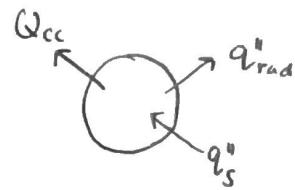


Problem 1

Problem 1

(1)

$$\dot{E}_{in} - \dot{E}_{out} + \dot{E}_{gen} = \dot{E}_{st}$$



$$-Q_{cc} + \bar{q}_s'' A_s - \bar{q}_{rad}'' A_s = \rho C_p \frac{\partial T}{\partial t} = M C_p \frac{\partial T}{\partial t}$$

$$-Q_{cc} + (\bar{q}_s'' + \Delta \bar{q}_s'' \sin(\omega t)) A_s - \varepsilon \sigma A_s (T^4 - T_{surr}^4) = M C_p \frac{\partial T}{\partial t}$$

$$\frac{\partial T}{\partial t} = \underbrace{-\frac{Q_{cc}}{M C_p}}_a + \underbrace{\frac{A_s \bar{q}_s''}{M C_p}}_b + \underbrace{\frac{A_s \Delta \bar{q}_s'' \sin(\omega t)}{M C_p}}_c - \underbrace{\frac{\varepsilon \sigma A_s (T^4 - T_{surr}^4)}{M C_p}}_d$$

$$T(t=0) = 300K$$

(2) *Jupyter Notebook*

(3) $T_{ss} = 166.5 K$

$$h_r = A_s \varepsilon \sigma (T_{ss} + T_{surr}) (T_{ss}^2 + T_{surr}^2)$$

E

$$\rho_{silicone} = 2330$$

$$\text{And } f = M$$

$$f = \frac{M}{g}$$

$$L_c = \frac{f}{A_s} =$$

$$Bi = \frac{hL}{k}$$

In [27]:

```
#@title
# import a few more things
from scipy.integrate import odeint
```

```

from scipy.constants import Stefan_Boltzmann as sigma #Stefan-Boltzmann Constant [W/m^2-K^4]
import math # math functions
import numpy as np # numerical math
from numpy import *
from scipy.linalg import solve
from scipy.constants import pi

# Widgets
import ipywidgets as widgets
from IPython.display import display
from ipywidgets import interact

#plotting
params = {'legend.fontsize': 'xx-large',
           'figure.figsize': (15, 10),
           'axes.labelsize': 'xx-large',
           'axes.titlesize':'xx-large',
           'xtick.labelsize':'xx-large',
           'ytick.labelsize':'xx-large'}
import matplotlib.pyplot as plt
plt.rcParams.update(params)
# Additional Parameters
T_surr = 300 #[K] surroudnings temperature
epsilon = 0.35      #[-] Estimate for emissivity
t_max=5000
sigma=5.67*10**-8
Ti=300
Qcc=5
M=0.05
As=0.02
omega=0.02094
cp=300
def model(y,t,a,b,c, d, T_surr):    # Set up the function that returns dTdt
    dTdt = a+b+c*sin(omega*t)+d*(y**4-T_surr**4)
    return dTdt

a = -Qcc/(M*cp)
b = 100*As/(M*cp)
c = 100*As/(M*cp)
d=-epsilon*sigma*As/(M*cp)

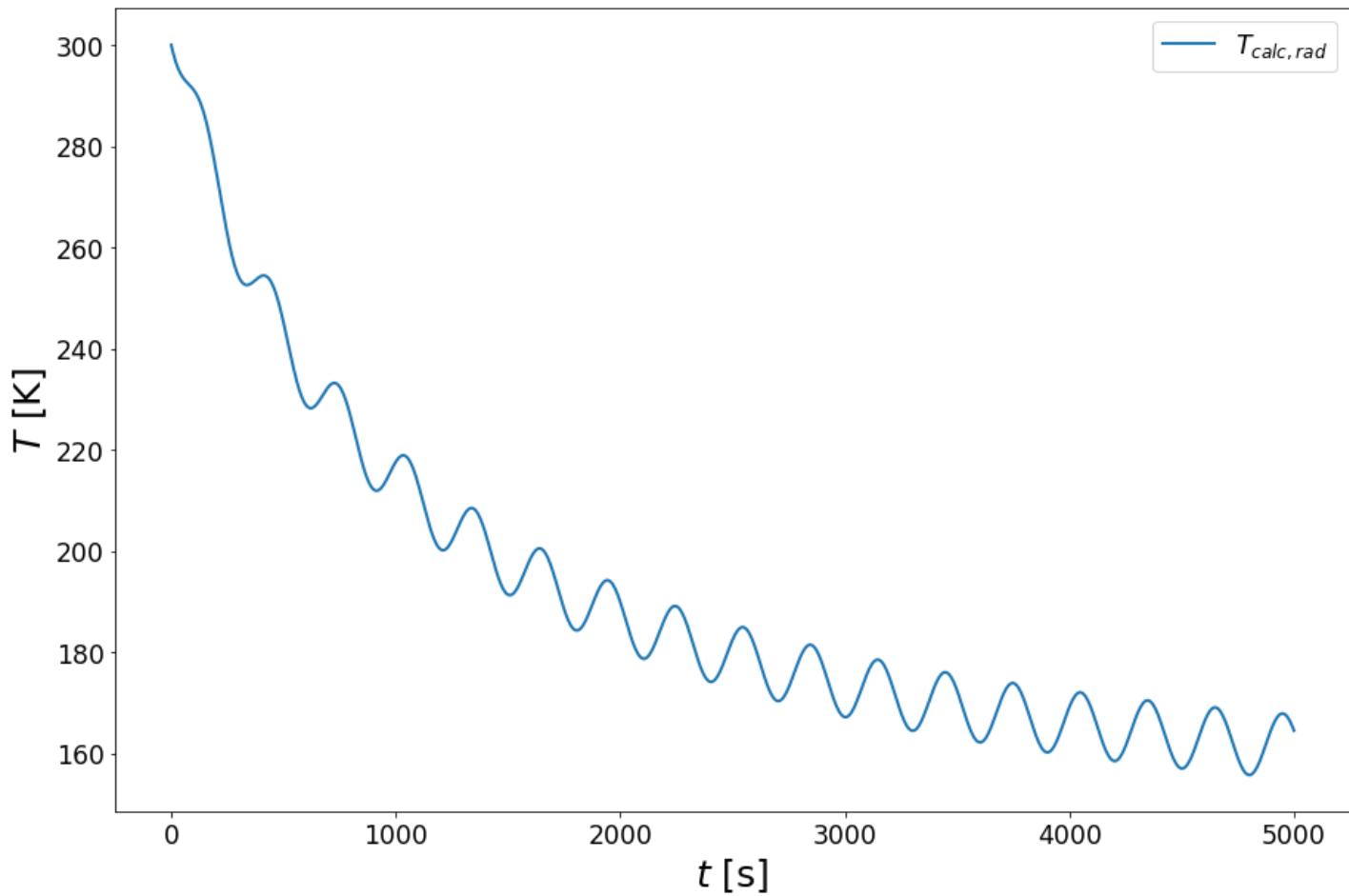
t = np.linspace(0,t_max,1000)

Tr = odeint(model,Ti, t, args=(a,b,c, d, T_surr))
plt.plot(t,Tr,label = '$T_{calc,rad}$', lw =2)
plt.xlabel("$t$ [s]", fontsize=25)
plt.ylabel("$T$ [K]", fontsize=25)
plt.legend( )

T_ss=sum(Tr[-10:])/len(Tr[-10:])
print('Steady state Temperature [K]:', T_ss)
h_r=epsilon*sigma*As*(T_ss+T_surr)*(T_ss**2+T_surr**2)
print('h_r [W/m^2K]:', h_r)

```

Steady state Temperature [K]: 166.46439747859714
h_r [W/m^2K]: 0.021792869612218595



In [28]:

```
rho=2330
ks=[884, 264]
kT=[100,200]
k=np.interp(T_ss, kT, ks)
V=M/rho
Lc=V/As
Bi=h_r*Lc/k
print('Bi=', Bi)
```

Bi= 4.954837866553643e-08

Problem 2

Problem 2

$$\frac{\partial}{\partial x} \left(k \frac{\partial T}{\partial x} \right) + \frac{\partial}{\partial y} \left(k \frac{\partial T}{\partial y} \right) + \frac{\partial}{\partial z} \left(k \frac{\partial T}{\partial z} \right) + \dot{q} = g C_p \frac{\partial T}{\partial t}$$

$$\alpha = \frac{k}{\rho C_p}$$

$$\frac{d^2 T}{dy^2} = \underbrace{\frac{1}{\alpha}}_{\eta} \frac{\partial T}{\partial t}$$

$$\eta = \frac{y}{\sqrt{4\alpha t}}, \quad \frac{\partial \eta}{\partial y} = \frac{1}{\sqrt{4\alpha t}}, \quad \frac{\partial \eta}{\partial t} = \frac{-y}{2t\sqrt{4\alpha t}}$$

$$\frac{\partial T}{\partial y} = \frac{\partial T}{\partial \eta} \frac{\partial \eta}{\partial y} = \underbrace{\frac{\partial T}{\partial \eta}}_{\frac{1}{\sqrt{4\alpha t}}} \frac{1}{\sqrt{4\alpha t}}$$

$$\frac{\partial T}{\partial t} = \frac{\partial T}{\partial \eta} \frac{\partial \eta}{\partial t} = \frac{\partial T}{\partial \eta} \left(\frac{-y}{2t\sqrt{4\alpha t}} \right) = \underbrace{\frac{-n}{2t} \frac{\partial T}{\partial \eta}}$$

$$\frac{\partial^2 T}{\partial y^2} = \frac{\partial}{\partial \eta} \left[\frac{\partial T}{\partial y} \right] \frac{\partial \eta}{\partial y} = \frac{\partial}{\partial \eta} \left[\frac{\partial T}{\partial \eta} \frac{1}{\sqrt{4\alpha t}} \right] \frac{1}{\sqrt{4\alpha t}} = \frac{1}{4\alpha t} \frac{\partial^2 T}{\partial \eta^2}$$

$$\frac{\partial^2 T}{\partial y^2} = \underbrace{\frac{1}{\alpha} \frac{\partial T}{\partial t}}_{\frac{1}{4\alpha t}} = \frac{1}{4\alpha t} \frac{\partial^2 T}{\partial \eta^2} \rightarrow \frac{\partial^2 T}{\partial \eta^2} = \frac{4\alpha t}{\alpha} \frac{\partial T}{\partial t}$$

$$\boxed{\frac{\partial^2 T}{\partial y^2} = -2\eta \frac{\partial T}{\partial \eta}}$$

Boundary Conditions

$$T(\eta=0) = T_s$$

$$T(\eta \rightarrow \infty) = T_i$$

$$\frac{d}{d\eta} \left(\frac{\frac{\partial T}{\partial \eta}}{\Theta} \right) = -2\eta \frac{\frac{\partial T}{\partial \eta}}{\Theta} \rightarrow \frac{d\Theta}{d\eta} = -2\eta \Theta \rightarrow \frac{d\Theta}{\Theta} = -2\eta d\eta \rightarrow \Theta = C_1 e^{-\eta^2} = \frac{dT}{d\eta}$$

$$dT = C_1 e^{-\eta^2} d\eta \rightarrow T = C_1 \int_0^\eta e^{-u^2} du + C_2$$

$$\text{Boundary Condition} \rightarrow T_s = C_1 \int_0^0 e^{-u^2} du + C_2 \rightarrow C_2 = T_s$$

$$T = T_s \int_0^\eta e^{-u^2} du + \frac{2}{\sqrt{\pi}} T_i - T_s \quad T_i = C_1 \left(\frac{\sqrt{\pi}}{2} \right) + T_s \rightarrow C_1 = \frac{2}{\sqrt{\pi}} T_i - T_s$$

$$\frac{T - T_s}{T_i - T_s} = \frac{2}{\sqrt{\pi}} \int_0^\eta e^{-u^2} du = \operatorname{erf}(\eta)$$

→

$$T(y, t) = \operatorname{erf} \left(\frac{y}{2\sqrt{\alpha t}} \right) (T_i - T_s) + T_s$$

Matches ✓

3

$$60 + 273.15 = \operatorname{erf} \frac{\frac{10E-3}{2\sqrt{\frac{k}{(2500)(1000)}}(80)}}{((20+273.15) - (60+273.15)) + (60+273.15)}$$

$$R = 3.4345 \frac{W}{mK}$$

4

$$\sqrt{E''} = \int q'' dt$$

$$\int q'' dt \rightarrow q'' = -k \frac{dT}{dx} \Big|_{x=0} = -\frac{k(T_i - T_s)}{\sqrt{\pi \alpha t}} \rightarrow \int_0^{80} -\frac{k(T_i - T_s)}{\sqrt{\pi \alpha t}} t^{-1/2} dt = \frac{2k(T_s - T_i)}{\sqrt{\pi \alpha}}$$

$$E'' = 2,365,890 \text{ J/m}^2$$

5

Jupyter Notebook

6

$$F_0 = \frac{\alpha t}{L^2} \leq 0.2 \quad \text{For thickness}$$

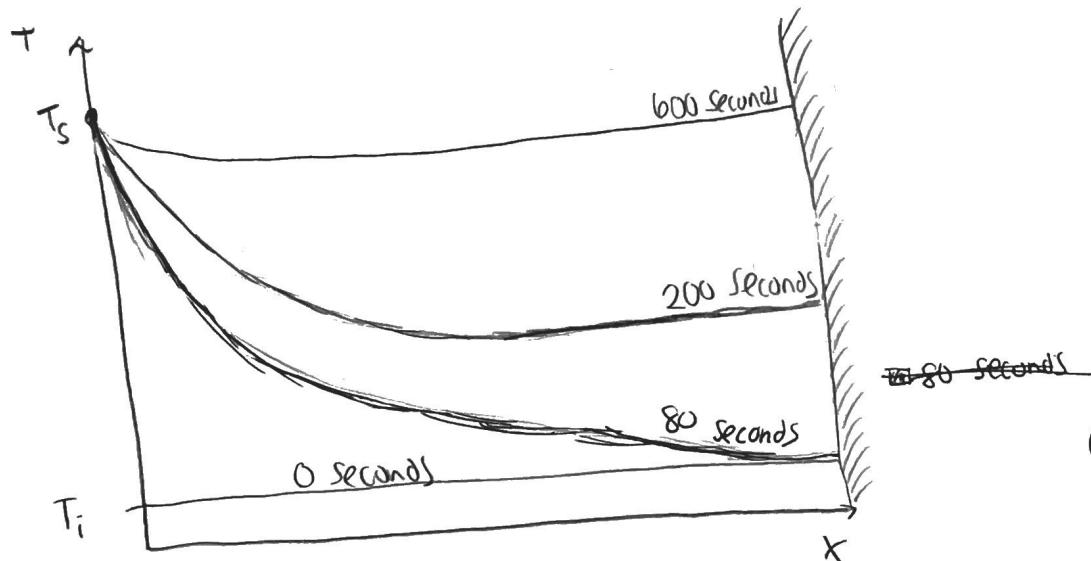
$$F_0 = \frac{1.374E-6 t}{(25E-3)^2} = 0.2$$

$$t \leq 91.7 \text{ seconds}$$

$$L = 25E-3 \text{ m}$$

$$\alpha = 1.374 E-6$$

7



In [29]:

```
#Symbolic math
```

```
import sympy # symbolic math
```

```
import numpy as np # numerical math
```

```
from sympy import *
```

```
import math # math functions
```

```
from sympy import oo
```

```
k, Ti, Ts, alpha, x, t, eta, rho, cp, L = symbols('k Ti Ts alpha x t eta rho cp L')
```

```

T = symbols('T', cls=Function)    # define T as a function
eq = T(eta).diff(eta,eta)+2*eta*T(eta).diff(eta)
T = dsolve(eq)
C1, C2 = symbols('C1 C2')      # define these as symbols
constants = sympy.solve([T.rhs.subs(eta,0)-Ts,T.rhs.subs(eta,oo)-Ti],[C1,C2]) #Solve for C1-C2
T = T.rhs.subs(constants).simplify() # Substitute constants C1 and C2 into the equation
T = T.subs(eta,x/(4*alpha*t)**(1/2)) # Substitute x,t into the equation instead of eta
T_s=60+273.15
rhos=2500
cps=1000
ts=80
Ts_s=100+273.15
Tis=20+273.15
Ls=10*10**-3
T = T.subs([(Ti, Tis), (Ts, Ts_s), (x, Ls), (t, ts)])
alphavalue=sympy.solve(T - T_s, alpha)
alphas=1.37381833644858*10**-6
ks=alphas*rhos*cps
print('k [W/mK] = ', ks)

```

k [W/mK] = 3.4345458411214502

In [30]:

```

Thermal_Energy = (2*ks*(Ts_s-Tis)*ts**(1/2))/((math.pi*alphas)**(1/2))
print(Thermal_Energy)

```

2365890.8399153897

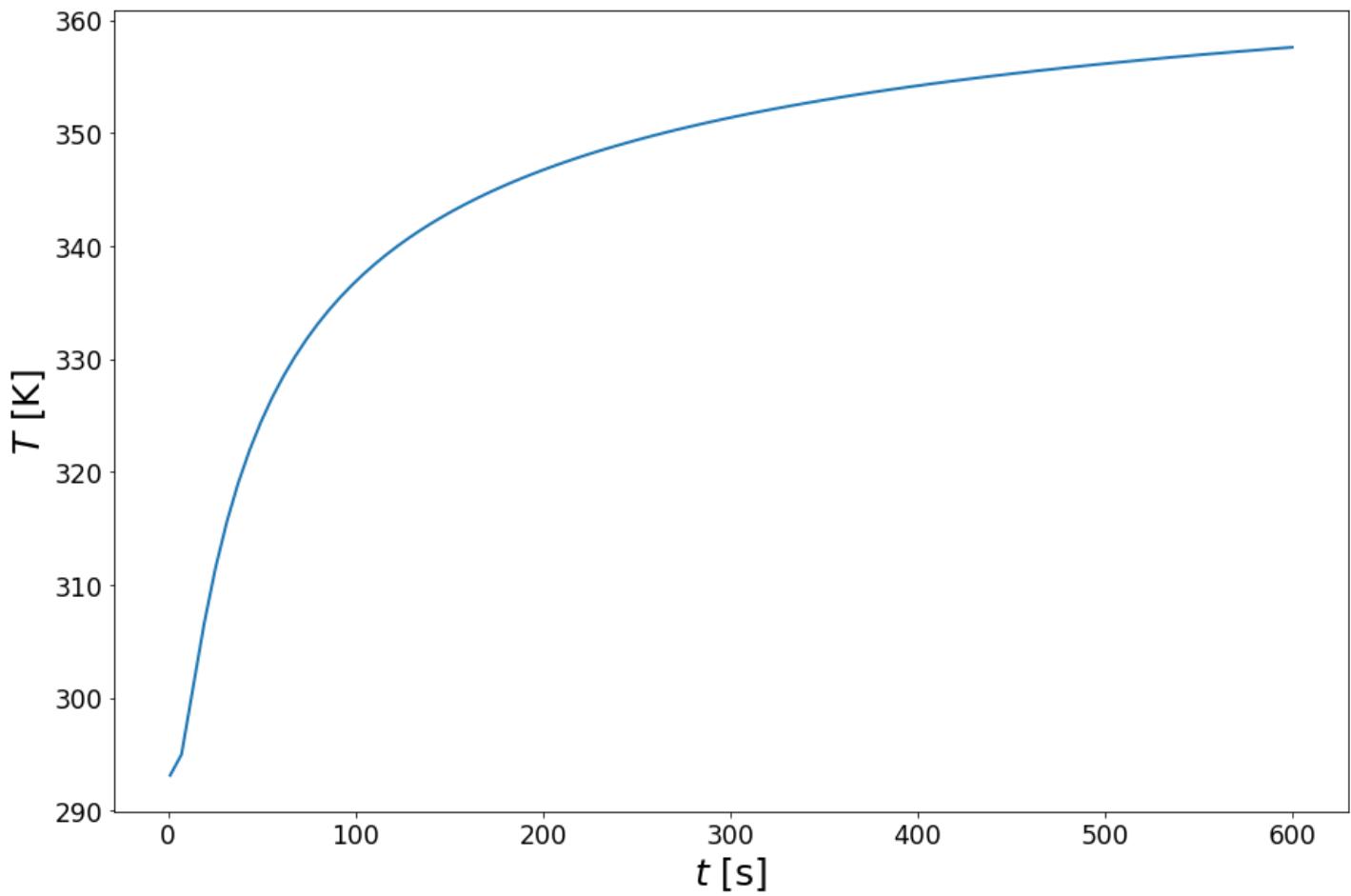
In [31]:

```

import matplotlib.pyplot as plt
k, Ti, Ts, alpha, x, t, eta, rho, cp, L = symbols('k Ti Ts alpha x t eta rho cp L')
T = symbols('T', cls=Function)    # define T as a function
eq = T(eta).diff(eta,eta)+2*eta*T(eta).diff(eta)
T = dsolve(eq)
C1, C2 = symbols('C1 C2')      # define these as symbols
constants = sympy.solve([T.rhs.subs(eta,0)-Ts,T.rhs.subs(eta,oo)-Ti],[C1,C2]) #Solve for C1-C2
T = T.rhs.subs(constants).simplify() # Substitute constants C1 and C2 into the equation
T = T.subs(eta,x/(4*alpha*t)**(1/2)) # Substitute x,t into the equation instead of eta
rhos=2500
ks=3.4345458411214502
alphas=1.37381833644858*10**-6
cps=1000
Ts_s=100+273.15
Tis=20+273.15
Ls=10*10**-3
Tp = T.subs([(Ti, Tis), (Ts, Ts_s), (x, Ls), (alpha, alphas)])
Tp = lambdify(t, Tp,"numpy")
Tp = np.vectorize(Tp)
tp = np.linspace(1, 10*60,num=100)
Tp=Tp(tp)
plt.plot(tp,Tp,label = 'T', lw =2)
plt.xlabel("$t$ [s]", fontsize=25)
plt.ylabel("$T$ [K]", fontsize=25)

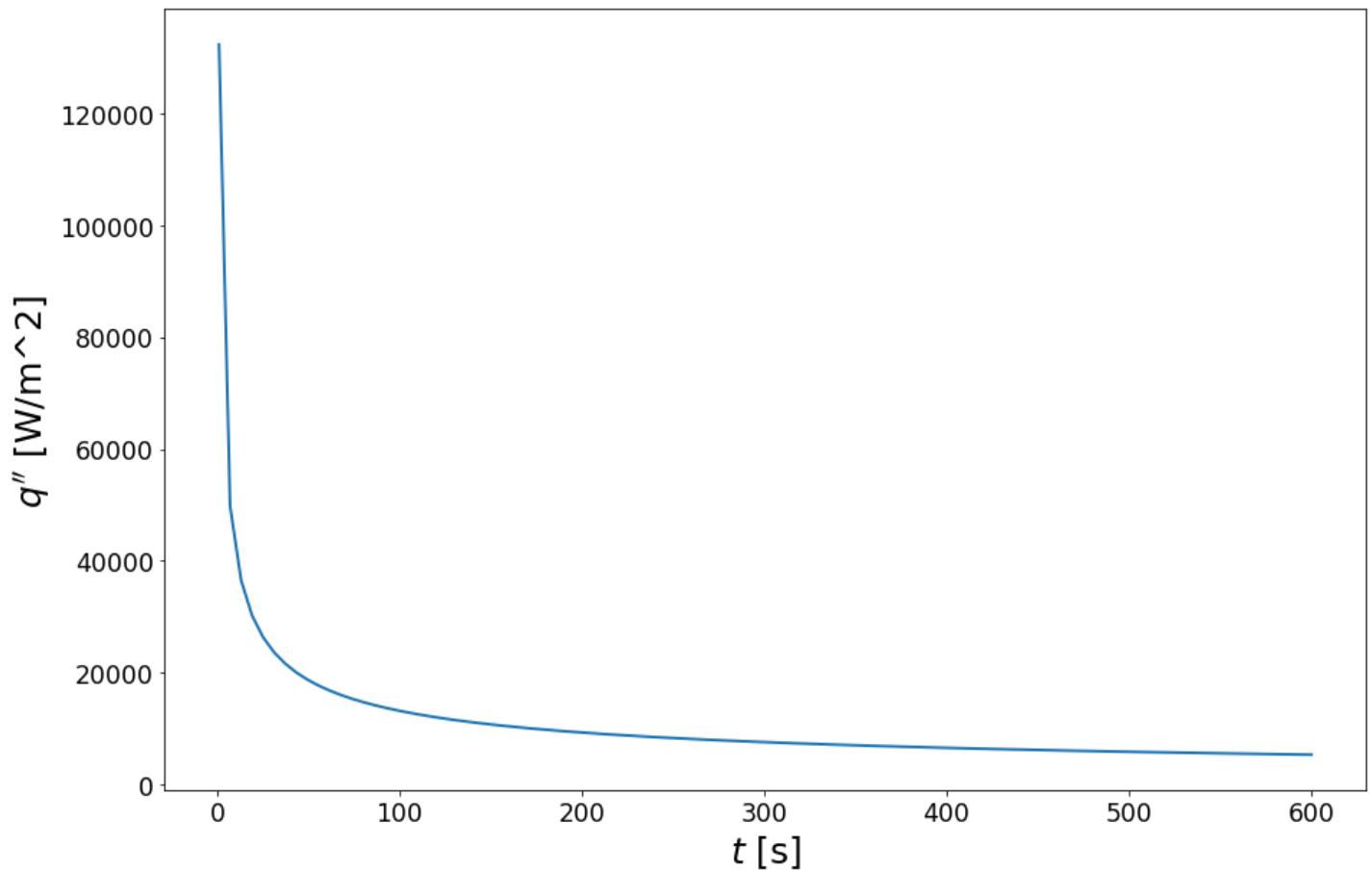
```

Out[31]: Text(0, 0.5, '\$T\$ [K]')



```
In [32]: q_dprime = ks*(Ts_s-Tis)/((math.pi*alphas*tp)**(1/2))
plt.plot(tp,q_dprime,label = 'T', lw =2)
plt.xlabel("$t$ [s]",fontsize=25)
plt.ylabel("$q''$ [W/m^2]",fontsize=25)
```

```
Out[32]: Text(0, 0.5, "$q''$ [W/m^2]")
```

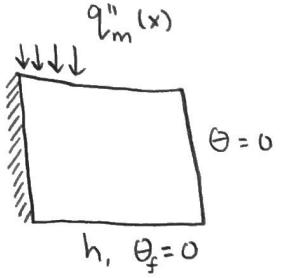


```
In [33]: time=0.2*(25*10**-3)**2/(1.374*10**-6)  
print('time less than', time)
```

```
time less than 90.97525473071326
```

Problem 3

Problem 3



Boundary Conditions

- ① $\Theta(x=w, y) = 0$
- ② $-k \frac{\partial \Theta}{\partial y} \Big|_{y=b} = -q''_m(x)$
- ③ $-k \frac{\partial \Theta}{\partial y} \Big|_{y=0} + h \Theta(x, y=0) = 0$
- ④ $-k \frac{\partial \Theta}{\partial x} \Big|_{x=0} = 0$

$$\Theta(x, y) = \bar{X}(x) Y(y)$$

$$\frac{d^2 \Theta}{dx^2} + \frac{d^2 \Theta}{dy^2} = 0 \rightarrow \frac{d^2(\bar{X}Y)}{dx^2} + \frac{d^2(\bar{X}Y)}{dy^2} = 0 \rightarrow Y \frac{d^2 \bar{X}}{dx^2} + \bar{X} \frac{d^2 Y}{dy^2} = 0$$

$$\left. \begin{array}{l} \bar{X} = C_1 \cos \lambda x + C_2 \sin \lambda x \\ Y = C_3 e^{\lambda y} + C_4 e^{-\lambda y} \end{array} \right\} \Theta(x, y) = [C_1 \cos \lambda x + C_2 \sin \lambda x] [C_3 e^{\lambda y} + C_4 e^{-\lambda y}]$$

BC 4

$$-k \frac{\partial \Theta}{\partial x} \Big|_{x=0} = 0 \rightarrow 0 = \frac{\partial(\bar{X}Y)}{\partial x} \rightarrow 0 = Y \frac{\partial \bar{X}}{\partial x} \rightarrow 0 = \frac{\partial}{\partial x} (C_1 \cos \lambda x + C_2 \sin \lambda x)$$

$$0 = -C_1 \lambda \underbrace{\sin(\lambda \cdot 0)}_0 + C_2 \lambda \underbrace{\cos(\lambda \cdot 0)}_1$$

$$C_2 = 0$$

BC 3

$$-k \frac{\partial \Theta}{\partial y} \Big|_{y=0} + h \Theta = 0$$

$$-k \bar{X} \frac{\partial Y}{\partial y} + h \bar{X} Y = 0$$

$$[-k (C_1 \cos \lambda x) (C_3 \lambda e^{\lambda y} - C_4 \lambda e^{-\lambda y}) + h (C_1 \cos \lambda x) (C_3 e^{\lambda y} + C_4 e^{-\lambda y})]_{y=0} = 0$$

$$-k (C_1 \cos \lambda x) (C_3 \lambda \underbrace{e^{\lambda(0)}}_1 - C_4 \lambda \underbrace{e^{-\lambda(0)}}_1) + h (C_1 \cos \lambda x) (C_3 \underbrace{e^{\lambda(0)}}_1 + C_4 \underbrace{e^{-\lambda(0)}}_1) = 0$$

$$(C_1 \cos \lambda x) (-k (C_3 \lambda - C_4 \lambda) + h (C_3 + C_4)) = 0$$

$$-k C_3 \lambda + k C_4 \lambda + h C_3 + h C_4 = 0$$

$$-k (C_3 \lambda - C_4 \lambda) + h (C_3 + C_4) = 0$$

$$C_4 (k \lambda + h) - C_3 (+k \lambda - h) = 0$$

$$C_4 = \frac{C_3 (-h + k \lambda)}{(h + k \lambda)}$$

BC1

$$\Theta(x=w, y) = 0 \rightarrow [C_1 \cos(\lambda \cdot w)] [C_3 e^{\lambda y} + \left(\frac{C_3(h+k\lambda)}{(h+k\lambda)} \right) e^{-\lambda y}] = 0 \quad C_3 \neq 0 \\ C_1 \neq 0$$

$$\cos(\lambda \cdot w) = 0 \rightarrow \lambda w = \frac{\pi}{2} + n\pi \quad \text{for } n=0, 1, 2, \dots \quad (2n+1)\pi$$

$$\Theta(x, y) = \sum_{n=0}^{\infty} A_n \cos(\lambda_n x) \left(e^{\lambda_n y} + \frac{(h+k\lambda_n)}{(h+k\lambda_n)} e^{-\lambda_n y} \right) \quad \lambda_n = \frac{(2n+1)\pi}{2w}$$

BC 2

$$-k \frac{\partial \Theta}{\partial y} \Big|_{y=b} = -q''_m(x)$$

$$+ k \frac{\partial}{\partial y} \left(\sum_{n=0}^{\infty} A_n \cos(\lambda_n x) \left(e^{\lambda_n y} + \frac{(h+k\lambda_n)}{(h+k\lambda_n)} e^{-\lambda_n y} \right) \right) \Big|_{y=b} = +q''_m(x)$$

$$k \left(\sum_{n=0}^{\infty} A_n \cos(\lambda_n x) \frac{\partial}{\partial y} \left(e^{\lambda_n y} + \frac{(h+k\lambda_n)}{(h+k\lambda_n)} e^{-\lambda_n y} \right) \right) \Big|_{y=b} = q''_m(x)$$

$$k \sum_{n=0}^{\infty} A_n \cos(\lambda_n x) \left[\lambda_n e^{\lambda_n b} + \lambda_n \frac{(h+k\lambda_n)}{(h+k\lambda_n)} e^{-\lambda_n b} \right] = q''_m(x)$$

$$\sum_{n=0}^{\infty} A_n \left[k \cos(\lambda_n x) (\lambda_n e^{\lambda_n b} - \lambda_n \frac{(h+k\lambda_n)}{(h+k\lambda_n)} e^{-\lambda_n b}) \right] [\cos(\lambda_m x)] = q''_m(x) \cos(\lambda_m x)$$

$$\sum_{n=0}^{\infty} A_n \left(\lambda_n e^{\lambda_n b} - \lambda_n \frac{(h+k\lambda_n)}{(h+k\lambda_n)} e^{-\lambda_n b} \right) k \left[\int_0^{L_j} \cos \lambda_n x \cos \lambda_m x + \int_{L_j}^W \cos \lambda_n x \cosh \lambda_m x \right] dx =$$

$$\left[\int_0^{L_j} q''_j \exp\left(-\frac{x}{L_j}\right) \cos \lambda_m x + \int_{L_j}^W q''_j \exp\left(\frac{x}{L_j}\right) \cos \lambda_m x \right] dx \xrightarrow{\text{orthogonality}} \text{Only } \lambda_m = \lambda_n \text{ term survives}$$

$$A_n \left(\lambda_n e^{\lambda_n b} - \lambda_n \frac{(h+k\lambda_n)}{(h+k\lambda_n)} e^{-\lambda_n b} \right) k \left[\int_0^{L_j} \cos^2(\lambda_n x) + \int_{L_j}^W \cos^2(\lambda_n x) \right] dx = \left[\int_0^{L_j} q''_j \exp\left(-\frac{x}{L_j}\right) \cos \lambda_n x \right] dx$$

$$A_n = \frac{\int_0^{L_j} q''_j \exp\left(-\frac{x}{L_j}\right) \cos(\lambda_n x) dx}{k \left(\lambda_n e^{\lambda_n b} - \lambda_n \frac{(h+k\lambda_n)}{(h+k\lambda_n)} e^{-\lambda_n b} \right) \left(\int_0^W \cos^2(\lambda_n x) dx \right)} \quad \lambda_n = \frac{(2n+1)\pi}{2w}$$

$$\Theta(x, y) = \sum_{n=0}^{\infty} A_n \cos(\lambda_n x) \left(e^{\lambda_n y} + \frac{(h+k\lambda_n)}{(h+k\lambda_n)} e^{-\lambda_n y} \right)$$

$\int_0^W \cos^2(\lambda_n x) dx = \frac{\sin(2\lambda_n W)}{4\lambda_n} + \frac{W}{2} \approx 0$

In [34]:

```
params = {'legend.fontsize': 'xx-large',
          'figure.figsize': (15, 10),
          'axes.labelsize': 'xx-large',
          'axes.titlesize': 'xx-large',
          'xtick.labelsize': 'xx-large',
          'ytick.labelsize': 'xx-large'}
import matplotlib.pyplot as plt
```

```

plt.rcParams.update(params)

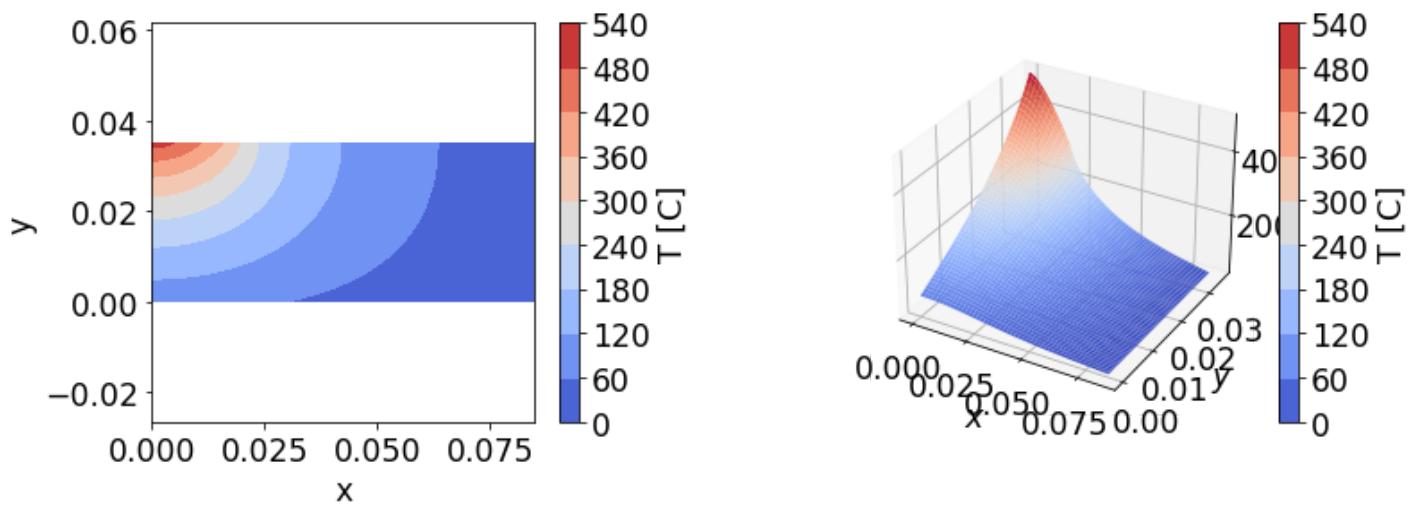
#Numerical Math
import math # math functions
import numpy as np # numerical math
import scipy
from numpy import *
from scipy import integrate
from scipy.linalg import solve

# Widgets
import ipywidgets as widgets
from IPython.display import display
from ipywidgets import interact
N=100
Nr=100
W=0.085
b=0.035
k=38
Ts=25+273.15
q_dprime_j=10**6
Lj=0.02
h=5000
x = np.linspace(0,W,num=N)
y = np.linspace(0,b,num=N)
xx,yy = meshgrid(x,y) # Set up the grid for plotting
T2 = np.zeros([N,N]) #Set up the temperature array for Theta = normalized T
A = zeros(Nr)
for n in range(0,Nr): # interate from n = 1 ... inf, where we approximate inf with a Large number
    # un at one value of n we have to sum up the solutions from n = 1 to ~inf
    l = (2*n+1)*pi/(2*W) #calcualte the nth root
    An,err = scipy.integrate.quad(lambda xp: q_dprime_j*exp(-xp/Lj)*cos(l*xp),0,Lj)
    Ad1 = k*(l*exp(l*b)-l*((-h+k*l)/(h+k*l))*exp(-l*b))
    Ad2 = W/2
    A = An/(Ad1*Ad2)
    un = A*cos(l*xx)*(exp(l*yy)+((-h+k*l)/(h+k*l))*exp(-l*yy))
    T2 = T2 + un
T=(T2+Ts)-273.15
#Plot temperature-----
fig = plt.figure(figsize=plt.figaspect(0.5))
ax = fig.add_subplot(2, 2, 1)
fig.subplots_adjust(wspace=0.5)
fig.set_figheight(9)
fig.set_figwidth(12)

ct = ax.contourf(xx,yy,T,cmap='coolwarm')
ax.axis('equal');
cbar = fig.colorbar(ct)
cbar.set_label('T [C]', fontsize = 18)
ax.set_xlabel('x')
ax.set_ylabel('y')

ax = fig.add_subplot(2, 2, 2, projection='3d')
ax.plot_surface(xx, yy, T, cmap='coolwarm')
ax.set(xlabel='$x$', ylabel='$y$')
cbar = fig.colorbar(ct)
cbar.set_label('T [C]', fontsize = 18)

```



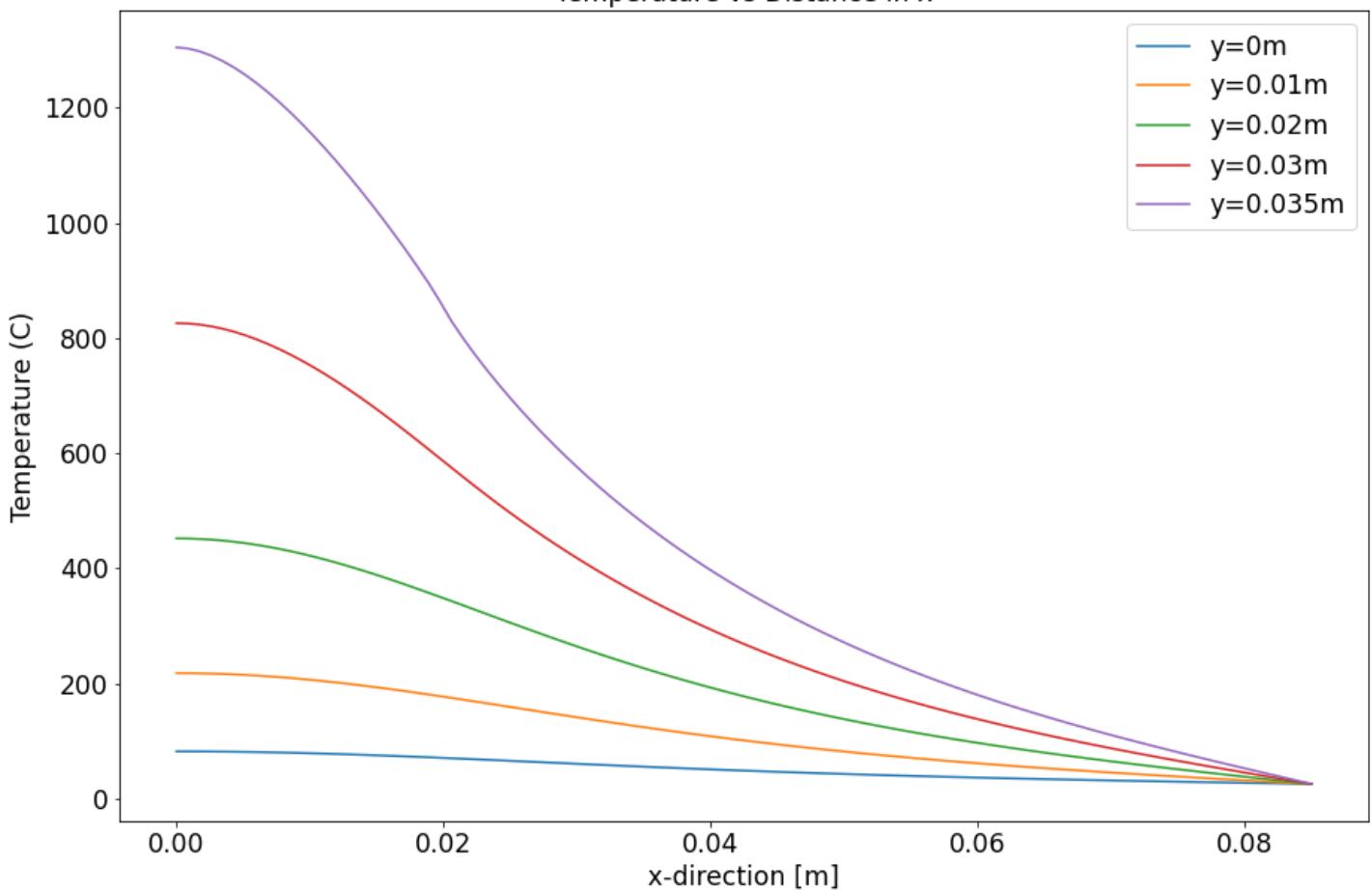
In [35]:

```
Ns=100
Nr=100
x=np.linspace(0,W, num=Ns)
y=[0, 0.01, 0.02, 0.03, 0.035]
T2=np.zeros(Ns)
Theta=[None, None, None, None, None]
for i in range(size(Theta)):
    for n in range(0, Nr):
        l = (2*n+1)*pi/(2*W) #calcualte the nth root
        An,err = scipy.integrate.quad(lambda xp: q_dprime_j*exp(-xp/Lj)*cos(l*xp),0,Lj)
        Ad1 = k*(l*exp((l*b))+l*((h-k*l)/(h+k*l))*exp((-l*b)))
        Ad2 = W/2
        A = An/(Ad1*Ad2)
        un = A*cos(l*x)*(exp(l*y[i])-((h-k*l)/(h+k*l))*exp(-l*y[i]))
        T2 = T2 + un
    Theta[i]=(T2+Ts)-273.15
```

In [36]:

```
for i in range(5):
    label=str("y=" +str(y[i])+'m')
    plt.plot(x, Theta[i], label=label)
    plt.xlabel('x-direction [m]')
    plt.ylabel('Temperature (C)')
    plt.title('Temperature vs Distance in x')
plt.legend()
plt.show()
```

Temperature vs Distance in x

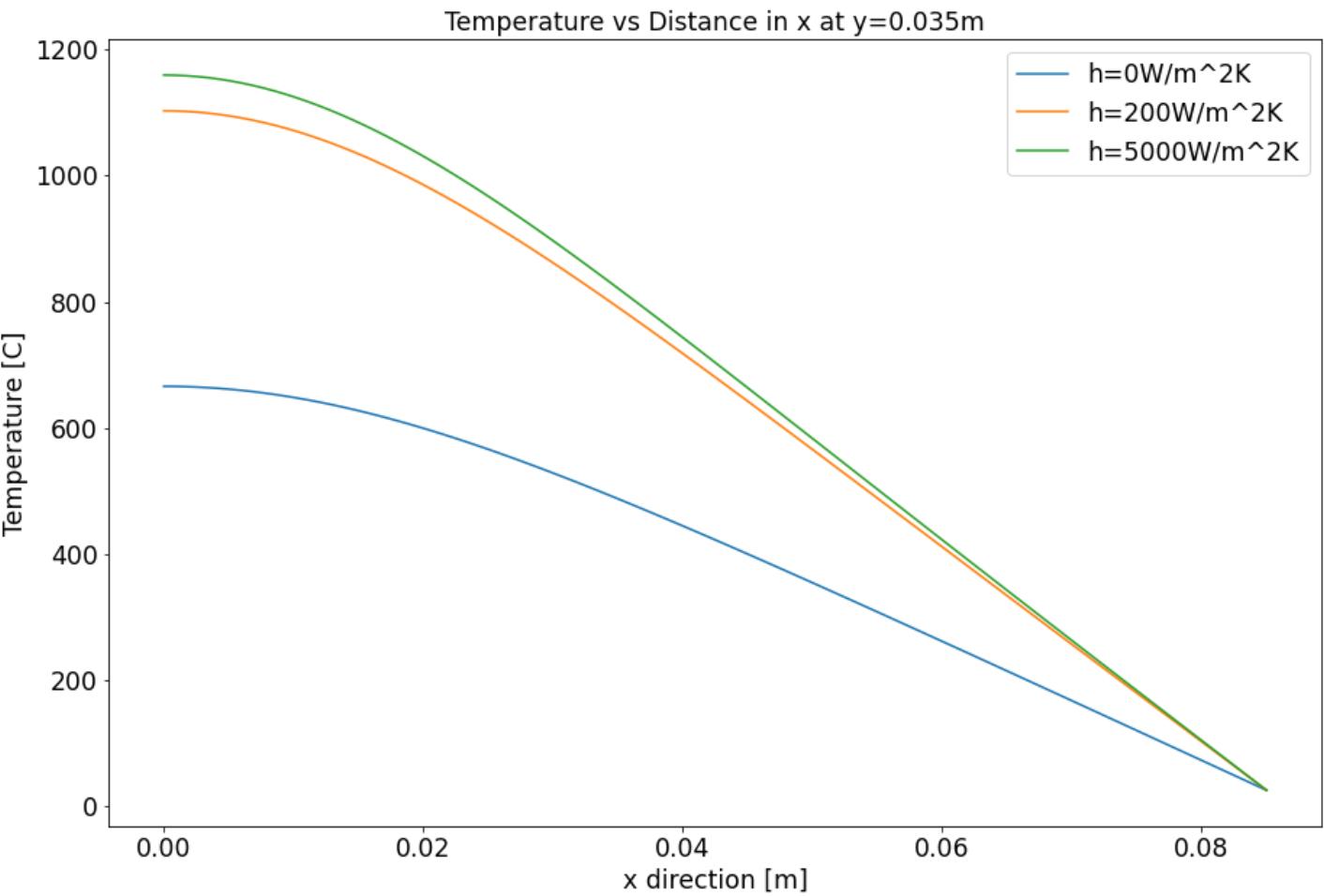


In [37]:

```

hy=[0,200,5000]
y=[0,0.035]
T2=np.zeros(Ns)
Thetay=[None, None, None]
Thetay0=[None, None, None]
for i in range(size(hy)):
    for n in range(0, Nr):
        l = (2*n+1)*pi/(2*W) #calcualte the nth root
        An,err = scipy.integrate.quad(lambda xp: q_dprime_j*exp(-xp/Lj)*cos(l*xp),0,Lj)
        Ad1 = k*(l*exp((l*b))+l*((hy[i]-k*l)/(hy[i]+k*l))*exp((-l*b)))
        Ad2 = W/2
        A = An/(Ad1*Ad2)
        un = A*cos(l*x)*(exp(l*y[0])-((hy[i]-k*l)/(hy[i]+k*l))*exp(-l*y[0]))
        T2 = T2 + un
    Thetay0[i]=(T2+Ts)-273.15
for i in range(3):
    label=str("h=" +str(hy[i])+'W/m^2K')
    plt.plot(x, Thetay0[i], label=label)
plt.xlabel('x direction [m]')
plt.ylabel('Temperature [C]')
plt.title('Temperature vs Distance in x at y=0.035m')
plt.legend()
plt.show()

```

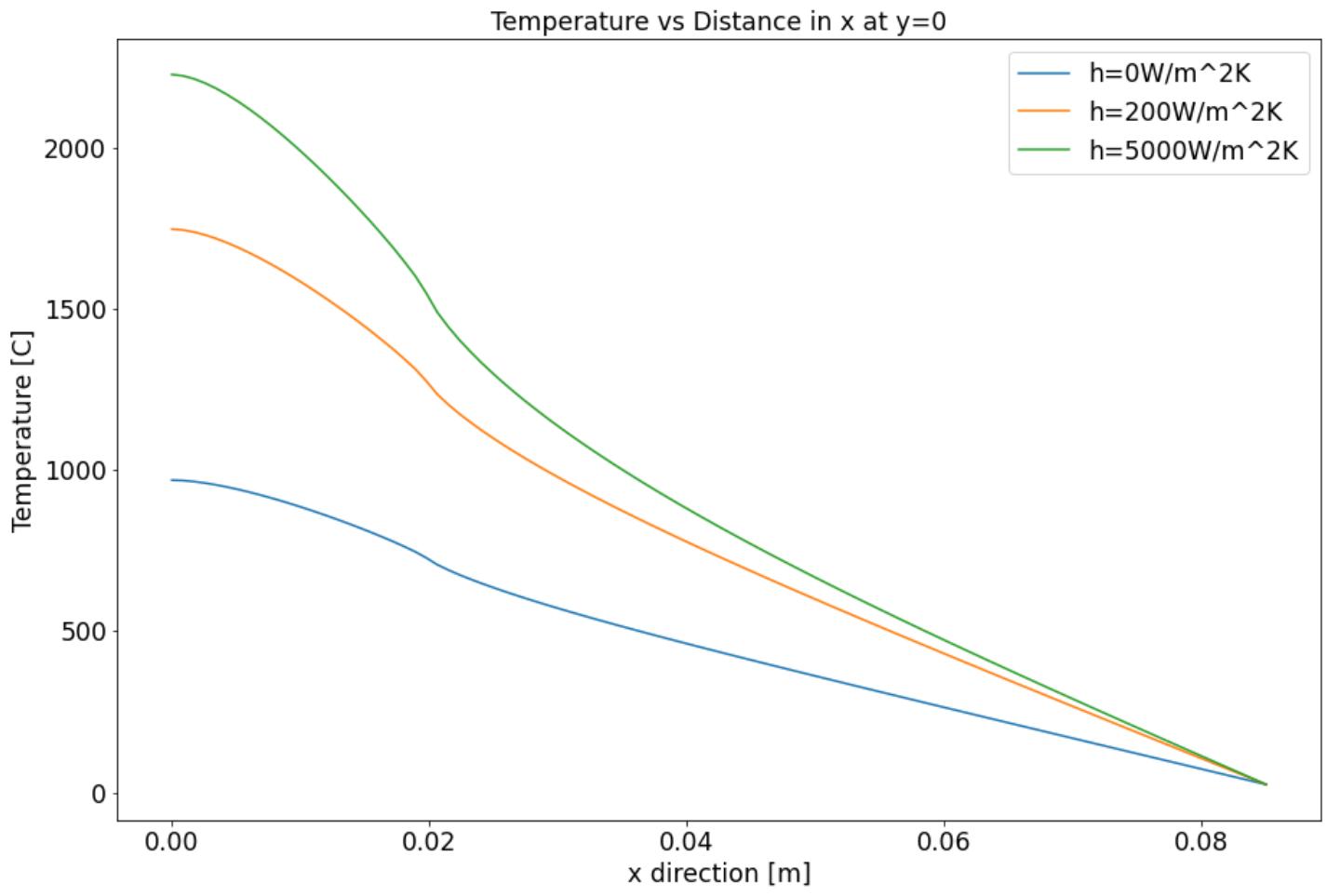


In [38]:

```

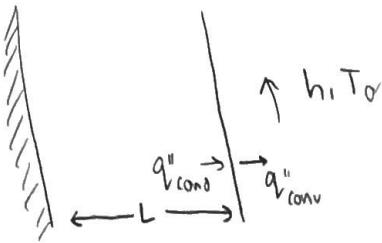
hy=[0,200,5000]
y=[0,0.035]
T2=np.zeros(Ns)
Thetay=[None, None, None]
Thetay35=[None, None, None]
for i in range(size(hy)):
    for n in range(0, Nr):
        l = (2*n+1)*pi/(2*W) #calcualte the nth root
        An,err = scipy.integrate.quad(lambda xp: q_dprime_j*exp(-xp/Lj)*cos(l*xp),0,Lj)
        Ad1 = k*(l*exp((l*b))+l*((hy[i]-k*l)/(hy[i]+k*l))*exp((-l*b)))
        Ad2 = W/2
        A = An/(Ad1*Ad2)
        un = A*cos(l*x)*(exp(l*y[1])-((hy[i]-k*l)/(hy[i]+k*l))*exp(-l*y[1]))
        T2 = T2 + un
    Thetay35[i]=(T2+Ts)-273.15
for i in range(3):
    label=str("h=" +str(hy[i])+'W/m^2K')
    plt.plot(x, Thetay35[i], label=label)
    plt.xlabel('x direction [m]')
    plt.ylabel('Temperature [C]')
    plt.title('Temperature vs Distance in x at y=0')
plt.legend()
plt.show()

```



Problem 4

Problem 4



$$\textcircled{1} \quad E_{in} - E_{out} = 0 \Rightarrow q''_{cond} = q''_{conv} \Rightarrow -k \frac{dT}{dx} \Big|_{x=L} = h (T(x=L) - T_0)$$

As $t \rightarrow \infty$ $\frac{dT}{dx} \rightarrow 0$ b/c of adiabatic condition at $x=0$
So $T(x=L) \rightarrow T_0$ and the entire slab is at the constant temperature

$$\textcircled{2} \quad \frac{\partial}{\partial x} \left(k \frac{\partial T}{\partial x} \right) + \frac{\partial}{\partial y} \left(k \frac{\partial T}{\partial y} \right) + \frac{\partial}{\partial z} \left(k \frac{\partial T}{\partial z} \right) + \dot{q}' = \rho C_p \frac{\partial T}{\partial t}$$

ID NO \dot{q}'

$$\Theta = T(x,t) - T_{ss}$$

$$\frac{\partial^2 T}{\partial x^2} = \frac{\partial^2 \Theta}{\partial x^2} \quad \text{and} \quad \frac{\partial T}{\partial t} = \frac{\partial \Theta}{\partial t} \quad \frac{\rho C_p}{k} = \frac{1}{\alpha}$$

$$\frac{\partial^2 \Theta}{\partial x^2} = \frac{1}{\alpha} \frac{\partial \Theta}{\partial t}.$$

B.C.

$$-k \frac{\partial T}{\partial x} \Big|_{x=L} = h (T(x=L) - T_0)$$

$$-k \frac{\partial T}{\partial x} \Big|_{x=0} = 0$$

$$\frac{\partial \Theta}{\partial x} \Big|_{x=L} = -\frac{h}{k} \Theta(x=L, t)$$

$$\frac{\partial \Theta}{\partial x} \Big|_{x=0} = 0$$

$$\Theta(x, t=0) = T(x, t=0) - T_{ss} = \bar{T}_i - T_{ss} = \Delta T_i$$

3

$$\Theta(x,t) = \bar{X}(x) \cdot \Gamma(t)$$

$$\frac{\partial^2 (\bar{X}\Gamma)}{\partial x^2} = \frac{1}{\alpha} \frac{\partial (\bar{X}\Gamma)}{\partial t} \rightarrow \Gamma \frac{\partial^2 \bar{X}}{\partial x^2} = \frac{\bar{X}}{\alpha} \frac{d\Gamma}{dt}$$

$$\text{Let } \bar{X}(x) = C_1 \cos(\lambda x) + C_2 \sin(\lambda x)$$

$$\Gamma(t) = C_3 e^{-\alpha \lambda^2 t}$$

Boundary Conditions

$$(1) \quad \left. \frac{\partial \Theta}{\partial x} \right|_{x=0} = 0 \Rightarrow \left. \frac{\partial (\bar{X}\Gamma)}{\partial x} \right|_{x=0} = C_1 \lambda \sin(\lambda t_0) + C_2 \lambda \cos(\lambda t_0) = 0 \quad C_2 = 0$$

(2)

$$\left. \frac{\partial \Theta}{\partial x} \right|_{x=L} = -\frac{h}{k} \Theta(x=L) \Gamma(t) = \left. \frac{d\bar{X}}{dx} \right|_{x=L} = -\frac{h}{k} \bar{X}(x=L)$$

$$\therefore C_1 \lambda \sin(\lambda L) = -\frac{h}{k} C_1 \cos(\lambda L)$$

$$C_1 \frac{\lambda k}{h} \sin(\lambda L) = C_1 \cos(\lambda L) \rightarrow C_1 \left[\frac{\lambda k}{h} \sin(\lambda L) - \cos(\lambda L) \right] = 0$$

$$\lambda + \tan(\lambda L) - \frac{h}{ik} = 0 \quad \text{periodic function w/ infinite roots}$$

$$\Theta(x,t) = \sum_{n=1}^{\infty} C_n C_3 \cos(\lambda_n x) \exp(-\alpha \lambda_n^2 t)$$

$$\Theta(x,t=0) = \sum_{n=1}^{\infty} A \cos(\lambda_n x) \exp(-\alpha \lambda_n^2 t_0) = \sum_{n=1}^{\infty} A \cos(\lambda_n x) = \Delta T_i$$

$$\int_0^L \sum_{n=1}^{\infty} A \cos(\lambda_n x) \cos(\lambda_m x) dx = \int_0^L \Delta T_i \cos(\lambda_m x) dx$$

$$A \int_0^L \cos^2(\lambda_n x) dx = \Delta T_i \int_0^L \cos(\lambda_n x) dx$$

$$A = \frac{\Delta T_i \int_0^L \cos(\lambda_n x) dx}{\int_0^L \cos^2(\lambda_n x) dx}$$

$$T(x,t) = \sum_{n=1}^{\infty} A \cos(\lambda_n x) \exp(-\alpha \lambda_n^2 t) + T_{ss}$$

In [1]:

L=0.025

Ti=20

h=1000

T0=100

rho=2500

cp=1000

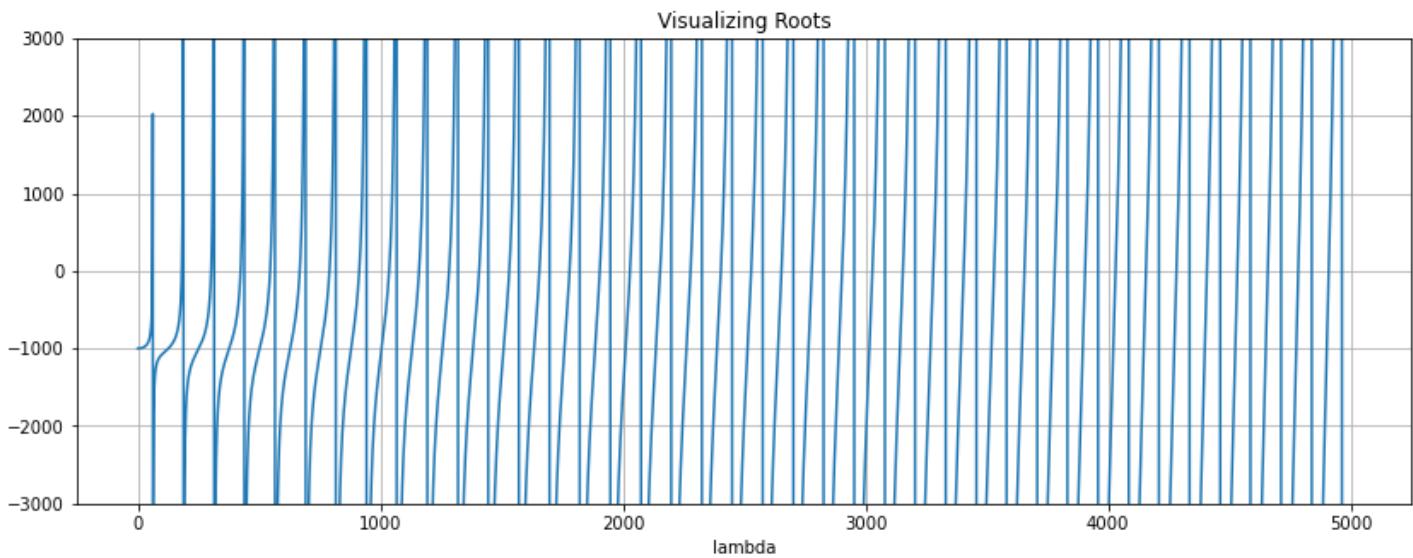
k=1

```
alpha=k/(rho*cp)
Tss=T0
TDelta=Ti-Tss
```

In [6]:

```
import matplotlib.pyplot as plt
#Numerical Math
import math # math functions
import numpy as np # numerical math
import scipy
from numpy import *
from scipy import integrate
from scipy.linalg import solve

# Widgets
import ipywidgets as widgets
from IPython.display import display
from ipywidgets import interact
l=np.linspace(0,5000,5000)
def f(l):
    return l*tan(L*l)-(h/k)
plt.figure(figsize=(14,5))
plt.plot(l, f(l))
plt.ylim(-3000, 3000)
plt.grid()
plt.title('Visualizing Roots')
plt.xlabel('lambda')
plt.show()
```



In [7]:

```
zx=np.where(np.diff(np.sign(f(l))))[0]
zx=zx[::2]
lambdas=np.zeros(len(zx))
for i in range(len(zx)):
    lambdas[i]=scipy.optimize.brentq(f, l[zx[i]],l[zx[i]+1])
print(lambdas)
```

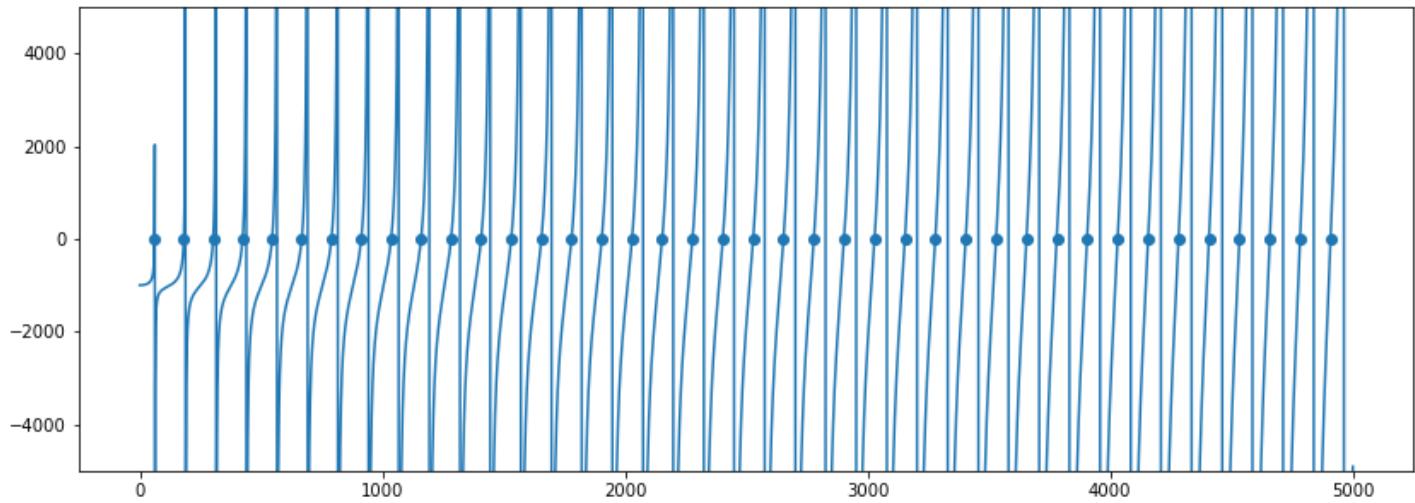
```
[ 60.41806468 181.32068402 302.41251964 423.78918628 545.51106639
 667.60432496 790.06787556 912.88231665 1036.01803233 1159.44125141
1283.11798177 1407.01624665 1531.10715076 1655.36522149 1779.76834291
1904.29748687 2028.93636299 2153.67105517 2278.48967871 2403.38207325
2528.33953583 2653.35459293 2778.42080797 2903.53261989 3028.68520808
3153.87437973 3279.09647568 3404.34829176 3529.62701301 3654.93015834
3780.25553412 3905.60119494 4030.96541042 4156.3466371 4281.74349443
4407.1547443 4532.57927356 4658.01607889 4783.46425388 4908.92297778]
```

In [9]:

```
plt.figure(figsize=(14,5))
plt.plot(l, f(l))
```

```
plt.scatter(lambdas, f(lambdas))
plt.ylim(-5000,5000)
```

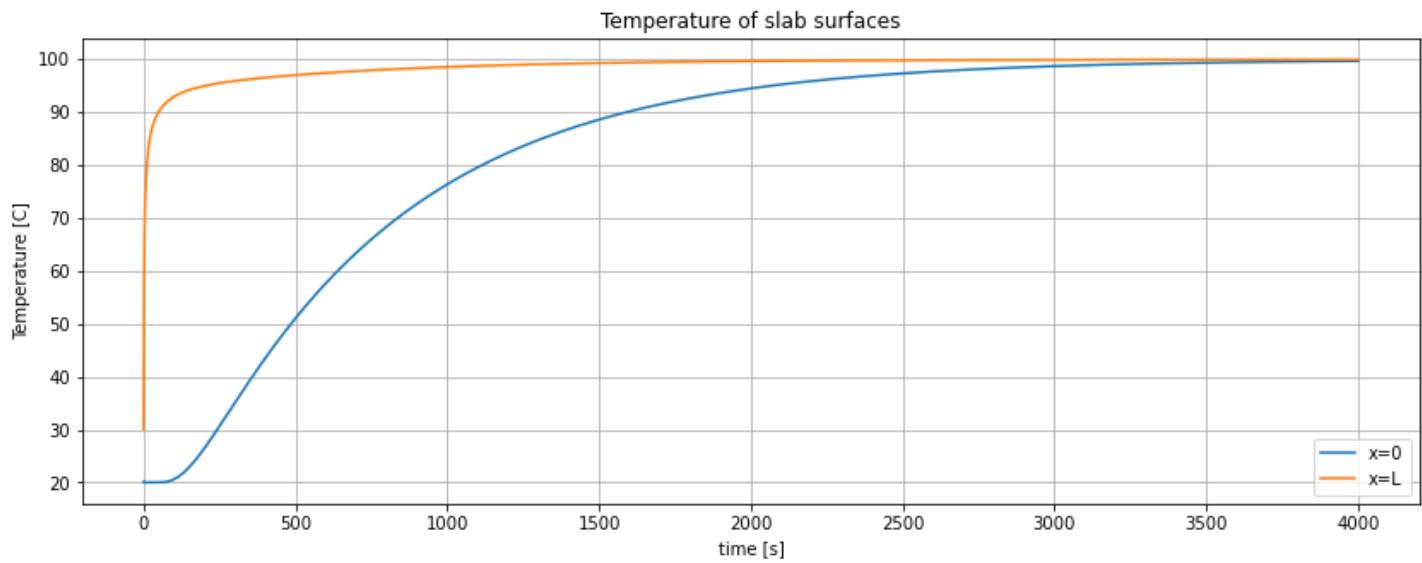
Out[9]: (-5000.0, 5000.0)



In [11]:

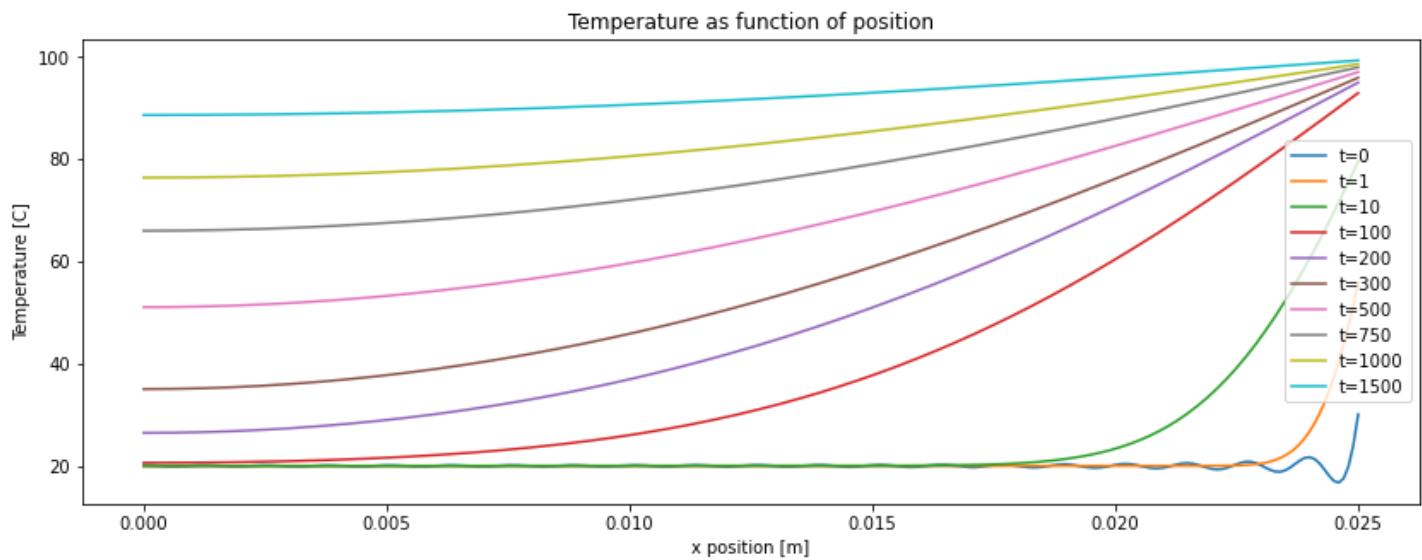
```
Nx=251
Nt=4000
x=np.linspace(0,L,num=Nx)
t=np.linspace(0, Nt, num=Nt)
xx, tt=meshgrid(x,t)
theta=np.zeros([len(t),len(x)])
A=np.zeros(len(lambdas))
for n in range(len(lambdas)):
    l=lambdas[n]
    AnumIntegral, err1=scipy.integrate.quad(lambda x: cos(l*x), 0, L)
    Anum=TDelta*AnumIntegral
    Aden, err2=scipy.integrate.quad(lambda x: (cos(l*x))**2, 0,L)
    A[n]=Anum/Aden
    xterm=cos(l*xx)
    tterm=exp(-alpha*l**2*tt)
    thetan=A[n]*xterm*tterm
    theta=theta+thetan
T=theta+Tss*np.ones([len(t), len(x)])

plt.figure(figsize=(14,5))
plt.plot(t, T[:,0], label='x=0')
plt.plot(t, T[:, -1], label='x=L')
plt.title('Temperature of slab surfaces')
plt.xlabel('time [s]')
plt.ylabel('Temperature [C]')
plt.legend(loc='best')
plt.grid()
plt.show()
```



In [18]:

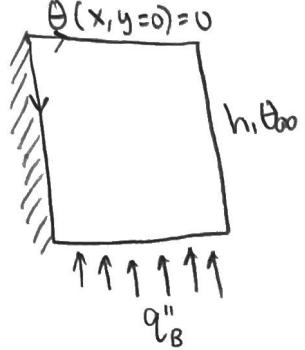
```
plt.figure(figsize=(14,5))
time_slice=[0,1,10,100,200,300,500,750,1000,1500]
for i in time_slice:
    plt.plot(x, T[i, :], label= f't={t[i]:1.0f}')
plt.title('Temperature as function of position')
plt.xlabel('x position [m]')
plt.ylabel('Temperature [C]')
plt.legend(loc='best')
plt.show()
```



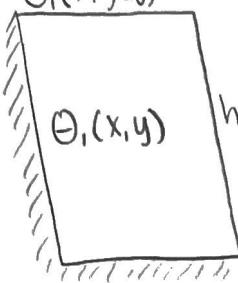
Problem 5

Problem 5

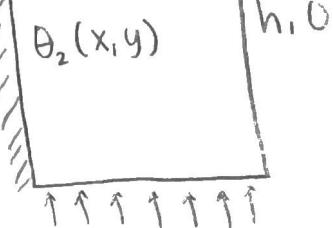
$$\Theta = T - T_{\infty}$$



$$\Theta_1(x,y=0) = 0$$



$$\Theta_2(x,y=0) = 0$$



Boundary Conditions

$$\Theta(x,y=0) = 0$$

$$-k \frac{\partial \Theta}{\partial y} \Big|_{y=H} = -q''_B$$

$$-k \frac{\partial \Theta}{\partial x} \Big|_{x=\frac{w}{2}} + h\Theta(\frac{w}{2},y) = h\Theta_\infty$$

$$-k \frac{\partial \Theta}{\partial x} \Big|_{x=0} = 0$$

$$\Theta_1(x,y=0) = 0$$

$$-k \frac{\partial \Theta_1}{\partial y} \Big|_{y=H} = 0$$

$$-k \frac{\partial \Theta_1}{\partial x} \Big|_{x=\frac{w}{2}} + h\Theta(\frac{w}{2},y) = h\Theta_\infty$$

$$-k \frac{\partial \Theta_1}{\partial x} \Big|_{x=0} = 0$$

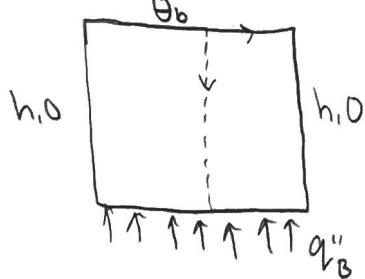
$$\Theta_2(x,y=0) = 0$$

$$-k \frac{\partial \Theta_2}{\partial y} \Big|_{y=H} = -q''_B$$

$$-k \frac{\partial \Theta_2}{\partial x} \Big|_{x=\frac{w}{2}} + h\Theta(\frac{w}{2},y) = 0$$

$$-k \frac{\partial \Theta_2}{\partial x} \Big|_{x=0} = 0$$

$$\Theta = T - T_\infty$$



$$\Theta_B$$

$$\Theta_B = 0$$

$$\Theta(x,y=0) = \Theta_B$$

$$-k \frac{\partial \Theta}{\partial y} \Big|_{y=H} = -q''_B$$

$$-k \frac{\partial \Theta}{\partial x} \Big|_{x=\frac{w}{2}} + h\Theta = 0$$

$$-k \frac{\partial \Theta}{\partial x} \Big|_{x=-\frac{w}{2}} + h\Theta = 0$$

$$-k \frac{\partial \Theta}{\partial x} \Big|_{x=0} = 0$$

$$-k \frac{\partial \Theta_1}{\partial y} \Big|_{y=H} = 0$$

$$-k \frac{\partial \Theta_1}{\partial x} \Big|_{x=\frac{w}{2}} + h\Theta(\frac{w}{2},y) = 0$$

$$-k \frac{\partial \Theta_1}{\partial x} \Big|_{x=-\frac{w}{2}} + h\Theta_1(\frac{w}{2},y) = 0$$

$$-k \frac{\partial \Theta_1}{\partial x} \Big|_{x=0} = 0$$

$$-k \frac{\partial \Theta_2}{\partial y} \Big|_{y=H} = -q''_B$$

$$-k \frac{\partial \Theta_2}{\partial x} \Big|_{x=\frac{w}{2}} + h\Theta_2(\frac{w}{2},y) = 0$$

$$-k \frac{\partial \Theta_2}{\partial x} \Big|_{x=-\frac{w}{2}} + h\Theta_2(\frac{w}{2},y) = 0$$

$$-k \frac{\partial \Theta_2}{\partial x} \Big|_{x=0} = 0$$

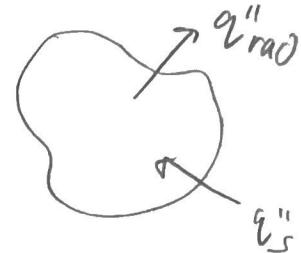
Problem 6

Problem 6

$$\frac{1}{1} h_r = \varepsilon \sigma (T_s + T_{sur}) (T_s^2 + T_{sur}^2) = 0.712 \frac{W}{m^2 K}$$

2

$$\dot{E}_in - \dot{E}_{out} + \dot{E}_{gen} = \dot{E}_{st}$$



$$q''_s A_s - q''_{rad} A_s = M C_p \frac{\partial T}{\partial t}$$

$$(\bar{q''}_s A_s + \Delta q''_s \sin(\omega t)) A_s - h_r (T - T_{sur}) A_s = M C_p \frac{\partial T}{\partial t}$$

$$\boxed{\varepsilon \bar{q''}_s A_s + \Delta q''_s \varepsilon \sin(\omega t) - h_r T(t) A_s + h_r A_s T_{sur} - M C_p \frac{\partial T}{\partial t} = 0}$$

$$\underbrace{\varepsilon \bar{q''}_s A_s}_e + \underbrace{\Delta q''_s \varepsilon \sin(\omega t)}_c + \underbrace{h_r A_s T_{sur}}_d = \underbrace{M C_p \frac{\partial T}{\partial t}}_a + \underbrace{h_r A_s T(t)}_b$$

$$a \frac{\partial T}{\partial t} + b T(t) = c \sin(\omega t)$$

ignore e + d

$$T(t) = A \sin(\omega t + \phi)$$

$$\cos(\omega t + \phi) = \sin(\omega t + \phi + \frac{\pi}{2})$$

$$\frac{\partial T}{\partial t} = A \omega \cos(\omega t + \phi)$$

$$\sin(A + B) = \sin A \cos B + \cos A \sin B$$

$$a \omega A \cos(\omega t + \phi) + b A \sin(\omega t + \phi) = c \sin(\omega t)$$

$$A(a \omega \cos(\omega t + \phi) + b \sin(\omega t + \phi)) = c \sin(\omega t)$$

$$A(a \omega \sin(\omega t + \phi + \frac{\pi}{2}) + b \cos(\omega t + \phi)) = c \sin(\omega t)$$

$$A(a \omega \sin(\omega t) \cos(\phi + \frac{\pi}{2}) + a \omega \cos(\omega t) \sin(\phi + \frac{\pi}{2}) + b \sin(\omega t) \cos(\phi) + b \cos(\omega t) \sin(\phi)) = c \sin(\omega t)$$

$$A \left(\sin(\omega t) \left(\frac{a \omega \cos(\phi + \frac{\pi}{2}) + b \cos(\phi)}{b \cos(\phi) - a \omega \sin(\phi)} \right) + \cos(\omega t) \left(\frac{a \omega \sin(\phi + \frac{\pi}{2}) + b \sin(\phi)}{a \omega \cos(\phi) + b \sin(\phi)} \right) \right) = c \sin(\omega t)$$

$$AA \left((a \omega \cos(\phi) + b \sin(\phi))^2 + (b \cos(\phi) - a \omega \sin(\phi))^2 \right)^{1/2} \sin(\omega t + \phi) = c \sin(\omega t)$$

$$\phi_p = \tan^{-1} \left(\frac{a \omega \cos(\phi) + b \sin(\phi)}{b \cos(\phi) - a \omega \sin(\phi)} \right) = 0$$

$$A((a \omega)^2 \cos^2(\phi) + 2 a b \omega \sin(\phi) \cos(\phi) + b^2 \sin^2(\phi) + b^2 \cos^2(\phi) - 2 a \omega b \sin(\phi) \cos(\phi) + a^2 \omega^2 \sin^2(\phi))^{1/2} \sin(\omega t + \phi) = c \sin(\omega t)$$

$$A(a^2 \omega^2 + b^2)^{1/2} \sin(\omega t) = c \sin(\omega t)$$

$$A = \frac{c}{a^2 \omega^2 + b^2}$$

$$\tan(\phi) = \frac{a\omega \cos \phi + b \sin \phi}{b \cos \phi - a \omega \sin \phi} = 0$$

$$0 = a \omega \cos \phi + b \sin \phi$$

$$b \sin \phi = -a \omega \cos \phi$$

$$\tan \phi = -\frac{a \omega}{b}$$

$$\phi = \tan^{-1} \left(-\frac{a \omega}{b} \right)$$

$$T_{\text{sensor}} = A \sin(\omega t + \phi)$$

$$A = \frac{c}{(a^2 \omega^2 + b^2)^{1/2}}$$

$$a = M C_p$$

$$\phi = \tan^{-1} \left(-\frac{a \omega}{b} \right)$$

$$b = h_r A_s$$

$$c = \Delta q_s'' A_s \epsilon$$

In [19]:

```
epsilon=0.35
sigma=5.67*10**-8
T=75
T_surr=300
hr=epsilon*sigma*(T+T_surr)*(T**2+T_surr**2)
print(hr)
```

0.7116292968749999

In [44]:

```
T_surr = 300 # [K] surroudnings temperature
epsilon = 0.35      # [-] Estimate for emissivity
sigma=5.67*10**-8
M=0.05
As=0.02
omega=0.02094
cp=300
delq_dprime_s=100
a=M*cp
b=hr*As
c=delq_dprime_s*As*epsilon
A=c/((a**2)*(omega**2)+(b**2))**0.5
phi=math.atan(-a*omega/b)
print('Amplitude= ', A)
```

Amplitude= 2.2263052654764186

In [40]:

```
#@title
# import a few more things
from scipy.integrate import odeint
from scipy.constants import Stefan_Boltzmann as sigma    # Stefan-Boltzmann Constant [W/m^2-K^4]
import math # math functions
import numpy as np # numerical math
from numpy import *
from scipy.linalg import solve
from scipy.constants import pi

# Widgets
import ipywidgets as widgets
from IPython.display import display
from ipywidgets import interact

# plotting
params = {'legend.fontsize': 'xx-large',
           'figure.figsize': (15, 10),
           'axes.labelsize': 'xx-large',
           'axes.titlesize':'xx-large',
           'xtick.labelsize':'xx-large',
           'ytick.labelsize':'xx-large'}
import matplotlib.pyplot as plt
plt.rcParams.update(params)
# Additional Parameters
T_surr = 300 # [K] surroudnings temperature
epsilon = 0.35      # [-] Estimate for emissivity
t_max=5000
sigma=5.67*10**-8
Ti=300
Qcc=5
M=0.05
As=0.02
omega=0.02094
cp=300
def model(y,t,a,b,c, d, T_surr):    # Set up the function that returns dTdt
    dTdt = a+b+c*sin(omega*t)+d*(y**4-T_surr**4)
    return dTdt

a = -Qcc/(M*cp)
b = 100*As/(M*cp)
c = 100*As/(M*cp)
d=-epsilon*sigma*As/(M*cp)

t = np.linspace(0,t_max,1000)
Tr = odeint(model,Ti, t, args=(a,b,c, d, T_surr))
```

```
T_ss=sum(Tr[-10:])/len(Tr[-10:])
Amplitude=T_ss-Tr[-1:]
print('Amplitude of 1', Amplitude)
```

```
Amplitude of 1 [[2.00511661]]
```