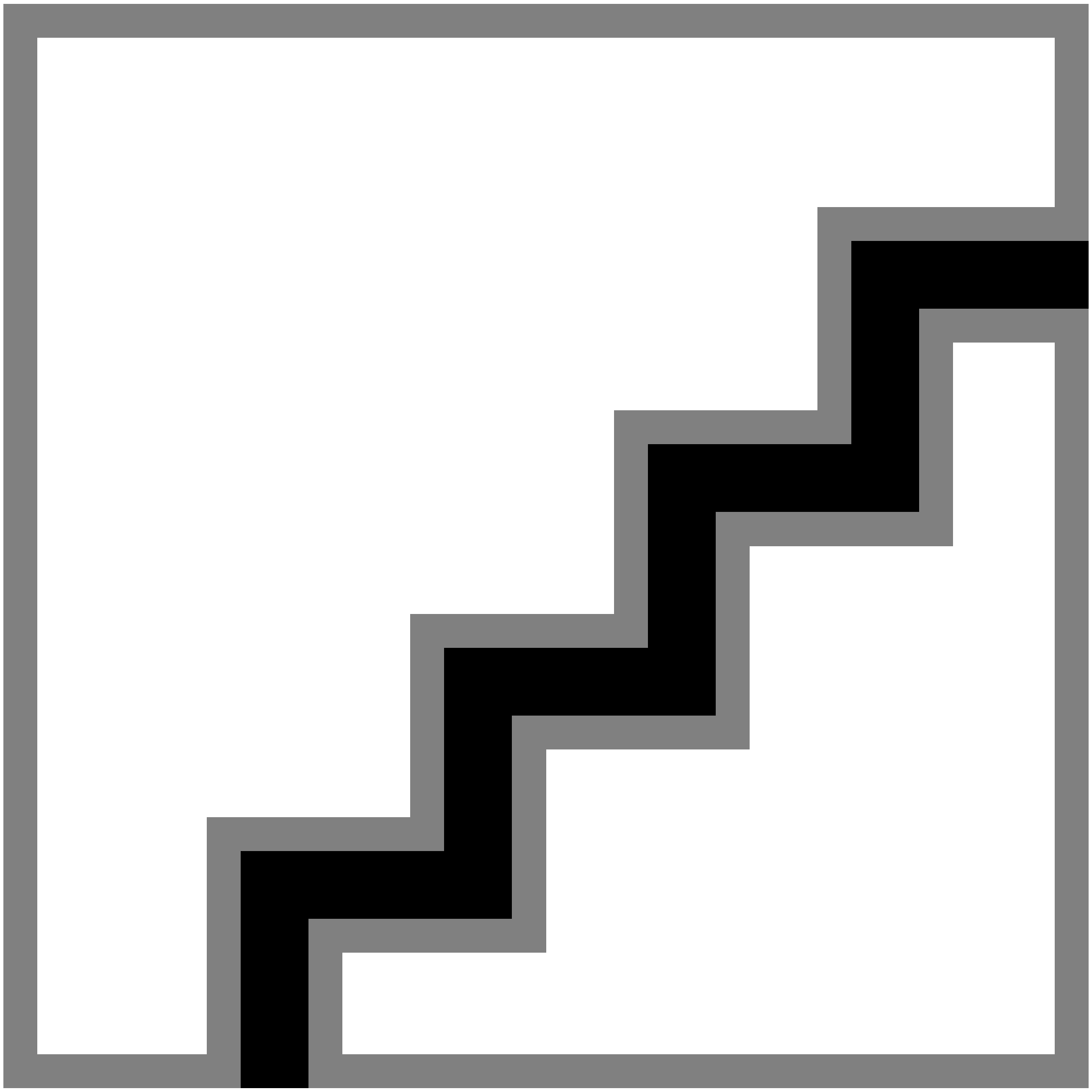
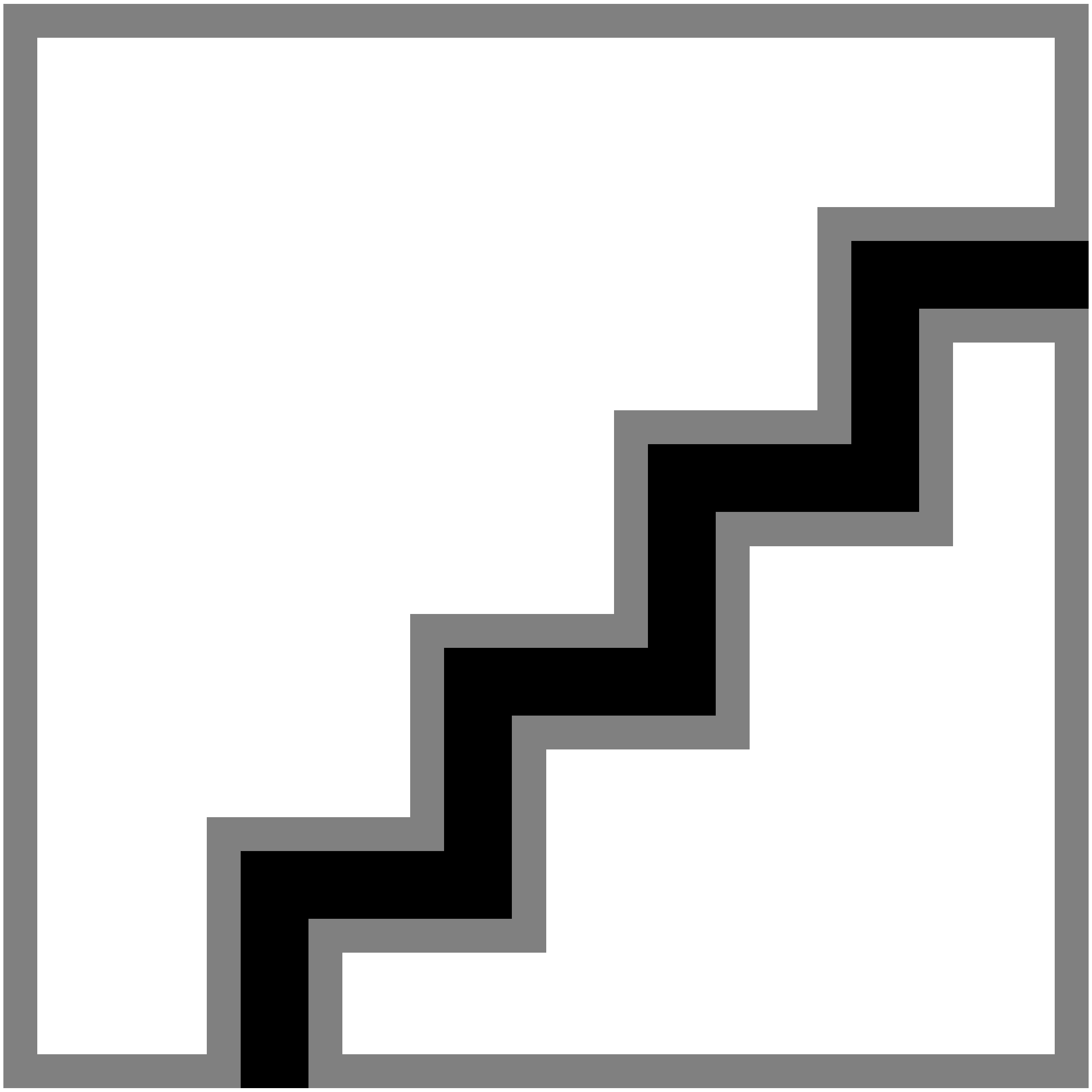
```
%Abhay Nanduri
%CH.EN.U4AIE21130
```

```
% Matlab Program 1: Solving the x'=-2x+t ODE by using the Mid point method
a=1.;
b=-2.;
c=1.;
tinit= 0.;
tmax=5.;
maxt = 3000;
dt = (tmax-tinit)/maxt;
x(1)=1.;
t(1)=tinit;
for j=1:maxt;
   x(j+1)=x(j)+dt*((b*x(j)+c*(j)*dt)/a);
   t(j+1)=tinit+j*dt;
end;
plot(t,x)
title('Euler Method')
xlabel('T')
ylabel('X(t)')
```



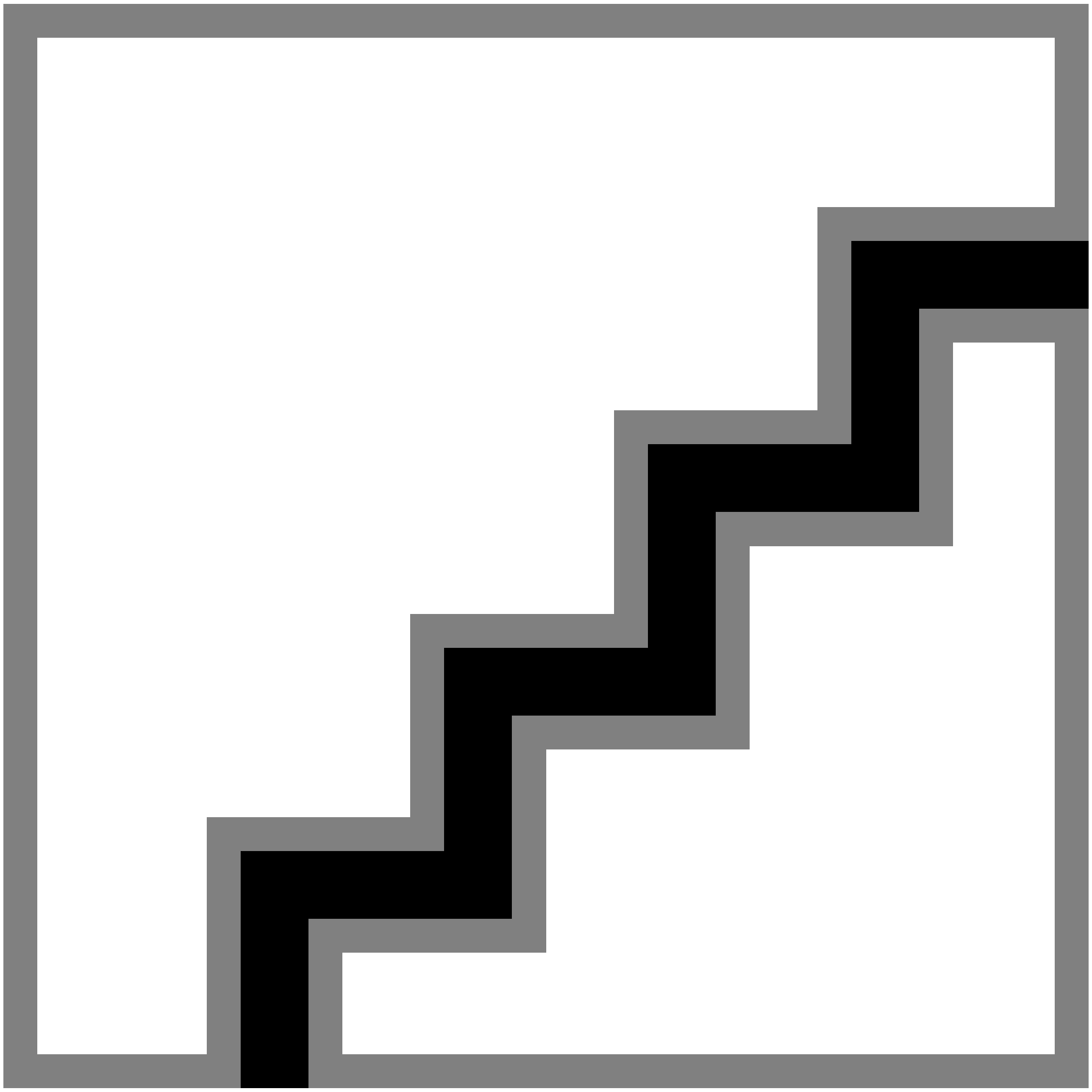
```
% Matlab Program 2: Solving the a x'=b x + c t ODE by using the
%midpoint method
a=1.;
b=-2.;
c=1.;
% Initial and Final Times
tinit= 0.;
tmax=5.;
% Number of Time Steps
maxt = 3000;
dt = (tmax-tinit)/maxt;
% Initial Condition
x(2)=1.;
x(1)=1.0-dt*((b*x(2)+c*(2)*dt)/a);
t(2)=tinit;
% Time Loop
for j=2:(maxt+1);
```

```
    x(j+1)=x(j-1)+2.*dt*((b*x(j)+c*(j)*dt)/a);
    t(j+1)=tinit+(j-1)*dt;
end;
plot(t,x)
title('Midpoint Method')
xlabel('T')
ylabel('X(t)')
```

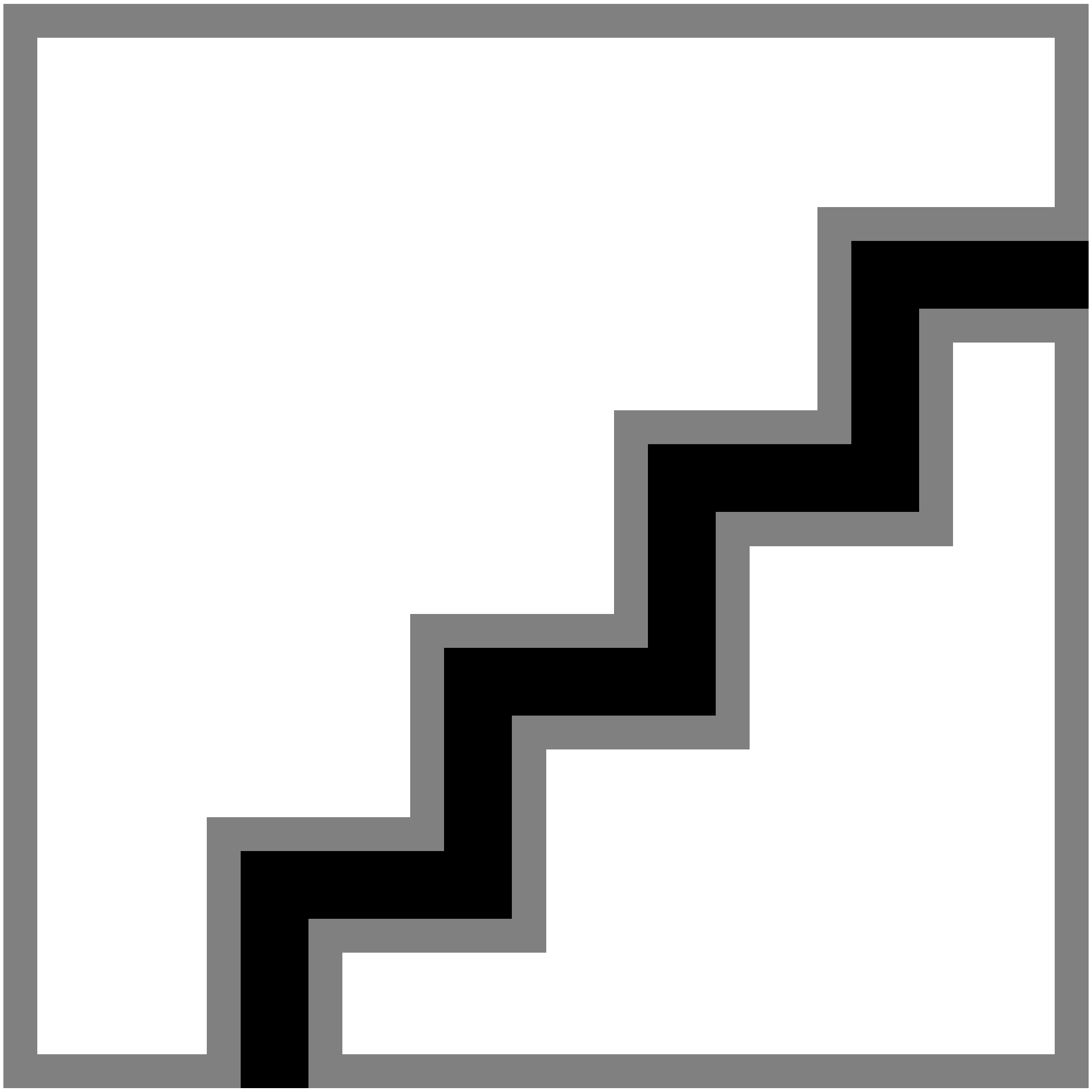


```
% Matlab Program 5: Heat Diffusion in one dimensional wire within the
% Explicit Method
clear;
% Parameters to define the heat equation and the range in space and time
L = 1.; % Length of the wire
T =1.; % Final time
% Parameters needed to solve the equation within the explicit method
maxk = 2500; % Number of time steps
dt = T/maxk;
n = 50; % Number of space steps
```

```
dx = L/n;
cond = 0.525; % Conductivity
b = 2.*cond*dt/(dx*dx); % Stability parameter (b=<1)
% Initial temperature of the wire: a sinus.
for i = 1:n+1
x(i) =(i-1)*dx;
u(i,1) =sin(pi*x(i));
end
% Temperature at the boundary (T=0)
for k=1:maxk+1
u(1,k) = 0.;
u(n+1,k) = 0.;
time(k) = (k-1)*dt;
end
% Implementation of the explicit method
for k=1:maxk % Time Loop
for i=2:n; % Space Loop
u(i,k+1) =u(i,k) + 0.5*b*(u(i-1,k)+u(i+1,k)-2.*u(i,k));
end
end
% Graphical representation of the temperature at different selected times
figure(1)
plot(x,u(:,1),'-',x,u(:,100),'-',x,u(:,300),'-',x,u(:,600),'-')
title('Temperature within the explicit method')
xlabel('X')
ylabel('T')
```

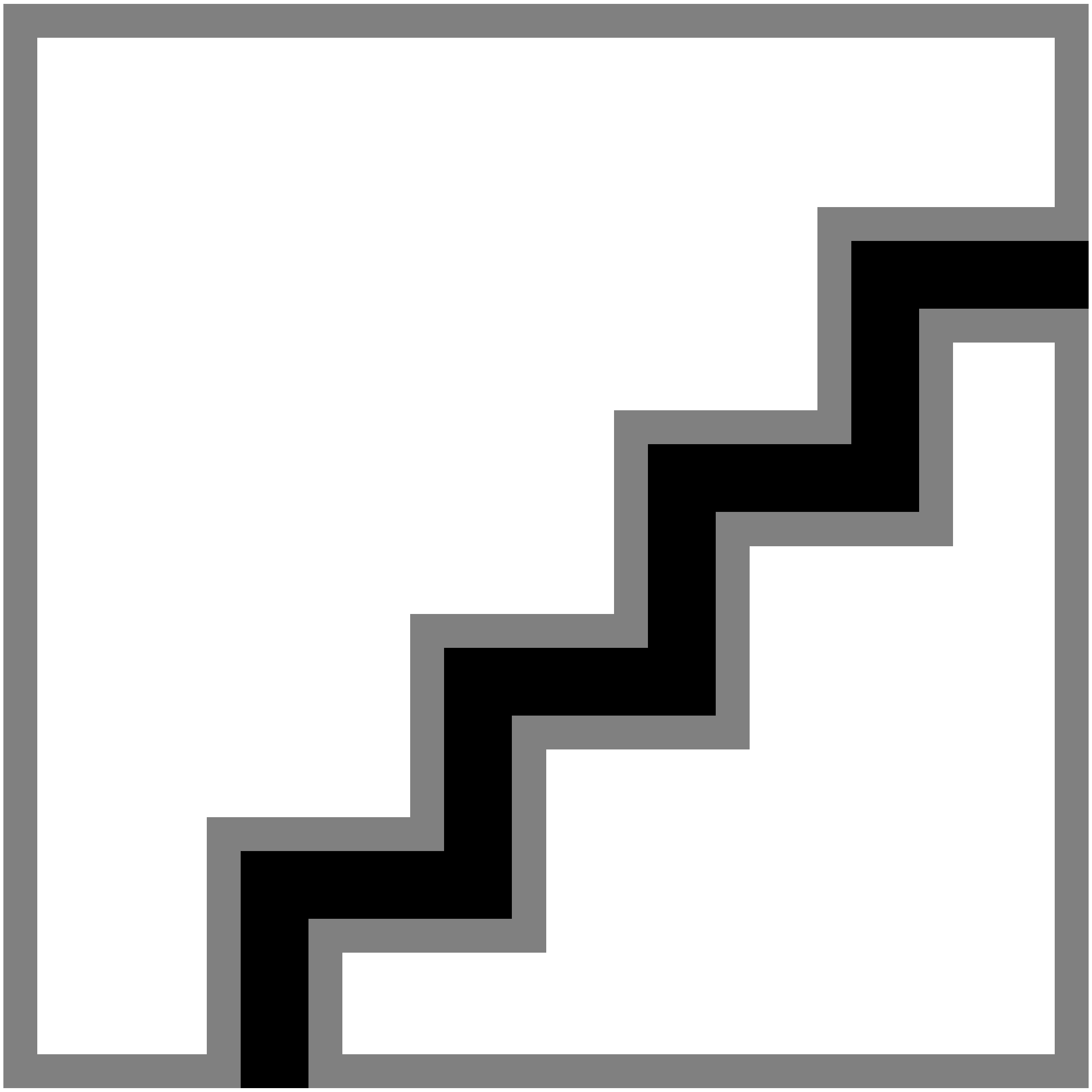


```
figure(2)
mesh(x,time,u')
title('Temperature within the explicit method')
xlabel('X')
ylabel('Temperature')
```

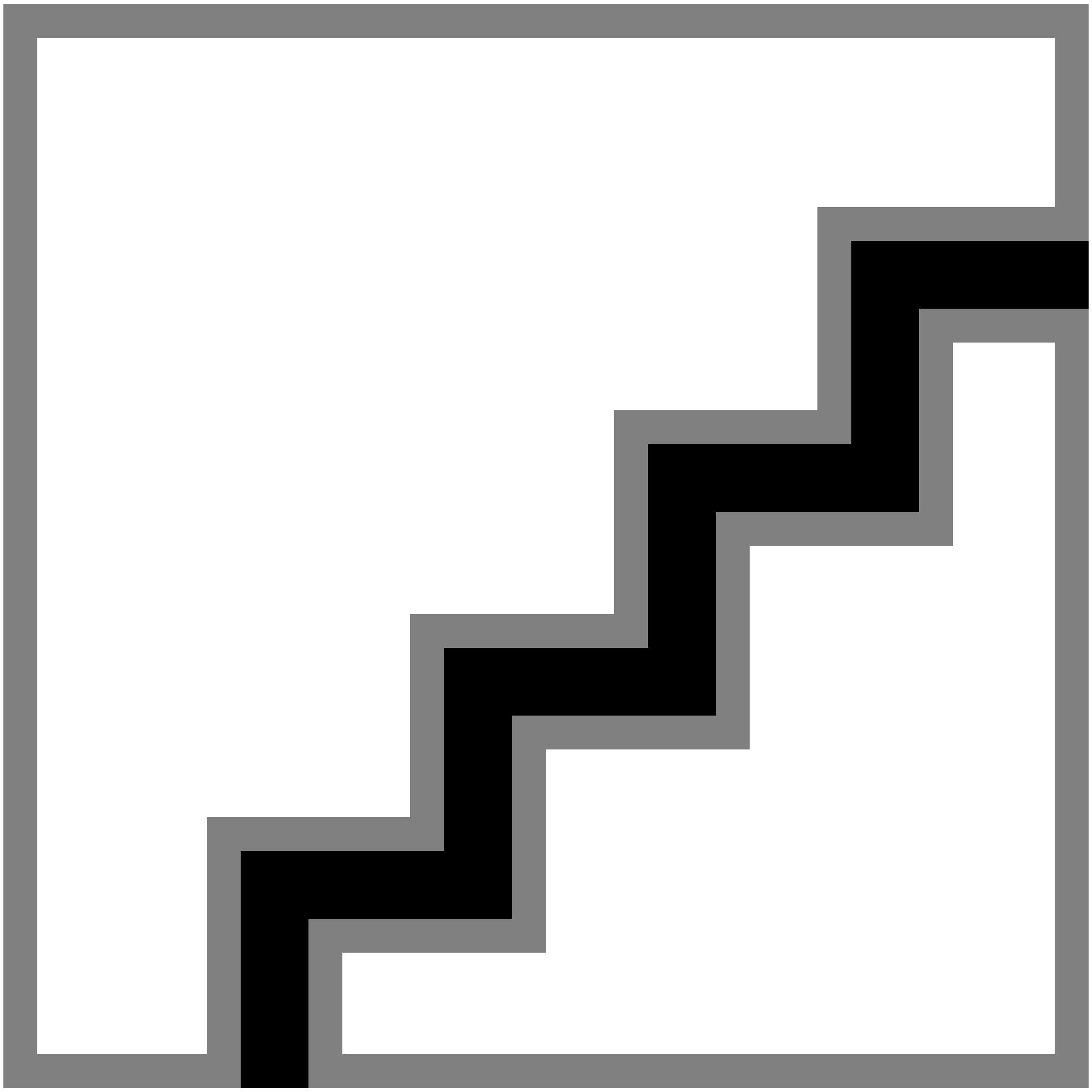


```
% Matlab Program 6: Heat Diffusion in one dimensional wire within the Fully
% Implicit Method
clear;
% Parameters to define the heat equation and the range in space and time
L = 1.; % Lenth of the wire
T =1.; % Final time
% Parameters needed to solve the equation within the fully implicit method
maxk = 2500; % Number of time steps
dt = T/maxk;
n = 50.; % Number of space steps
dx = L/n;
cond = 1./4.; % Conductivity
b = cond*dt/(dx*dx); % Parameter of the method
% Initial temperature of the wire: a sinus.
for i = 1:n+1
x(i) =(i-1)*dx;
u(i,1) =sin(pi*x(i));
```

```
end
% Temperature at the boundary (T=0)
for k=1:maxk+1
u(1,k) = 0.;
u(n+1,k) = 0.;
time(k) = (k-1)*dt;
end
aa(1:n-2)=-b;
bb(1:n-1)=1.+2.*b;
cc(1:n-2)=-b;
MM=inv(diag(bb,0)+diag(aa,-1)+diag(cc,1));
% Implementation of the implicit method
for k=2:maxk % Time Loop
uu=u(2:n,k-1);
u(2:n,k)=MM*uu;
end
% Graphical representation of the temperature at different selected times
figure(1)
plot(x,u(:,1),'-',x,u(:,100),'-',x,u(:,300),'-',x,u(:,600),'-')
title('Temperature within the fully implicit method')
xlabel('X')
ylabel('T')
```



```
figure(2)
mesh(x,time,u')
title('Temperature within the fully implicit method')
xlabel('X')
ylabel('Temperature')
```



```
clear variables
close all
% Backward euler method
a = 0; b = 0.5;
h = 0.002;
N = (b - a)/h;
M = 20;
n = zeros(1,N);
t = n;
n(1) = 2000;
t(1) = a;
for i=1:N
t(i + 1) = t(i) + h;
x = n(i);
% Newton's implicit method starts.
for j = 1:M
num = x + 0.800*x.^(3/2) *h - 10.*n (1) * (1 - exp(-3*t(i + 1))) *h - n(i) ;
```

```
denom = 1 + 0.800*1.5*x.^(1/2) *h;
xnew = x - num/ denom;
if abs((xnew - x)/x) < 0.0001
break
else
 x = xnew;
end
end
%Newton's method ends.
n(i + 1) = xnew;
end
figure(1)
plot(t,n,'linewidth',1.5,'color','b')
grid;
a = title('Backward Euler Method');
set(a,'fontsize',14);
a = ylabel('n');
set(a,'Fontsize',14);
a = xlabel('t(s)');
set(a,'Fontsize',14);
axis([0 0.5 0 2000])
```