Fourier's Law of Heat Conduction.

Consider a uniform rod of length l with non-uniform temperature lying on the x-axis from $x=0$ to $x=L$. The rod has density ρ, specific heat c, thermal conductivity K_0, cross-sectional area A, all constant. Assumed, that the sides of rod are insulated so that no heat passes through. Only the ends of rod are exposed. One more assumption is made that there is no heat source within the rod. An arbitrary thin slice of width Δx is considered between x and $x+\Delta x$. The slice is so thin that temperature throughout the slice is $T(x,t)$. Thus,

$$\text{Heat energy of segment} = c \times \rho A \Delta x \times T$$
$$= c\rho A \Delta x \, T(x,t).$$

Conservation of energy.

$$\begin{matrix}\text{Change of heat energy} \\ \text{of segment in time } \Delta t\end{matrix} = \begin{matrix}\text{heat in from} \\ \text{left boundary}\end{matrix} - \begin{matrix}\text{heat out from} \\ \text{right boundary.}\end{matrix}$$

Fourier's law states,

$$\frac{\text{Rate of heat transfer}}{\text{Area}} = -K_0 \frac{\partial T}{\partial x}. \quad (1)$$

$$\Rightarrow c\rho A \Delta x \, T(x,t+\Delta t) - c\rho A \Delta x \, T(x,t)$$
$$= \Delta t A \left(-K_0 \frac{\partial T}{\partial x}\right)_x - \Delta t A \left(-K_0 \frac{\partial T}{\partial x}\right)_{x+\Delta x}$$

$$\Rightarrow \frac{T(x,t+\Delta t) - T(x,t)}{\Delta t} = \frac{K_0}{c\rho}\left(\frac{\left(\frac{\partial u}{\partial x}\right)_{x+\Delta x} - \left(\frac{\partial u}{\partial x}\right)_x}{\Delta x}\right)$$

lim $(\Delta t, \Delta x) \to 0$ gives 1-D heat equation.

$$\frac{\partial T}{\partial t} = K \frac{\partial^2 T}{\partial x^2} \quad -(2).$$

where $K = \frac{k_0}{c\rho}$ is called thermal diffusivity.

Energy flux

The energy flux density, J, is defined as the rate of heat transfer per unit area.

$$\Rightarrow \quad J = \frac{1}{A}\frac{dQ}{dt} \quad -(3)$$

From fourier's law, we had,

$$\frac{dQ}{dt} = -KA\frac{dT}{dx}. \quad -(4).$$

Using (3) & (4), we get,

$$J = -K\frac{dT}{dx}. \quad -(5).$$

We also know,

$$dQ_{net} = mc\,dT. \quad -(6).$$

$$\Rightarrow \frac{dQ}{dt} = mc\frac{dT}{dt} \quad -(7).$$

The assumptions made about uniform rod, states

$$V = A\,dx$$

& $$m = \rho A\,dx \qquad — ⑧.$$

Using ③, ⑥ & ⑦.,

$$\frac{dT}{dt} = \frac{1}{\rho c A\,dx}\frac{dQ_{net}}{dt} = \frac{1}{\rho c\,dx} J_{net}. \qquad —⑨.$$

$$J_{net} = \frac{1}{A}\frac{dQ_{net}}{dt}.$$

The energy flux density we obtain is due to conduction of energy along the rod, internal heating of rod (although we neglected it in assumption), and also the transfer of heat from and to the environment. (neglected in ass)

$$\Rightarrow J_{net} = J_{cond} + J_{int} + J_{env}.$$

$$\Rightarrow J_{net} = J_{cond} = -K\frac{dT}{dx}.$$

$$\because \frac{dT}{dt} = \frac{1}{\rho c\,dx} J_{net}.$$

Problem

Rod of length = L.

Divided into N cylindrical elements.

Width of each element, $\Delta x = L/N$.

Element is small, assumption: all properties of element are uniform throughout the volume of element.

Radius of n-th element = $R(n)$
Thermal conductivity = $K(n)$
Specific heat capacity = $c(n)$.
density = $\rho(n)$.

Time-dependent variables:—
Temperature = $T(t,n)$.
Energy flux density = $J_{int}(t,n)$

$x_n = \Delta x/2 + (n-1)\Delta x$. {mid point of element}.

Energy transfer happens through conduction.
$J(t,n)$ is flux density into nth element at $x_j(n) = (n-1)\Delta x$.
Energy flux density exiting in element is $J(t,n+1)$ at $x_j(n+1) = x_j(n) + \Delta x$.

$A(n) = \pi R(n)^2$,

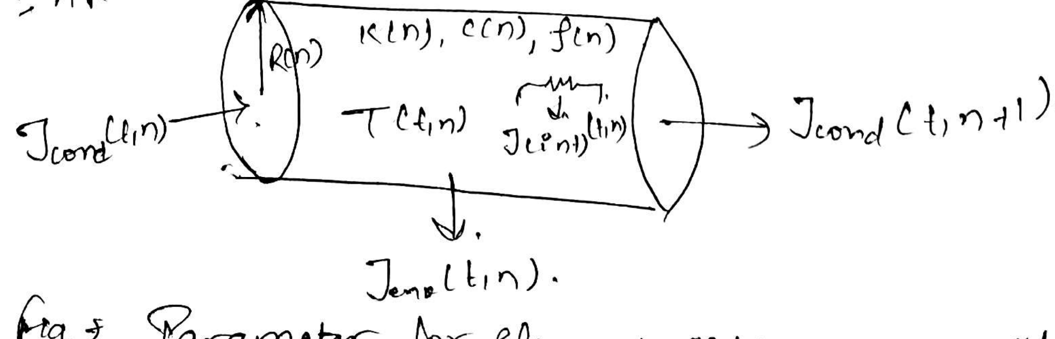


Fig : Parameter for element with index n at time t

Differential equation for eqⁿ ⑧.

$$J(t, x_n) = -\frac{K_n}{\Delta x}\left[T(t, x_{n+1}) - T(t, x_n)\right].$$

Differential equation for eqⁿ ⑨.

$$T(t+\Delta t, x_n) = T(t, x_n) + \frac{\Delta t}{\rho c\, dx}\left[J(t, x_{j,n-1}) - J(t, x_{j,n})\right].$$

For stability of numerical method, we assume,

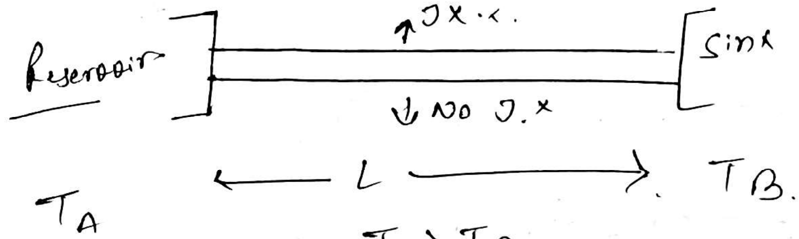
$$\frac{dt}{\rho c\, dx} < 0.5.$$

MATLAB codes.

In all the codes, output plots ⇒.
legend for figure (2) is same as legend for figure (1)
legend for figure (4) is same as legent for figure (2).

Simulations.

1). Conduction_rod_copper.m



$T_A > T_B$.

Uniform radius.

Boundary conditions:

$T(t,1) = T_A$
$T(t,N) = T_B$
$T(0,n) = T_{initial}$.

$J(t,1) = J(t,2)$ & $J(t,N) = J(t,N+1)$

The script attached, "Conduction_rod_copper.m" shows energy, temperature vs x & t plots.

Let, Length $L = 0.25$ m.
Radius $R = 0.01$ m.
$T_A = 100°C$
$T_B = 0°C$.

For copper,
Thermal conductivity, $K = 400 \, W.m^{-1}K^{-1}$.
Specific heat capacity, $c = 380 \, J \, kg^{-1} K^{-1}$
Density, $\rho = 8900 \, kg m^{-3}$

Result of program. conduction_rod_copper.m

$$\frac{dT}{dx} = -402.6846. \quad \{\text{Appears on command window}\}$$

$$\frac{d\theta}{dt} = 50.0759. \quad \{\quad " \quad\}$$

$$J = \sim 1.6 \times 10^5 \ \mathrm{W m^{-2}}$$

* Note → We can alter values of K, C, r, L in the program and obtain diff values.

Theoretical Calculations :-

$$\frac{dT}{dx} = \frac{T_B - T_A}{x(N) - x(1)} \sim \frac{100 - 0}{0.25} \ ^\circ\mathrm{C m^{-1}} = 400 \ ^\circ\mathrm{C m^{-1}}.$$

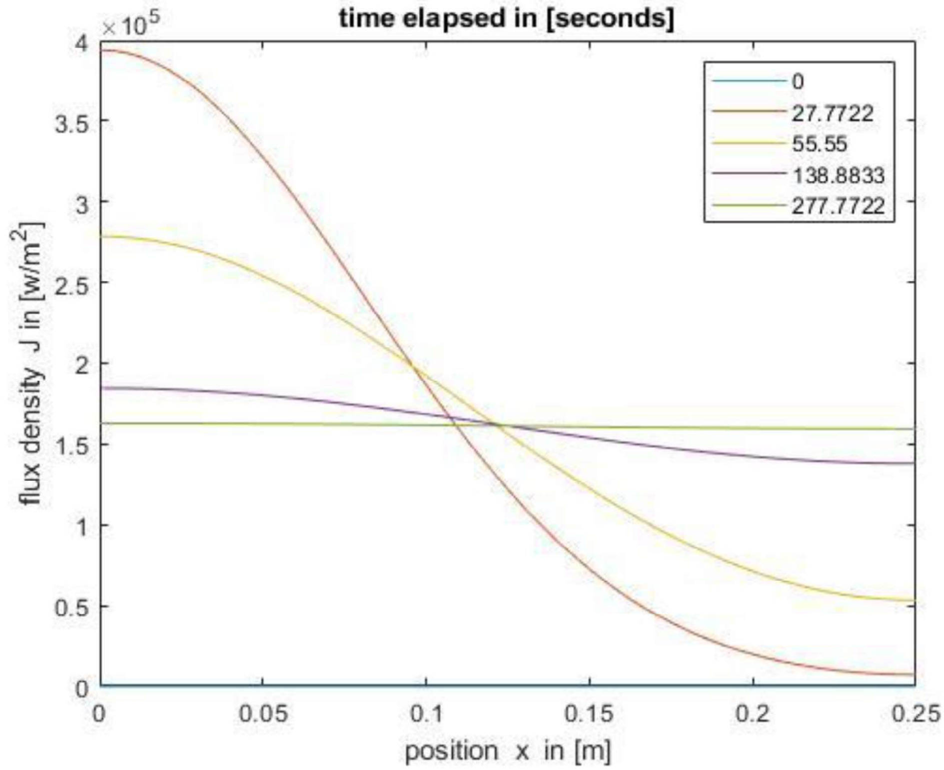
$$J = -K\frac{dT}{dx} = -400 \times 400 = 1.6 \times 10^5 \ \mathrm{W m^{-2}}.$$

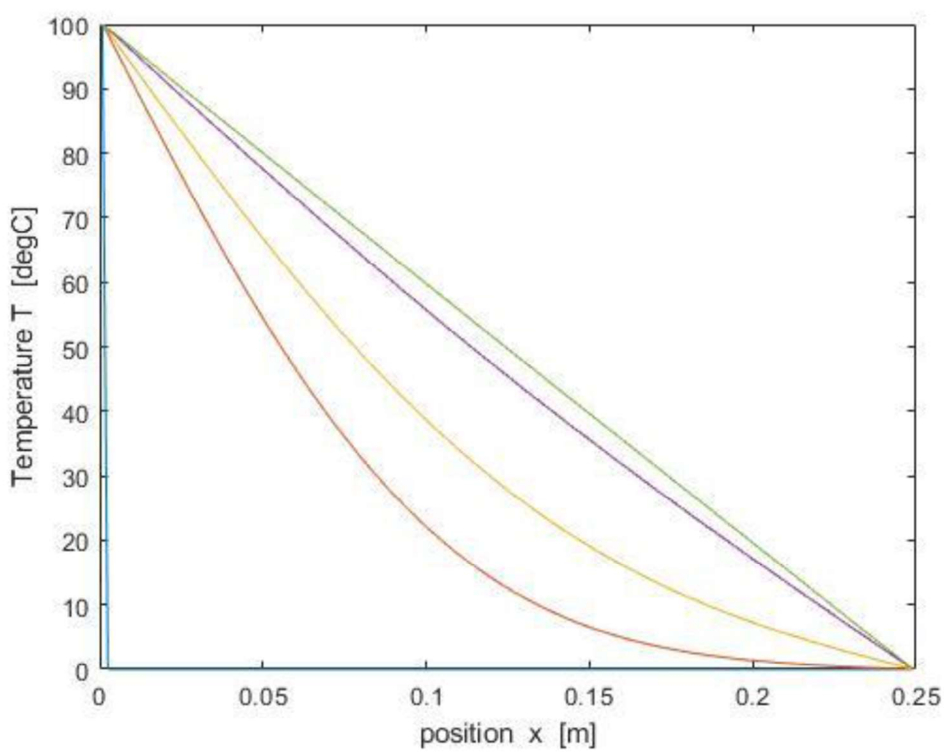
Conclusion

The programs works well. with theoretical results as well.

conduction_rod_copper.m

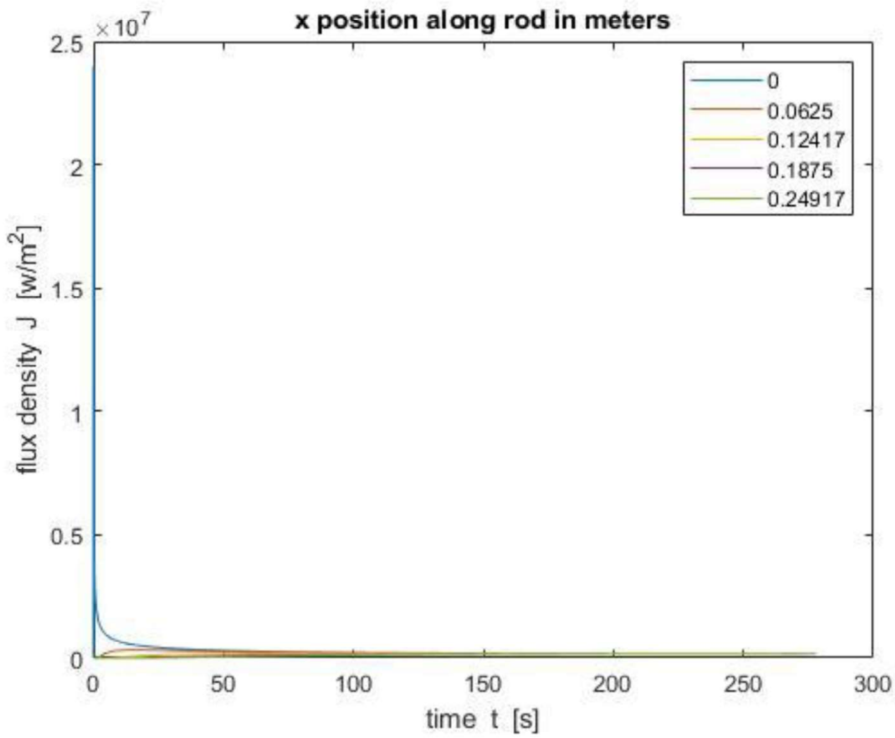
figure(1,2)

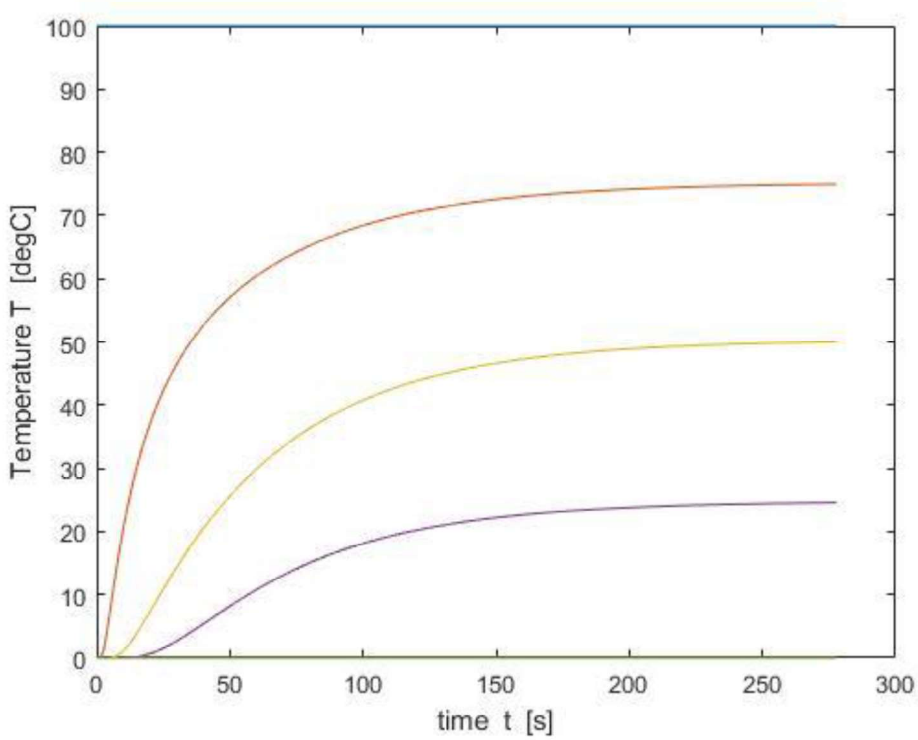




conduction_rod_copper.m

figure(3,4)





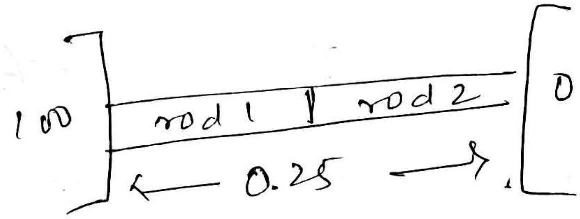
2) conduction _ rod _ series.

We now face two rod with

Copper $K_1 = 400\ W m^{-1} K^{-1}$.

$C_1 = 380\ J\ kg^{-1} K^{-1}$

$\rho_1 = 8900\ kg m^{-3}$



Iron $K_2 = 50\ W m^{-1} K^{-1}$

$C_2 = 450\ J\ kg^{-1} K^{-1}$

$\rho_2 = 7900\ kg m^{-3}$

$L = 0.250$ divided in between rods.

$R = 0.01\ m$

$T(A) = 100^\circ C$

$T(B) = 0^\circ C$.

Theoretical results.

$$\frac{\Delta T}{\Delta x} = -K_1 \frac{(T_A - T_C)}{L/2} = -K_2 \frac{T_C - T_B}{L/2}.$$

$$T_C = \frac{K_1 T_A + K_2 T_B}{K_1 + K_2} = \frac{(400)(100)}{400 + 50.} = 89^\circ C.$$

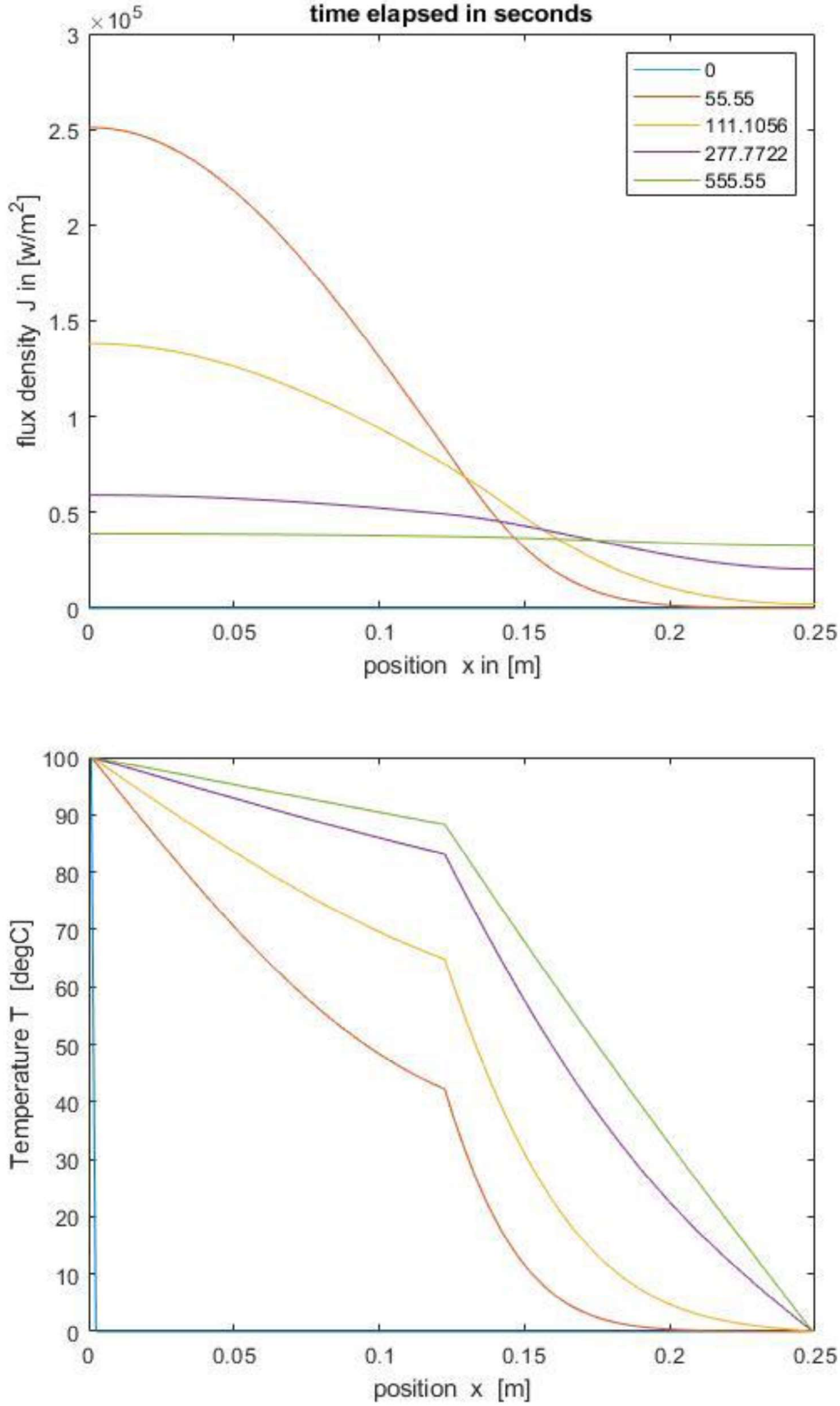
$$J = -K_1 \frac{T_A - T_C}{L/2} = \frac{-(400)(100 - 89)}{0.250/2.} = -3.52 \times 10^{4}\ W m^{-2}$$

Results from program.

$\frac{dT}{dx} = -402.6846.$ $\frac{dQ}{dt} = 10.2740.$

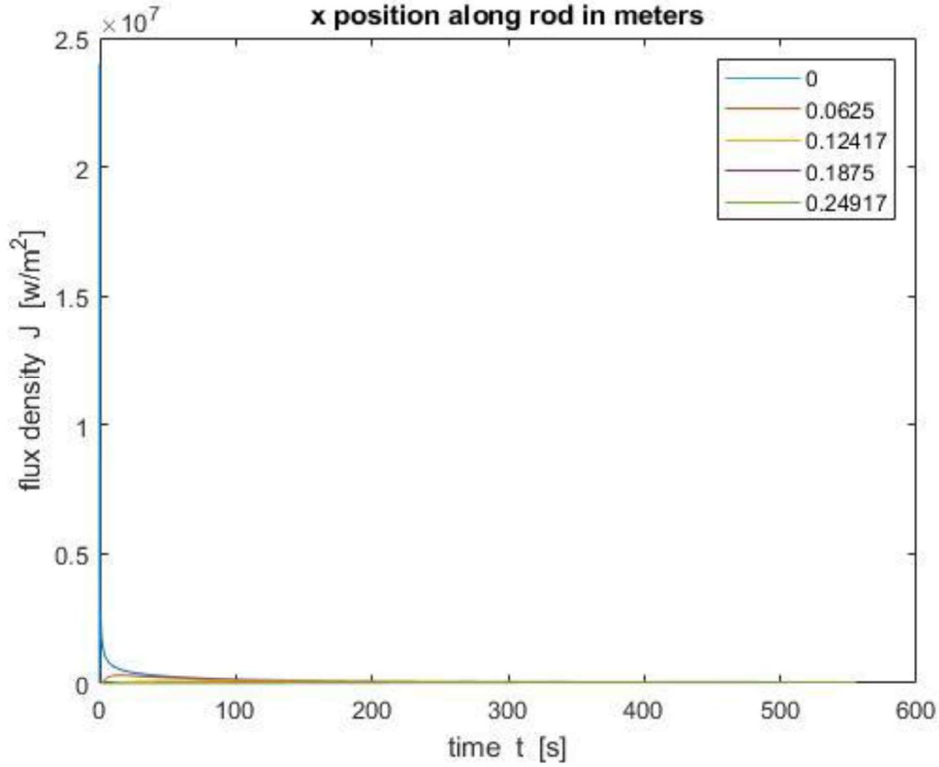
$T_C = 90^\circ C$ (steady state) figure) conduction_rod_series.m

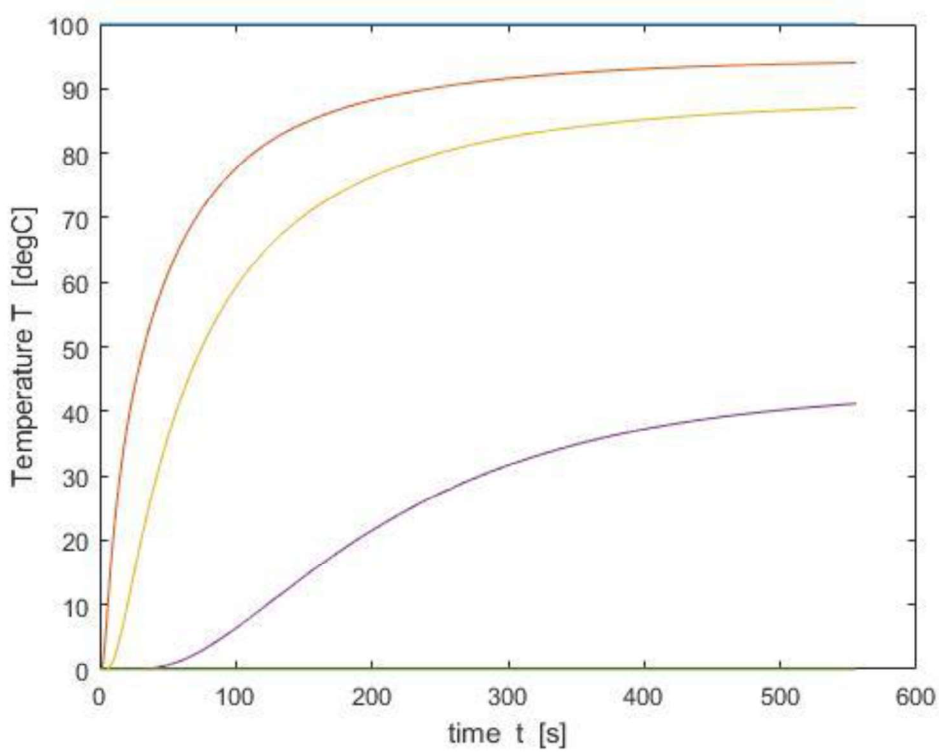
figure(1,2)



conduction_rod_series.m

figure(3,4)





3) conduction_rod_parallel.m.

~~long~~. I assumed 3 rods with.

Rod 1

$k = 400\ W m^{-1} K^{-1}$

$L_1 = 0.25/3\ m$

rad $= 0.01\ m$

$c = 380\ J\ kg^{-1} K^{-1}$

$\rho = 8900$.

Rod 2

$k = 350\ W m^{-1} K^{-1}$

$L_2 = 0.25/3\ m$

rad $= 0.01 m$

$c = 350\ J\ kg^{-1} K^{-1}$

$\rho = 8000$

Rod 3

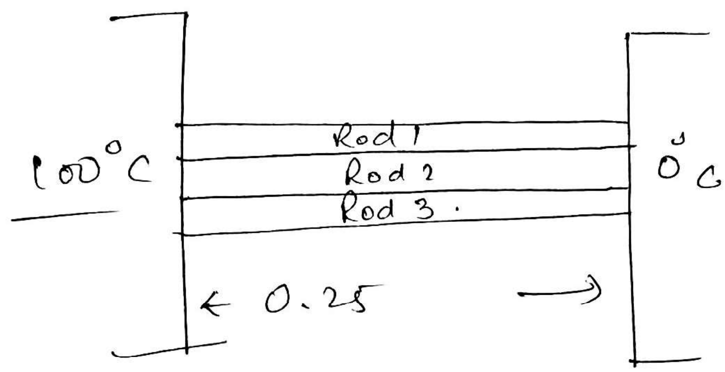
$k = 325\ W m^{-1} K^{-1}$.

$L_3 = 0.25/3\ m$

rad $= 0.01\ m$.

$c = 320\ J\ kg^{-1} K^{-1}$

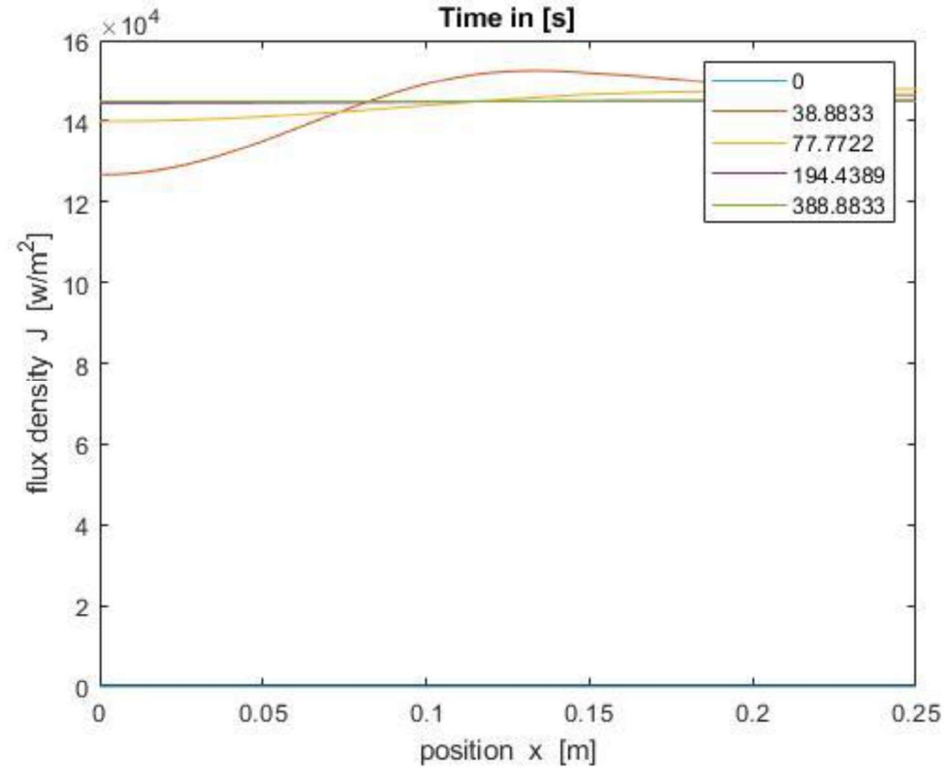
$\rho = 5000$

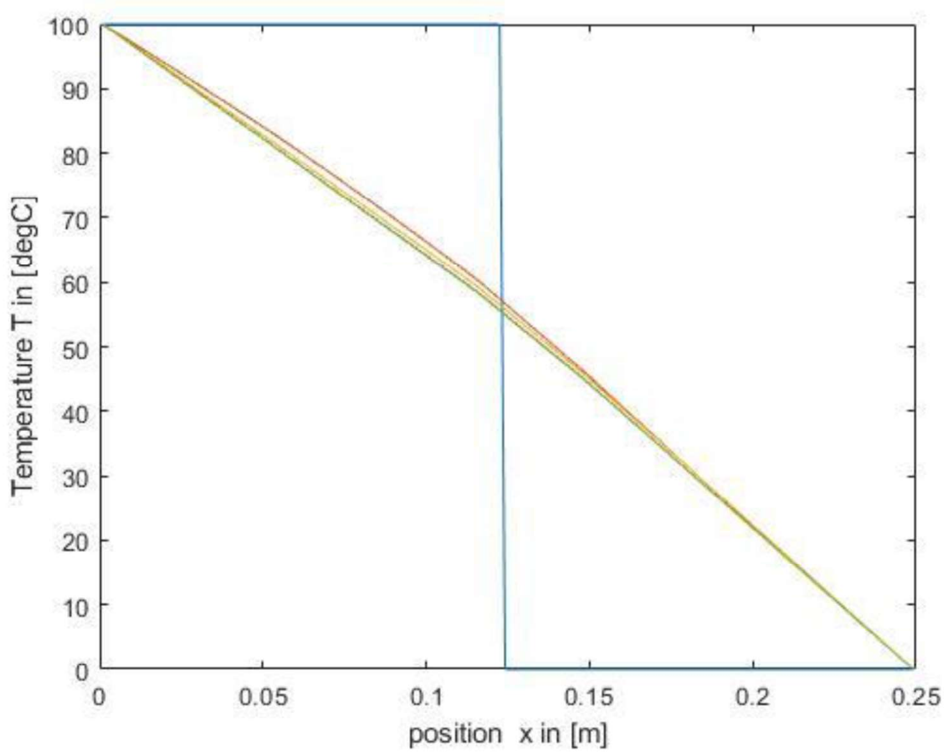


$T_A = 100$, $T_B = 0$.

conduction_rod_parallel.m

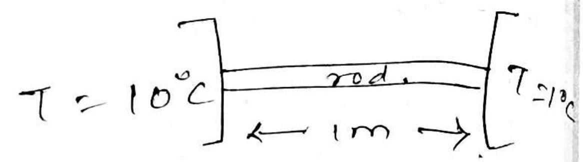
figure(1,2)





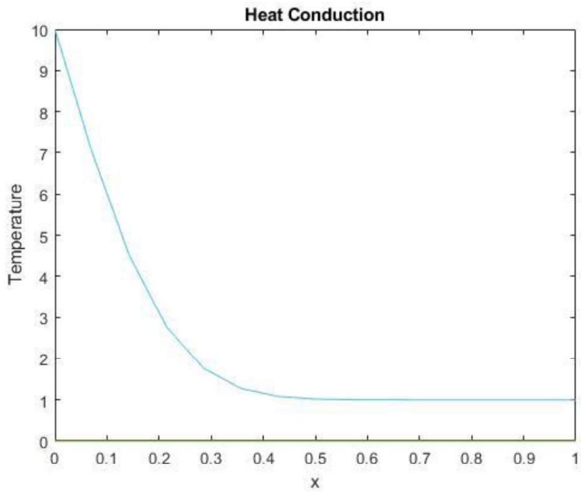
4) conduction_rod.m

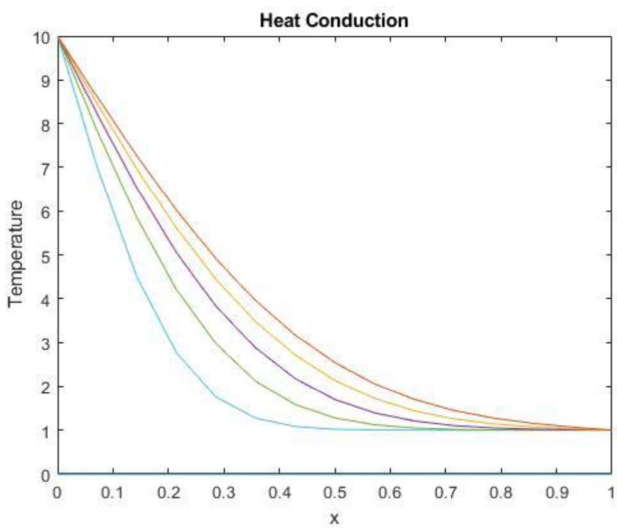
This program shows the variation of temperature vs x as time passes.

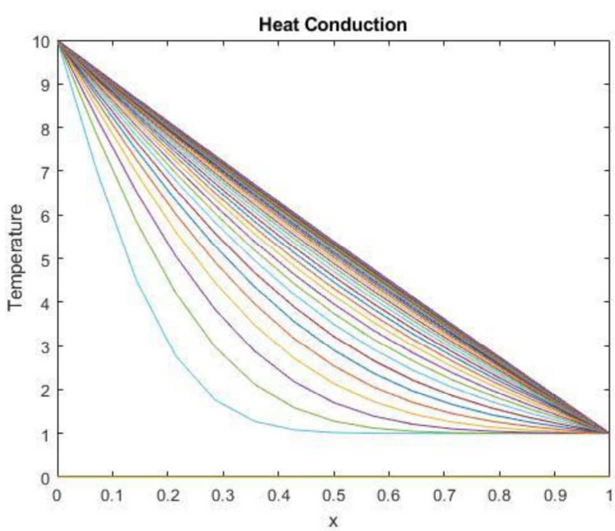


conduction_rod.m

figure at time = t_0,t_mid,t_end







%--MATLAB-CODES---%

%-----The output plots of conduction_rod_copper.m , conduction_rod_series.m &conduction_rod_parallel.m----%

%--figure 2 has same legend as figure 1 and figure 4 has same legend as figure 3-----------------------%

%---conduction_rod_copper.m--%

```
clear;clc;
%-------------------------INPUT-----------------------------%
% number of divisions made
    N = 150;
% number of time steps
    time_steps = 50000;
% length of rod
    L = 0.250;
% radius of rod
    rad_1 = 0.01;
    rad = rad_1 .* ones(1,N);
% thermal conductivity (in W/(m.degC)
    k_1 = 400;
    k = k_1 .* ones(1,N);
% specific heat capacity (in J/(kg.degC)
    c_1 = 380;
    c = c_1 .* ones(1,N);
% density  (in kg/m^3)
    rho_1 = 8900;
    rho = rho_1 .* ones(1,N);
% initial temperature of rod
    T = zeros(time_steps,N);
% intial energy flux density
    J = zeros(time_steps,N+1);

%-----------------BOUNDARY CONDITIONS-----------------------%

% End surface temperatures
    T(:,1) = 100;
    T(:,end) = 0;


%---------------------CALCULATIONS----------------------------%

dx = L / N                 % length of elements
dt = (10/3)*dx              % increment in time
x = dx/2 + dx .*(0:N-1);
xi = 0:dx:L;
t = 0 : (time_steps-1);
t = dt .* t;
K_1 = (k ./ dx);
K_2 = dt ./ (rho .* c .* dx);
area = pi * rad(1)^2;       % Area of cross section

for i = 1 : time_steps-1
for j = 2 : N
    J(i+1,j) = K_1(j) * ( T(i,j-1) - T(i,j) );
    J(i+1,1) = J(i+1,2);
    J(i+1,end) = J(i+1,end-1);
end

for j = 2 : N-1
    T(i+1,j) = T(i,j) + K_2(j) * ( J(i+1,j) - J(i+1,j+1) ) ;
end
end

% energy flux dQ/dt = JA
    dQdt = J(end,end) .* area
```

```
% temperature gradient dT/dx
    dTdx = (T(end,end) - T(end,1)) / (x(end) - x(1))
% max time
t_Max = t(end)

%-----------------------------Plots-----------------------------%

                        figure(1)
 %-------------------Energy flux density vs position-------------------%
 index = round(0.1 * time_steps) .* [1/(0.1*time_steps) 1 2 5 10];
 x_plot = xi;
 y_plot = J(index,:);
   plot(x_plot,y_plot);
   xlabel('position  x  in [m]');
   ylabel('flux density  J in [w/m^2]');
   h_title = title('time elapsed in [seconds]')
   time_1 = '0';
   time_2 = num2str(t(index(2)));
   time_3 = num2str(t(index(3)));
   time_4 = num2str(t(index(4)));
   time_5 = num2str(t(index(5)));
   h_legend = legend(time_1,time_2,time_3,time_4, time_5);

figure(2)
   x_plot = x;
   y_plot = T(index,:);
   plot(x_plot,y_plot);
   xlabel('position  x in [m]');
   ylabel('Temperature T in [degC]');

                        figure(3)
%----------------- energy flux density J vs time plot -----------------%
index = round((0.25 * N) .* [1/(0.25*N) 1 2 3 4]);

   x_plot = t; y_plot = J(:,index);
   plot(x_plot,y_plot);
   xlabel('time  t  [s]');
   ylabel('flux density  J  [w/m^2]');
   h_title = title('x position along rod in meters');

   position_1 = '0';
   position_2 = num2str(x(index(2)));
   position_3 = num2str(x(index(3)));
   position_4 = num2str(x(index(4)));
   position_5 = num2str(x(index(5)));
   h_legend = legend(position_1,position_2,position_3,position_4, position_5);

                        figure(4)
%-----------------temperature T vs time t-------------------%
   x_plot = t; y_plot = T(:,index);
   plot(x_plot,y_plot);
   xt = 'time  t  [s]';
   yt = 'Temperature T  [degC]';
   xlabel('time  t in [s]');
   ylabel('Temperature T in [degC]');
```

%---conduction_rod_series.m---%

```
clear;clc;
%---------------------------INPUTS------------------------%
% number of divisions made
    N = 150;
% number of time steps
    time_steps = 100000;
% length of rod
    L = 0.250;
% radius of rod
    rad_1 = 0.01;
    rad = rad_1 .* ones(1,N);
% thermal conductivity of rod (in W/(m.degC)
    k_1 = 400;
    k_2 = 50;
    k = k_1 .* ones(1,N);
    k(N/2:end) = k_2;
    %k(100:end) = k3;
% specific heat capacity (in J/(kg.degC)
    c_1 = 380;
    c_2 = 450;
    c = c_1 .* ones(1,N);
    c(N/2:end) = c_2;
% density  (in kg/m^3)
    rho_1 = 8900;
    rho_2 = 7900;
    rho = rho_1 .* ones(1,N);
    rho(N/2:end) = rho_2;

% initial temperature of rod
    T = zeros(time_steps,N);

% intial energy flux density of rod
    J = zeros(time_steps,N+1);

%-----------------BOUNDARY CONDITIONS-----------------------%

% End surface temperatures
    T(:,1) = 100;
    T(:,end) = 0;


%---------------------CALCULATIONS-----------------------------%

dx = L / N                  % length of elements
dt = (10/3)*dx               % increment in time
x = dx/2 + dx .*(0:N-1);
xi = 0:dx:L;
t = 0 : (time_steps-1);
t = dt .* t;
K_1 = (k ./ dx);
K_2 = dt ./ (rho .* c .* dx);
area = pi * rad(1)^2;        % Area of cross section

for i = 1 : time_steps-1
for j = 2 : N
    J(i+1,j) = K_1(j) * ( T(i,j-1) - T(i,j) );
    J(i+1,1) = J(i+1,2);
    J(i+1,end) = J(i+1,end-1);
end

for j = 2 : N-1
    T(i+1,j) = T(i,j) + K_2(j) * ( J(i+1,j) - J(i+1,j+1) ) ;
end
end
```

```
% energy flux dQ/dt = JA
    dQdt = J(end,end) .* area
% temperature gradient dT/dx
    dTdx = (T(end,end) - T(end,1)) / (x(end) - x(1))
% max time
t_Max = t(end)

%-----------------------------Plots-----------------------------%

                        figure(1)
 %------------------Energy flux density vs position-------------------%
 index = round(0.1 * time_steps) .* [1/(0.1*time_steps) 1 2 5 10];
 x_plot = xi;
 y_plot = J(index,:);
   plot(x_plot,y_plot);
   xlabel('position  x  in [m]');
   ylabel('flux density  J in [w/m^2]');
   h_title = title('time elapsed in [seconds]')
   time_1 = '0';
   time_2 = num2str(t(index(2)));
   time_3 = num2str(t(index(3)));
   time_4 = num2str(t(index(4)));
   time_5 = num2str(t(index(5)));
   h_legend = legend(time_1,time_2,time_3,time_4, time_5);

figure(2)
   x_plot = x;
   y_plot = T(index,:);
   plot(x_plot,y_plot);
   xlabel('position  x in [m]');
   ylabel('Temperature T in [degC]');

                        figure(3)
%----------------- energy flux density J vs time plot -----------------%
index = round((0.25 * N) .* [1/(0.25*N) 1 2 3 4]);

   x_plot = t; y_plot = J(:,index);
   plot(x_plot,y_plot);
   xlabel('time  t  [s]');
   ylabel('flux density  J  [w/m^2]');
   h_title = title('x position along rod in meters');

   position_1 = '0';
   position_2 = num2str(x(index(2)));
   position_3 = num2str(x(index(3)));
   position_4 = num2str(x(index(4)));
   position_5 = num2str(x(index(5)));
   h_legend = legend(position_1,position_2,position_3,position_4, position_5);

                        figure(4)
%-----------------temperature T vs time t-------------------%
   x_plot = t; y_plot = T(:,index);
   plot(x_plot,y_plot);
   xt = 'time  t  [s]';
   yt = 'Temperature T  [degC]';
   xlabel('time  t in [s]');
   ylabel('Temperature T in [degC]');
```

%---conduction_rod_parallel.m--%

```
clear;clc;
%------------------------------INPUTS---------------------------------

% number of divisions made
    N = 150;
    n(1) = 70; n(2) = 20 + n(1);
% number of time steps
    time_steps = 70000;
% length of rods (in m)
    L = 0.25;
% radius of rod (in m)
    r1 = .01;
    rad = r1 .* ones(1,N);
% thermal conductivity of rods (in W/(m.degC)
    k_1 = 400;
    k_2 = 350;
    k_3 = 325;
    k = k_1 .* ones(1,N);
    k(n(1)+1:end) = k_2;
    k(n(2)+1:end) = k_3;

% specific heat capacity of rods (in J/(kg.degC)
    c_1 = 380;
    c_2 = 350;
    c_3 = 320;
    c = c_1 .* ones(1,N);
    c(n(1)+1:end) = c_2;
    c(n(2)+1:end) = c_3;

% density of rods (in kg/m^3)
    rho1 = 8900;
    rho2 = 8000;
    rho3 = 5000;
    rho = rho1 .* ones(1,N);
    rho(n(1)+1:end) = rho2;
    rho(n(2)+1:end) = rho3;


% initial temperature of rods  (in degC)
    T = zeros(time_steps,N);

% intial energy flux density
    J = zeros(time_steps,N+1);

%------------------ BOUNDARY CONDITIONS --------------------%
    T(:,1) = 100;
    T(:,end) = 0;
    T(:,1:end) = 100;
    T(:,end/2:end) = 0;
%------------------------- CALCULATIONS --------------------------%
dx = L / N;                    % width of elements
dt = (10/3)*dx                   %increment in time
x = dx/2 + dx .*(0:N-1);
xi = 0:dx:L;
t = 0 : (time_steps-1);
t = dt .* t;
K_1 = (k ./ dx);
K_2 = dt ./ (rho .* c .* dx);
area = pi * rad(1)^2;                % cross-sectional area
for i = 1 : time_steps-1
for j = 2 : N
    J(i+1,j) = K_1(j) * ( T(i,j-1) - T(i,j) );
    J(i+1,1) = J(i+1,2);
    J(i+1,end) = J(i+1,end-1);
```

```
end

for j = 2 : N-1
    T(i+1,j) = T(i,j) + K_2(j) * ( J(i+1,j) - J(i+1,j+1) ) ;
end
end

%-----------------------------plots--------------------------------%
                            figure(1);
%-------------- energy flux density J vs position -----------------%
index = round(0.1 * time_steps) .* [1/(0.1*time_steps) 1 2 5 10];

   xplot = xi;
   yplot = J(index,:);
   plot(xplot,yplot);
   xt = 'position  x  [m]';
   yt = 'flux density  J  [w/m^2]';
   xlabel(xt);
   ylabel(yt);
   h_title = title('Time in [s]');
   time_1 = '0';
   time_2 = num2str(t(index(2)));
   time_3 = num2str(t(index(3)));
   time_4 = num2str(t(index(4)));
   time_5 = num2str(t(index(5)));
   h_legend = legend(time_1,time_2,time_3,time_4, time_5);

                            figure(2)

% -----------------temperature T vs position ----------------------%
  index = round(0.1 * time_steps) .* [1/(0.1*time_steps) 1 2 5 10];
  xplot = x; yplot = T(index,:);
   xplot = x; yplot = T(index,:);
   plot(xplot,yplot);
   xt = 'position  x in [m]';
   yt = 'Temperature T in [degC]';
   xlabel(xt);
   ylabel(yt);
```

%--conduction_rod.m--%

```
% Heat conduction

clear; close all; clc;
n = 15;                 %no. of divisions
T = ones(n,1);
%Fourier's law

Temp = T;
T(1) = 10;           %end surface temperatures
T(n) = 1;
x = linspace(0,1,n);    %length in m
dx = x(2)-x(1);
dt = dx.^2/5;
kappa = 1;       %thermal diffusivity

error = 1;
min = .001;
k = 0;
while error > min,
   Temp = T;
   k = k+1;
   for i = 2:n-1
    T(i) = dt*kappa*(Temp(i+1)-2*Temp(i)+Temp(i-1))/dx^2+Temp(i);
   end
   error = max(abs(T-Temp));
   if mod(k,13)==0,
       y(k,:) = T;
       plot(x,y)
        xlabel('x'),ylabel('Temperature'),
        title(['Heat Conduction']),
        pause(.2),
   end
end
```