

```

% The program uses BDF2 for the time integratorl instead of the trapezoidal
% Rule.

% 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.% This function Numerically solves the 1-D heat equation using the
% Crank-Nicolson scheme
% input

function [u,t,x] = BDF2(f,g0,g1,tspan,alp,N,m)

h = 1/(m+1);
k = tspan/N;
r = (2*alp*k)/(3*h^2);

%space and time steps
x = 0:h:1;
t = 0:k:tspan;

%Crank-Nicolson
u1 = cnhteq(f,g0,g1,tspan,alp,N,m);

U = zeros(m+2,m+2);
gl = zeros(m,1);

U0 = f(x);
U(1,:) = U0;
U1 = u1(2,:);
U(2,:) = U1;

%Matrix Coefficients
a= (1+2*r);
b = r;

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

for n = 3: N+1

    gl(1) = g0(t(n));
    gl(m) = g1(t(n));

    Un1 = U0(2:m+1)';
    Un2= U1(2:m+1)';

    % The Right Hand Side
    B = (4/3)*Un2 - (1/3)*Un1 + r*gl;

    % Solving the linear sysytem for the interior points

    Un = A\B;

    % Treating the boundary conditions
    Un = [gl(1), Un', gl(m)];
    U(n,:) = Un;

    %update solution
    U0 = U1;
    U1 = Un;

end
u = U;
end

```

Not enough input arguments.

Error in BDF2 (line 19)

h = 1/(m+1);

Published with MATLAB® R2020a