

```

% This function Numerically solves the 1-D heat equation using the
% Crank-Nicolson scheme
% input
% f : function representing intial condition
% g0 and g1 : function represents boundary conditions
% tspan : time span over
% N : Number of time steps
% m : Number of points for the uniform spatial discretisation.
% Return
% u : Approximate solution at each time step (matrix)
% t : Vector containing all time steps
% x : Vector containing the spatial discretization.

function [u,t,x] = cnhteq(f,g0,g1,tspan,alp,N,m)

h = 1/(m+1);
k = tspan/N;
r = (alp*k)/(2*h^2);
U0 = zeros(1,m+2);
U = zeros(N+1,m+2);

%at to
t= zeros(N,1); t(1) = 0;
x= zeros(m+2,1);

for i = 1:m+2
x(i) = (i-1)*h;
U0(i) = f(x(i));
end

U(1,:) = U0;

%BC
gl = zeros(m,1); %left
gr = zeros(m,1); %Right

a1 = (1+2*r);
a2 = (1-2*r);
b = r;

% Using sparse library to transform A
A = sparse(toeplitz([a1 -b zeros(1, m-2)]));
B = sparse(toeplitz([a2 b zeros(1, m-2)]));

for i = 1:N
    t(i+1) = i*k;
    %BC
    gl(1) = g0(t(i+1));
    gr(1) = g0(t(i));
    gl(m) = g1(t(i+1));
    gr(m) = g1(t(i));
    %interior
    e = U0(2:m+1)';
    d = B*e + r*(gl + gr);
    %solving the linear system
    U1 = A\d;
    %Treating bounary values
    Un = [gl(1), U1', gl(m)];
    U(i+1,:) = Un;
    U0 = Un;
end

```

```
u = U;  
end
```

Not enough input arguments.

Error in cnhteq (line 17)
h = 1/(m+1);

Published with MATLAB® R2020a