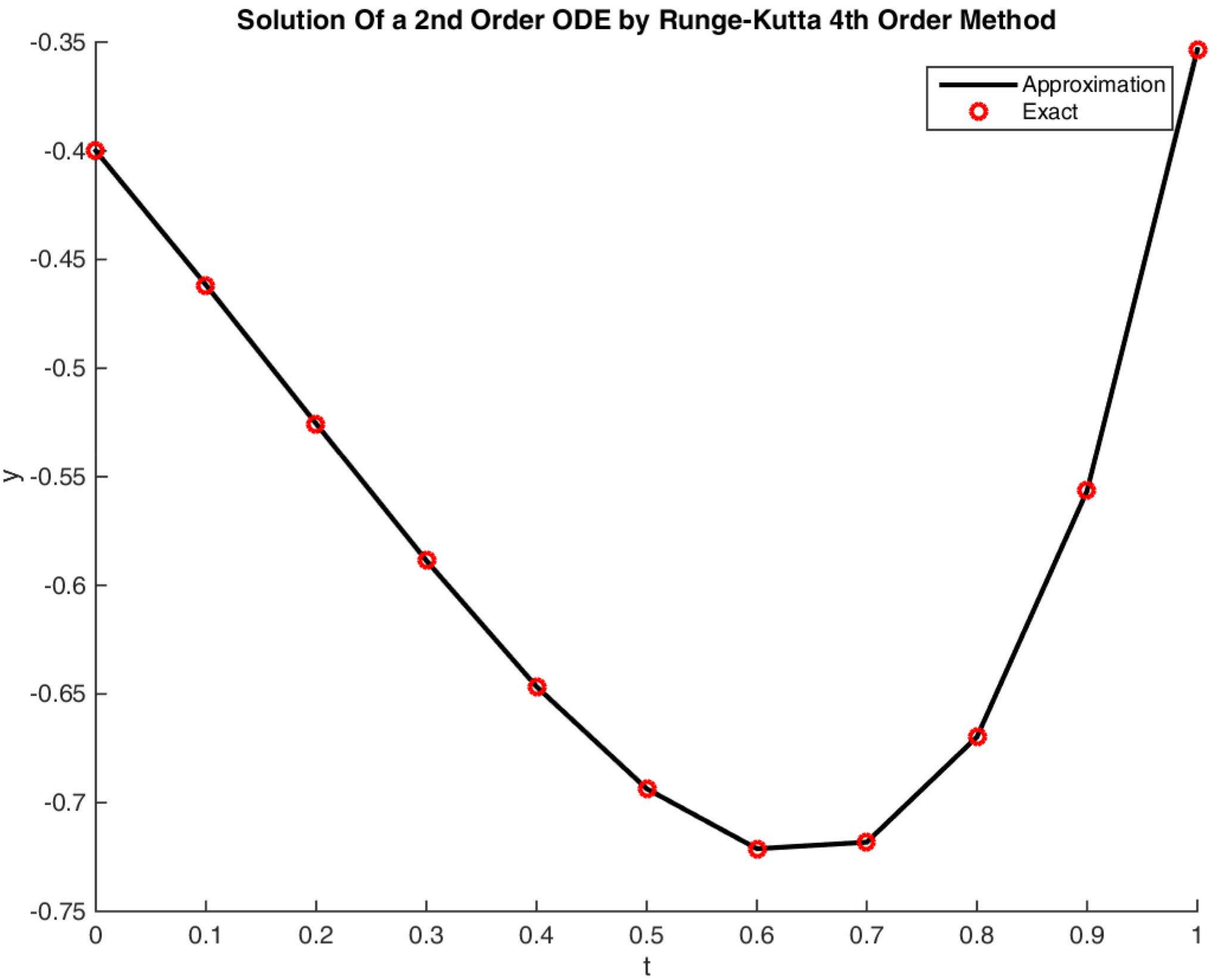
ZEHROURI TAHAR G2

TP Solution Of a 2nd Order ODE by Runge-Kutta 4th Order Method

```
f1     = @(t,y,z) z;
f2     = @(t,y,z) sin(t)*exp(2*t) - 2*y + 2*z;
t(1)   = 0;
z(1)   = -0.6;
y(1)   = -0.4;
h      = 0.1;
tfinal = 1;
m      = (tfinal-t(1))/h;

for j=1:m
   k11    = h*feval( f1 , t(j)     , y(j)       , z(j)      );
   k21    = h*feval( f2 , t(j)     , y(j)       , z(j)      );
   k12    = h*feval( f1 , t(j)+h/2 , y(j)+k11/2 , z(j)+k21/2);
   k22    = h*feval( f2 , t(j)+h/2 , y(j)+k11/2 , z(j)+k21/2);
   k13    = h*feval( f1 , t(j)+h/2 , y(j)+k12/2 , z(j)+k22/2);
   k23    = h*feval( f2 , t(j)+h/2 , y(j)+k12/2 , z(j)+k22/2);
   k14    = h*feval( f1 , t(j)+h   , y(j)+k13   , z(j)+k23  );
   k24    = h*feval( f2 , t(j)+h   , y(j)+k13   , z(j)+k23  );
   y(j+1) = y(j) + (k11 + 2*k12 + 2*k13 + k14)/6;
   z(j+1) = z(j) + (k21 + 2*k22 + 2*k23 + k24)/6;
   t(j+1) = t(j) + h;
end

fprintf('\n %.7f',y);
fprintf('\n %.7f',z);

                   %exact solution Of the ODE

fun = @(t,y)[y(2);sin(t)*exp(2*t) - 2*y(1) + 2*y(2)];
tspan = 0:h:tfinal;
[T,Y] =
ode15s(fun,tspan,[y(1),z(1)],odeset('RelTol',1e-8,'AbsTol',1e-8));
fprintf('\n %.7f',Y);

                   %ploting

title('Solution Of a 2nd Order ODE by Runge-Kutta 4th Order
Method');
xlabel('t');
ylabel('y');
hold on
plot(t,y,'-','LineWidth',2,'Color',[0 0 0]);
plot(T,Y,'o','LineWidth',2,'Color',[1 0 0]);
legend('Approximation','Exact');
```

Time	Exact value of y	Approximate value of y	Exact value of y'	Approximate value of y'
0,00	-0.4000000	-0.4000000	-0.6000000	-0.6000000
0,10	-0.4617330	-0.4617333	-0.6316310	-0.6316312
0,20	-0.5255590	-0.5255599	-0.6401486	-0.6401489
0,30	-0.5886000	-0.5886014	-0.6136635	-0.6136638
0,40	-0.6466102	-0.6466123	-0.5365821	-0.5365820
0,50	-0.6935639	-0.6935667	-0.3887389	-0.3887381
0,60	-0.7211484	-0.7211519	-0.1443830	-0.1443809
0,70	-0.7181487	-0.7181530	0.2289927	0.2289970
0,80	-0.6697066	-0.6697113	0.7719841	0.7719918
0,90	-0.5564379	-0.5564429	1.5347689	1.5347815
1,00	-0.3533941	-0.3533989	2.5787469	2.5787663

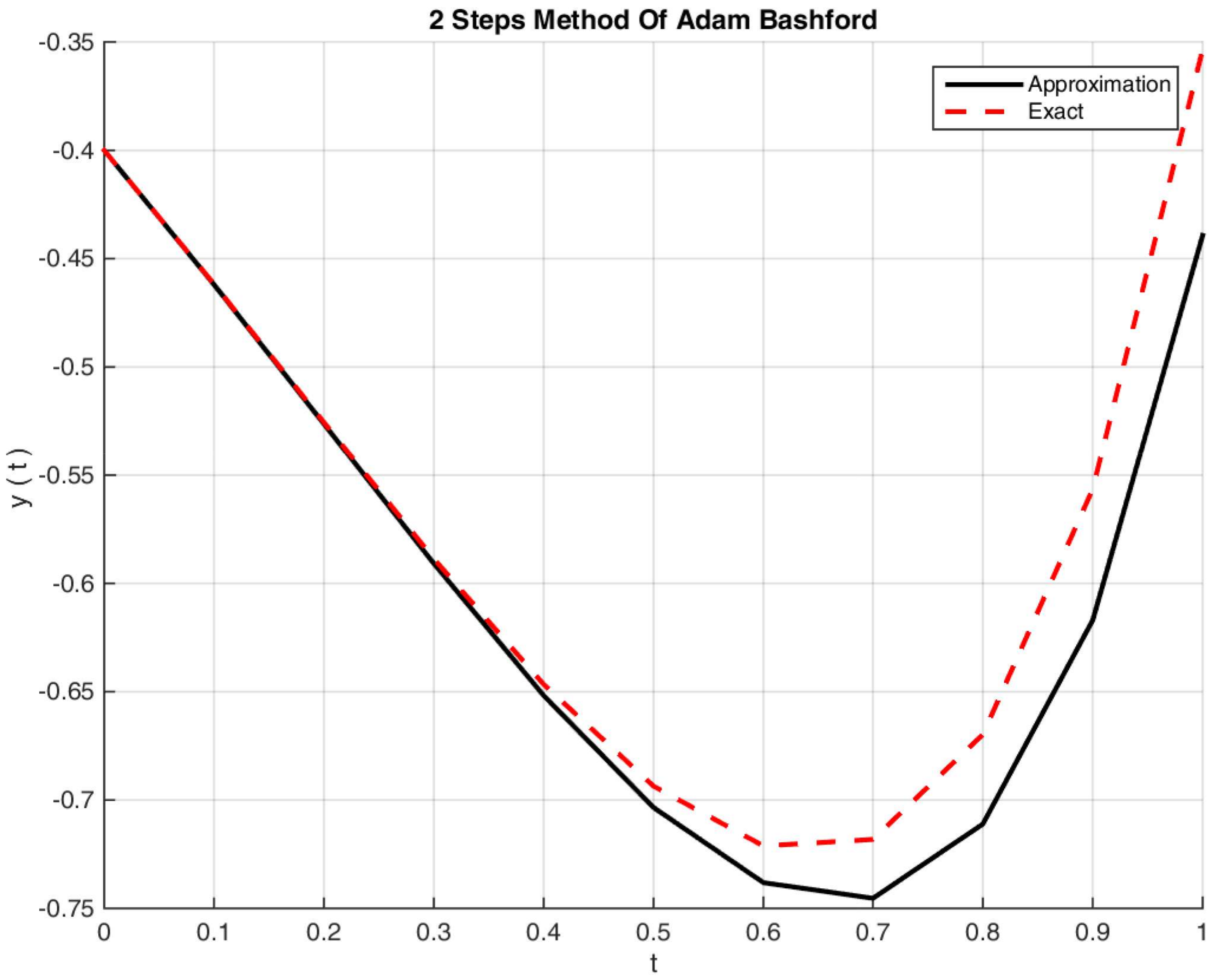


ZEHROURI TAHAR G2

TP Solution Of a 2nd Order ODE by 2 Steps Method Of Adam Bashford

```
f1     = @(t,y,z) z;
f2     = @(t,y,z) sin(t)*exp(2*t) - 2*y + 2*z;
t(1)   = 0;
z(1)   = -0.6;
y(1)   = -0.4;
h      = 0.1;
tfinal = 1;
m      = (tfinal-t(1))/h;

                  %modified euler method
for i=1:2
k11=h*feval(f1,t(i),y(i),z(i));
k21=h*feval(f2,t(i),y(i),z(i));

k12=h*feval(f1,t(i)+h,y(i)+k11,z(i)+k21);
k22=h*feval(f2,t(i)+h,y(i)+k11,z(i)+k21);

y(i+1)=y(i)+1/2*(k11+k12);
z(i+1)=z(i)+1/2*(k21+k22);
t(i+1)=t(i)+h;
end

                  % 2 Steps Method Of Adam Bashford
for i=3:m
y(i+1)=y(i)+(3/2)*h*feval(f1,t(i),y(i),z(i))-(1/2)*h*feval(f1,t(i-
1),y(i-1),z(i-1));
z(i+1)=z(i)+(3/2)*h*feval(f2,t(i),y(i),z(i))-(1/2)*h*feval(f2,t(i-
1),y(i-1),z(i-1));
t(i+1)=t(i)+h;
end
disp(t);
fprintf('\n %.7f',y);
fprintf('\n %.7f',z);

                  %exact solution Of the ODE

fun = @(t,y)[y(2);sin(t)*exp(2*t) - 2*y(1) + 2*y(2)];
tspan = 0:h:tfinal;
[T,Y] =
ode15s(fun,tspan,[y(1),z(1)],odeset('RelTol',1e-8,'AbsTol',1e-8));
%fprintf('\n %.7f',Y);

                  %ploting

title('2 Steps Method Of Adam Bashford');
xlabel('t');
ylabel('y ( t ) ');
hold on
plot(t,y,'-','LineWidth',2,'Color',[0 0 0]);
plot(T,Y,'--','LineWidth',2,'Color',[1 0 0]);
legend('Approximation','Exact');
```

Time	Exact value of y	Approximate value of y	Exact value of y'	Approximate value of y'
0,00	-0.4000000	-0.4000000	-0.6000000	-0.6000000
0,10	-0.4617330	-0.4620000	-0.6316310	-0.6319032
0,20	-0.5255590	-0.5262797	-0.6401486	-0.6408276
0,30	-0.5886000	-0.5908086	-0.6136635	-0.6198416
0,40	-0.6466102	-0.6517435	-0.5365821	-0.5511447
0,50	-0.6935639	-0.7034231	-0.3887389	-0.4149854
0,60	-0.7211484	-0.7381137	-0.1443830	-0.1863653
0,70	-0.7181487	-0.7453192	0.2289927	0.1663567
0,80	-0.6697066	-0.7110475	0.7719841	0.6828154
0,90	-0.5564379	-0.6169430	1.5347689	1.4121482
1,00	-0.3533941	-0.4392615	2.5787469	2.4146627



```
%TP methode de Tir pour les ODE lineaires
%(problemes aux limites: Dirichlet)

          %methode de Tir
f1=@(t,y,z) z;
f2=@(t,y,z) -4*z + 4*y;
y(1)=1;
z(1)=0;

f3=@(t,v,w) w;
f4=@(t,v,w) -4*w + 4*v;
v(1)=0;
w(1)=1;
t(1)=0;
tfinal=2;
h=0.2;
m=(tfinal-t(1))/h;
         %Euler (R-K order 1)
for j=1:(m-1)
    y(j+1) = y(j) + h*feval(f1,t(j),y(j),z(j));
    z(j+1) = z(j) + h*feval(f2,t(j),y(j),z(j));
    v(j+1) = v(j) + h*feval(f3,t(j),v(j),w(j));
    w(j+1) = w(j) + h*feval(f4,t(j),v(j),w(j));
    t(j+1) = t(j) + h;
end
disp(y);
disp(z);
disp(v);
disp(w);
```

❖ Calcule De Distribution De Température Pour L'équation De Poisson Par La Méthode De Difference Finie Centré Du Seconde Ordre

$$\begin{cases} -y'' = 1 \\ y(x_1) = 0 \qquad\qquad x_1 = 0 \ , x_8 = 1 \\ y(x_8) = 0 \\ h = dx = 1/7 \\ t = 0{:}dx{:}1 \\ n = \text{length}(x) = 8 \end{cases}$$

-(y(j-1) - 2*y(j) + y(j+1))/dx^2 = f(x(j),y(j))

-(y(j-1) - 2*y(j) + y(j+1))/dx^2 = 1

-y(j-1) + 2y(j) - y(j+1) = dx^2

T.q: j = $\overline{1{:}n}$-1

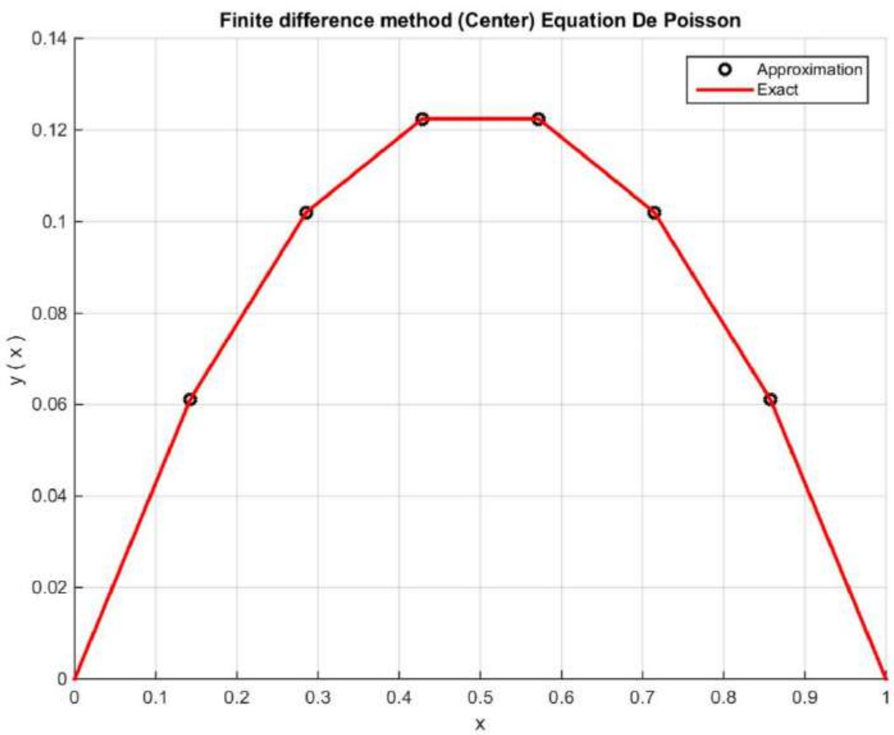
❖ On peut écrire le système précédent sous forme de matrice comme suit :

$$\begin{cases} j=1 \\ j=2 \\ j=3 \\ j=4 \\ j=5 \\ j=6 \end{cases} \begin{pmatrix} 2 & -1 & 0 & 0 & 0 & 0 \\ -1 & 2 & -1 & 0 & 0 & 0 \\ 0 & -1 & 2 & -1 & 0 & 0 \\ 0 & 0 & -1 & 2 & -1 & 0 \\ 0 & 0 & 0 & -1 & 2 & -1 \\ 0 & 0 & 0 & 0 & -1 & 2 \end{pmatrix} . \begin{pmatrix} y(x_1) \\ y(x_2) \\ y(x_3) \\ y(x_4) \\ y(x_5) \\ y(x_6) \end{pmatrix} = \begin{pmatrix} h^2 \\ h^2 \\ h^2 \\ h^2 \\ h^2 \\ h^2 \end{pmatrix}$$

A . Y = B

❖ La solution du système est donnée par le vecteur Y T.q : Y = A\B

$$\begin{pmatrix} y(x_1) \\ y(x_2) \\ y(x_3) \\ y(x_4) \\ y(x_5) \\ y(x_6) \end{pmatrix} = \begin{pmatrix} 0.061224 \\ 0.102040 \\ 0.122448 \\ 0.122448 \\ 0.102040 \\ 0.061224 \end{pmatrix}$$



```matlab
%ZEHROURI TAHAR G2
%Méthode Des Différences Finies(Centré Du 2nd Ordre)
%Equation De Poisson : -y''=1
%Avec Les Conditions Aux Limites De Dirichlet: y(0)=y(1)=0
% -y'' = 1
% -(y(j-1) - 2*y(j) + y(j+1))/h^2 = f(x(j),y(j))
% -(y(j-1) - 2*y(j) + y(j+1))/h^2 = 1
% -y(j-1) + 2*y(j) - y(j+1) = h^2
% j=1:n-2
clc;
clear all;
close all;
      % discretisation
a=0;
b=1;
h=1/7;
interval = a:h:b;
n=length(interval);
      %Creation De La Matrice "A" Et Vecteur "B"
A = zeros(n-2,n-2);
B = zeros(n-2,1);
      %calcul diagonal Principale De La Matrice "A"
for j=1:n-2
    A(j,j)=[2];
end
      %calcul diagonal Inf De La Matrice "A"
for j = 2:n-2
  A(j,j-1) = [-1];
end
      %calcul diagonal Supp De La Matrice "A"
for j=1:n-3
    A(j,j+1)=[-1];
end
      %calcul Vecteur B
for j = 1:n-2
  B(j)   = h^2;
end
%Affichage De Matrice A Et Vecteur B
A=A
B=B
     %Resoudre Systeme D'Equations A.Y=B
Y = A\B
%ploting and comparaison between exact and Approximation solution
X=(a+h):h:(b-h); %Interval De Plot Solution Numerique
y_reel= @ (x) (x-x.^2)/2; %Solution Exacte
hold on;
plot(X,Y,'O','LineWidth',2,'Color',[0 0 0]);
plot(interval,y_reel(interval),'-','LineWidth',2,'Color',[1 0 0]);
legend('Approximation','Exact');
title('Finite difference method (Center) Equation De Poisson');
xlabel('x');
ylabel('y ( x ) ');
grid on;
```

❖ **Les Valeurs Approximations De y Aux Points t(2), t(3) et t(4) Par La Méthode Des Differences Finies Centré Du Seconde Ordre**

$$\begin{cases} y'' = (1 - t/5)*y + t \\ y(t_1) = 2 \qquad\qquad t_1 = 1 \;, t_5 = 3 \\ y(t_5) = -1 \\ h = dt = 1/2 \\ t = 1:h:3 \\ n = length(t) = 5 \end{cases}$$

(y(j-1) - 2*y(j) + y(j+1))/h^2 = f(x(j),y(j))

(y(j-1) - 2*y(j) + y(j+1))/h^2 = (1-t(j)/5)*y(j)+t(j)

-y(j-1) + (2+h^2*(1-t(j)/5)*y(j) - y(j+1) = -h^2*t(j)

-y(j-1) + s(j)*y(j) - y(j+1) = -h^2*t(j)

T.q: s(j)= 2+h^2*(1-t(j)/5 , $j = \overline{2:4}$

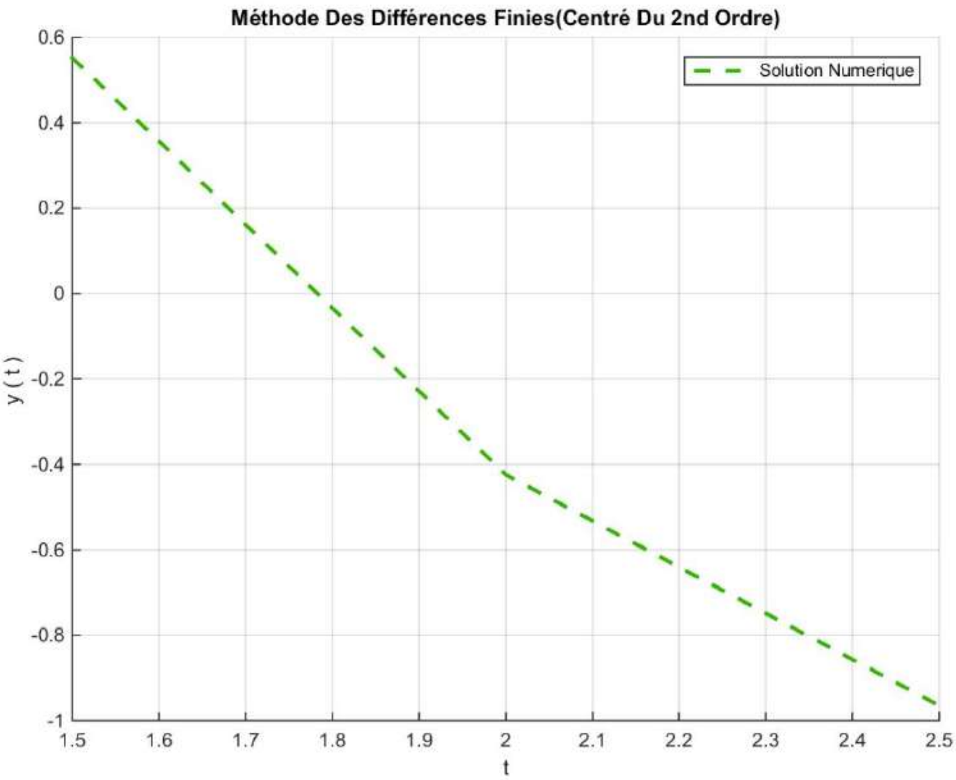
❖ **On peut écrire le système précédent sous forme de matrice comme suit :**

$$\begin{Bmatrix} j=2 \\ j=3 \\ j=4 \end{Bmatrix} \begin{pmatrix} s(t_2) & -1 & 0 \\ -1 & s(t_3) & -1 \\ 0 & -1 & s(t_4) \end{pmatrix} . \begin{pmatrix} y(t_2) \\ y(t_3) \\ y(t_4) \end{pmatrix} = \begin{pmatrix} -(h\wedge 2)*t(t_2) + y(t_1) \\ -(h\wedge 2)*t(t_3) \\ -(h\wedge 2)*t(t_4) + y(t_5) \end{pmatrix}$$

A . Y = B

❖ **La solution du système est donnée par le vecteur Y T.q : Y = A\B**

$$\begin{pmatrix} y(t_2) \\ y(t_3) \\ y(t_4) \end{pmatrix} = \begin{pmatrix} 0.552013 \\ -0.424370 \\ -0.964409 \end{pmatrix}$$



```matlab
%ZEHROURI TAHAR G2
%Méthode Des Différences Finies(Centré Du 2nd Ordre)
%L'Equation Suivante : y'' = (1 - t/5)*y + t
%Avec Les Conditions Aux Limites : y(1)=2, y(3)=-1
% (y(j-1) - 2*y(j) + y(j+1))/h^2 = f(t(j),y(j))
% (y(j-1) - 2*y(j) + y(j+1))/h^2 = (1 - t(j)/5)*y(j) + t(j)
% -y(j-1) + (2 + h^2*(1 - t(j)/5))*y(j) - y(j+1) = -h^2*t(j)
% j=1:n-2
clc;
clear all;
close all;
      % discretisation
a=1;
b=3;
h=1/2;
t(1)=a;
y(a)=2;
y(b)=-1;
interval = a:h:b;
n=length(interval);
      %Creation De La Matrice "A" Et Vecteur "B"
A = zeros(n-2,n-2);
B = zeros(n-2,1);
      %calcul diagonal Principale De La Matrice "A"
for j=1:n-2
    t(j+1) = t(j)+h;
    A(j,j) = [(2 + h^2*(1 - t(j+1)/5))];
end
      %calcul diagonal Inf De La Matrice "A"
for j=2:n-2
    A(j,j-1) = [ -1 ];
end
      %calcul diagonal Supp De La Matrice "A"
for j=1:n-3
    A(j,j+1) = [ -1 ];
end
      %calcul Vecteur B
B(1)   = -(h^2)*t(2)   + y(a); %Avec Les Conditions Aux
B(n-2) = -(h^2)*t(n-1) + y(b); %Limites : y(a)=2, y(b)=-1
for  j = 2:n-3
  B(j) = -(h^2)*t(j+1);
end
%Affichage De Matrice A Et Vecteur B
A=A
B=B
     %Resoudre Systeme D'Equations A.Y=B
Y = A\B
%ploting and comparaison between exact and Approximation solution
X=(a+h):h:(b-h); %Interval De Plot Solution Numerique
hold on;
plot(X,Y,'--','LineWidth',2,'Color',[0.2 0.7 0]);
legend('Solution Numerique');
title('Méthode Des Différences Finies(Centré Du 2nd Ordre)');
xlabel('x');
ylabel('y ( x ) ');
grid on;
```

```matlab
%ZEHROURI TAHAR G2
%TP Methode Des Elements Finis Lineaires De Lagrange(1st Order FEM)
%Equation De Poisson : -y''=1
%Avec Les Conditions Aux Limites De Dirichlet: y(0)=y(1)=0
clc;
clear all;
close all;
syms t   % Definition Des Variables
      % discretisation
a=0;
b=1;
h=1/7;
interval = a:h:b; %Le Maillage
N=length(interval); %nbr De Noeuds
n=(b-a)/h; %nbr Des Elements
p=1;  %Fonction p(t)
q=0;  %Fonction q(t)
f=1;  %Fonction f(t)
ph1=(h*i-t)/h;    %ph(i-1) si   t(i-1) <= t <= t(i)
ph2=(t-h*i)/h;    %ph(i+1) si   t(i)   <= t <= t(i+1)
ph3=(t-h*(i-1))/h;%ph(i)   si   t(i-1) <= t <= t(i)
ph4=(h*(i+1)-t)/h;%ph(i)   si   t(i)   <= t <= t(i+1)
     %Creation De La Matrice De Rigidite "A" Et Vecteur "B"
A = zeros(N-2,N-2);
B = zeros(N-2 ,1);
     %calcul diagonal Inf De La Matrice "A"
for i=2:N-2
A(i,i-1)=int((p*diff(ph1,t)*diff(ph3,t)+q*ph1*ph3),[h*(i-1) h*i]);
end
     %calcul diagonal Principale De La Matrice "A"
for i=1:N-2
A(i,i)=int((p*diff(ph3,t)^2+q*(ph3)^2),[h*(i-1)
h*i])+int((p*diff(ph2,t)^2+q*(ph2)^2),[h*i h*(i+1)]);
end
     %calcul diagonal Supp De La Matrice "A"
for i=1:N-3
A(i,i+1)=int((p*diff(ph2,t)*diff(ph4,t)+q*ph2*ph4),[h*i h*(i+1)]);
end
 %calcul Vecteur B
for i=1:length(interval)-2
B(i)=int((f*ph3),[h*(i-1) h*i])+int((f*ph4),[h*i h*(i+1)]);
end
%Affichage De Matrice A Et Vecteur B
A=A
B=B
Y=A\B     %Resoudre Systeme D'Equations A.Y=B
%ploting and comparaison between exact and Approximation solution
T=(a+h):h:(b-h); %Interval De Plot Solution Numerique
y_reel= @ (t) (t-t.^2)/2; %Solution Exacte
hold on;
plot(T,Y,'O','LineWidth',2,'Color',[0 0 0]);
plot(interval,y_reel(interval),'-','LineWidth',2,'Color',[1 0 0]);
legend('Approximation','Exact');
title('Methode Des Elements Finis Lineaires De Lagrange(1st Order FEM)');
xlabel('t');
ylabel('y ( t ) ');
grid on;
```

A =

14	-7	0	0	0	0
-7	14	-7	0	0	0
0	-7	14	-7	0	0
0	0	-7	14	-7	0
0	0	0	-7	14	-7
0	0	0	0	-7	14

B =

0.1429
0.1429
0.1429
0.1429
0.1429
0.1429

Y =

0.061224
0.102040
0.122448
0.122448
0.102040
0.061224