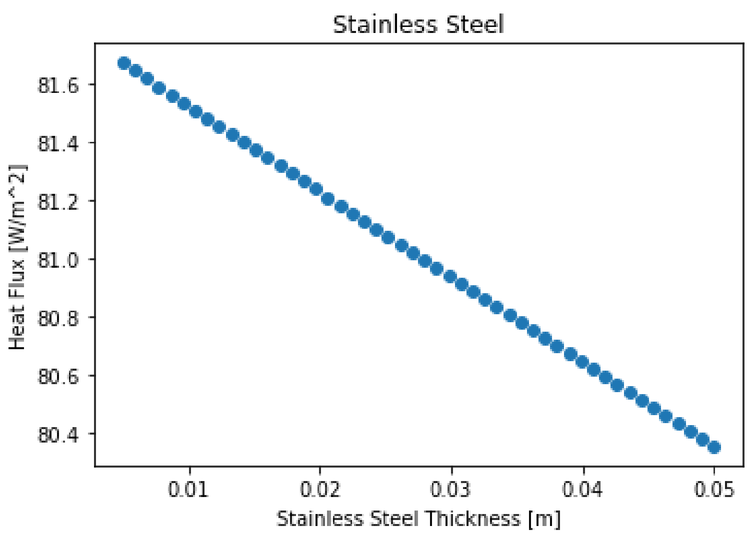
Problem 2

In [1]:
```
#Problem 2 Part 2
T_f=-10+273.15
T_r=20+273.15
h_f=10
h_r=10
th_b=1/100
th_w=5/1000
k_b=0.06
k_w=15
R_conv1=1/h_r
R_cond1=th_w/k_w
R_cond2=th_b/k_b
R_cond3=th_w/k_w
R_conv2=1/h_f
R_dprime_tot=R_conv1+R_cond1+R_cond2+R_cond3+R_conv2
q_dprime_net=(T_r-T_f)/(R_dprime_tot)
print('R"_tot:', R_dprime_tot, 'q"_total: ', q_dprime_net)
```

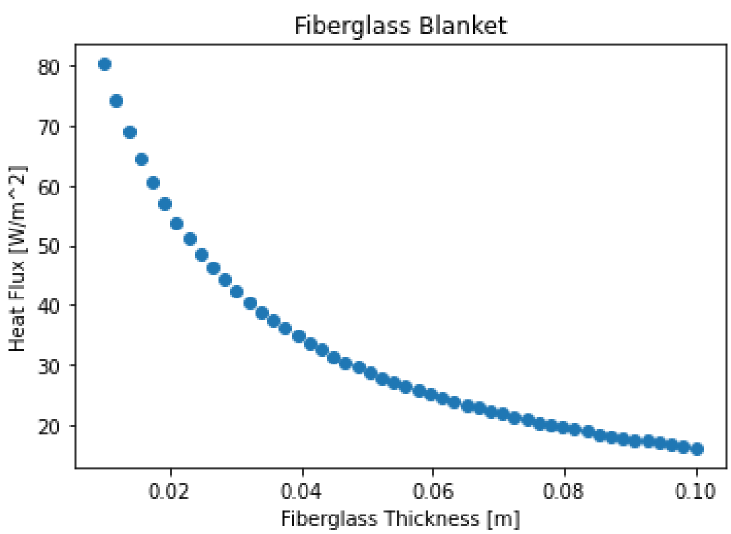
```
R"_tot: 0.3673333333333334 q"_total:  81.66969147005443
```

In [2]:

```
#Problem 2 Part 3a
import matplotlib.pyplot as plt
import numpy as np
T_f=-10+273.15
T_r=20+273.15
h_f=10
h_r=10
th_b=1/100
th_w=np.linspace(5/1000, 50/1000, 50)
k_b=0.06
k_w=15
R_conv1=1/h_r
R_cond1=th_w/k_w
R_cond2=th_b/k_b
R_cond3=th_w/k_w
R_conv2=1/h_f
R_dprime_tot=R_conv1+R_cond1+R_cond2+R_cond3+R_conv2
q_dprime_net=(T_r-T_f)/(R_dprime_tot)
plt.plot(th_w, q_dprime_net, 'o')
plt.title("Stainless Steel")
plt.xlabel("Stainless Steel Thickness [m]")
plt.ylabel("Heat Flux [W/m^2]")
plt.show()
```



In [3]:
```
#Problem 2 Part 3b
import matplotlib.pyplot as plt
import numpy as np
T_f=-10+273.15
T_r=20+273.15
h_f=10
h_r=10
th_b=np.linspace(1/100, 10/100, 50)
th_w=5/100
k_b=0.06
k_w=15
R_conv1=1/h_r
R_cond1=th_w/k_w
R_cond2=th_b/k_b
R_cond3=th_w/k_w
R_conv2=1/h_f
R_dprime_tot=R_conv1+R_cond1+R_cond2+R_cond3+R_conv2
q_dprime_net=(T_r-T_f)/(R_dprime_tot)
plt.plot(th_b, q_dprime_net, 'o')
plt.title("Fiberglass Blanket")
plt.xlabel("Fiberglass Thickness [m]")
plt.ylabel("Heat Flux [W/m^2]")
plt.show()
```



Problem 2c

Increasing the stainless steel thickness will decrease the rate of heat transfer linearly, but increasing the thickness of the fiberglass blanket will decrease the rate of heat transfer at a quicker rate and by more. In relation to the difference in heat flux for the fiberglass blanket, the difference for the stainless steel wall is minimal. Of course you have to consider the cost differences of increasing the thickness of the fiberglass versus the stainless steel.

In [4]:
```
#Problem 2 Part 4
T_f=-10+273.15
T_r=20+273.15
h_f=10
h_r=10
th_b=1/100
th_w=5/1000
k_b=0.06
k_w=15
R_conv1=1/h_r
R_cond1=th_w/k_w
R_cond2=th_b/k_b
R_cond3=th_w/k_w
R_dprime4=R_conv1+R_cond1+R_cond2+R_cond3
q_dprime_net4=81.67
T_freezer_surface=(-(q_dprime_net4*R_conv1-T_r))-273.15
print('T_freezer_surface [°C]:', T_freezer_surface)
```

```
T_freezer_surface [°C]: 11.833000000000027
```

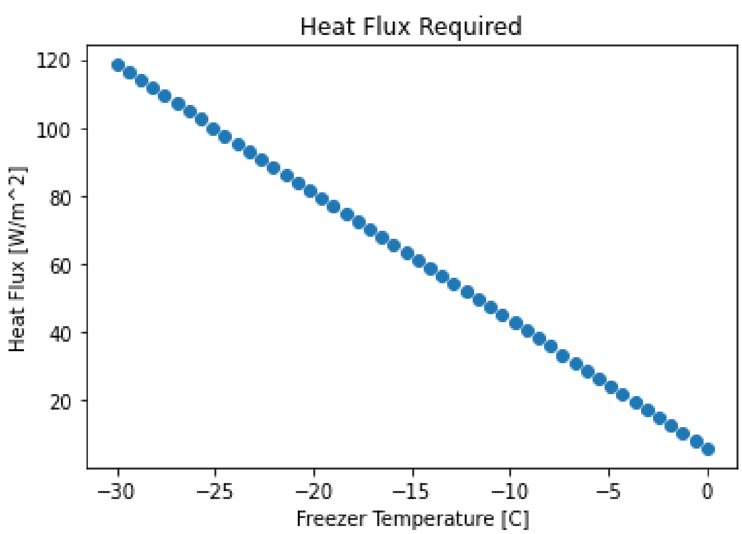
No, the current freezer does not satisfy the design requirement. It is less than 15°C

In [5]:
```
#Problem 2 Part 5a
T_f=-10+273.15
T_r=20+273.15
h_f=10
h_r=10
th_b=1/100
th_w=5/1000
k_b=0.06
k_w=15
R_conv1=1/h_r
R_cond1=th_w/k_w
R_cond2=th_b/k_b
R_cond3=th_w/k_w
R_conv2=1/h_f
q1_dprime=(T_r-(15+273.15))/R_conv1
T2=-(q1_dprime*R_cond1-(15+273.15))
q2_dprime=(T2-T_f)/(R_cond2+R_cond3+R_conv2)
q_dprime_heater=q2_dprime-q1_dprime
print('Heat required [W/m^2]:', q_dprime_heater)
```

```
Heat required [W/m^2]: 43.57053682896385
```

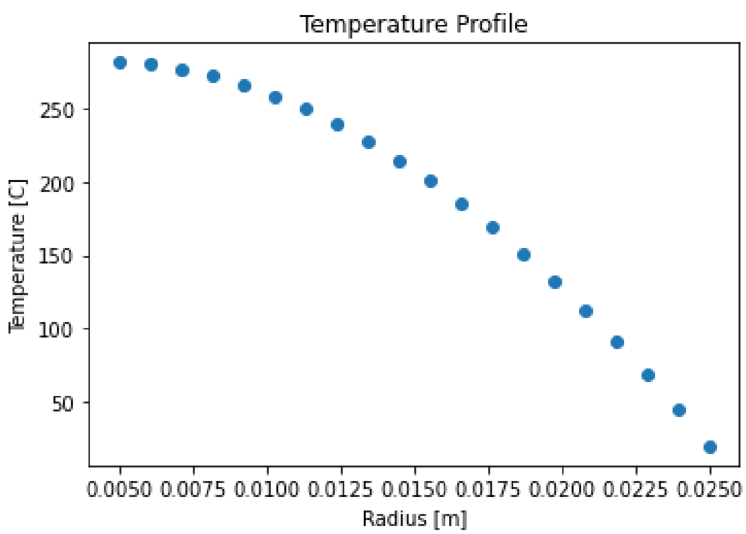
In [6]:

```
#Problem 2 Part 5b
T_f=np.linspace(-30+273.15, 0+273.15, 50)
T_r=20+273.15
h_f=10
h_r=10
th_b=1/100
th_w=5/1000
k_b=0.06
k_w=15
R_conv1=1/h_r
R_cond1=th_w/k_w
R_cond2=th_b/k_b
R_cond3=th_w/k_w
R_conv2=1/h_f
q1_dprime=(T_r-(15+273.15))/R_conv1
T2=-(q1_dprime*R_cond1-(15+273.15))
q2_dprime=(T2-T_f)/(R_cond2+R_cond3+R_conv2)
q_dprime_heater=q2_dprime-q1_dprime

plt.plot(T_f-273.15, q_dprime_heater, 'o')
plt.title("Heat Flux Required")
plt.xlabel("Freezer Temperature [C]")
plt.ylabel("Heat Flux [W/m^2]")
plt.show()
```



Problem 3

In [7]:
```
#Problem 3 Part 5
r_in = 5/1000
r_out=25/1000
q_dot=10**6
q_dprime_in = 100
T_out = 20+273.15
k=0.5
r=np.linspace(r_in, r_out, 20)
C1=(-r_in/k)*(q_dprime_in-(q_dot*r_in/2))
C2=T_out+((q_dot*(r_out**2))/(4*k))-C1*np.log(r_out)
T=((-q_dot*(r**2))/(4*k))+C1*np.log(r)+C2
plt.plot(r, T-273.15, 'o')
plt.title("Temperature Profile")
plt.xlabel("Radius [m]")
plt.ylabel("Temperature [C]")
plt.show()
```



Problem 4

In [8]:
```
#Square
D=1*10**(-3)
h=35
k=2.8
L=10*10**-3
P_square=4*D
A_f_square=P_square*L
A_c_square=D**2
m_square=np.sqrt((h*P_square)/(k*A_c_square))
Efficiency_Square=np.tanh(m_square*L)/(m_square*L)
print("Square Efficiency:", Efficiency_Square)
```

```
Square Efficiency: 0.43711204016107363
```

In [9]:
```
#Triangle
D=1*10**(-3)
k=2.8
h=35
L=10*10**-3
P_triangle=D*3
A_f_triangle=P_triangle*L
A_c_triangle= (np.sqrt(3)/4)*D**2
m_triangle=np.sqrt((h*P_triangle)/(k*A_c_triangle))
Efficiency_Triangle=np.tanh(m_triangle*L)/(m_triangle*L)
print("Triangle Efficiency:", Efficiency_Triangle)
```

```
Triangle Efficiency: 0.33792542125981023
```

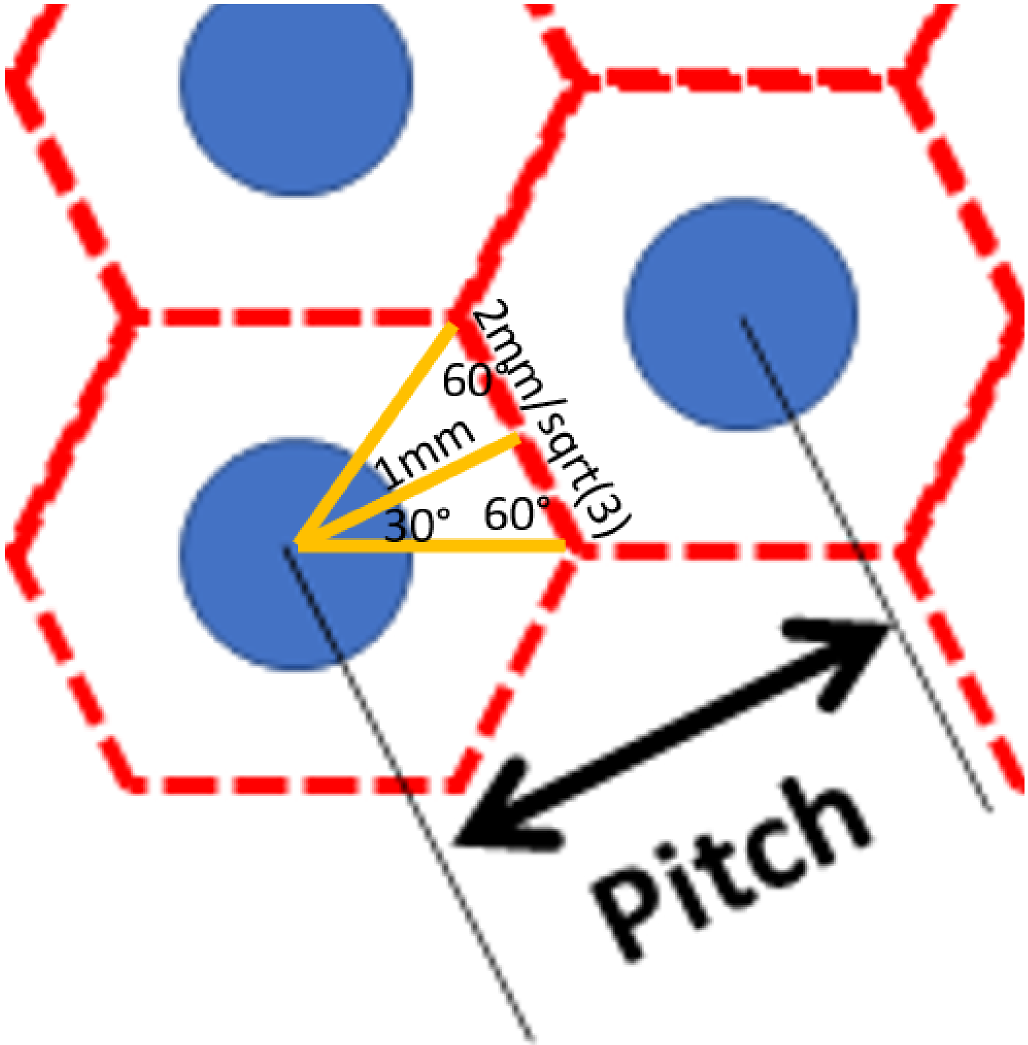
In [11]:
```
#Circle
import math # math functions
D=1*10**(-3)
k=2.8
h=35
L=10*10**-3
P_circle=math.pi*D
A_f_circle=P_circle*L
A_c_circle= (math.pi*D**2)/4
m_circle=np.sqrt((h*P_circle)/(k*A_c_circle))
Efficiency_Circle=np.tanh(m_circle*L)/(m_circle*L)
print("Circle Efficiency:", Efficiency_Circle)
```

```
Circle Efficiency: 0.43711204016107363
```

The circle and the square have the same efficiencies

In [12]:
```
#Number of Fins: 1
side=(2*10**(-3))/np.sqrt(3)
A_Hexagon = (3*np.sqrt(3)/2)*side**2
N=1/A_Hexagon
print(N)
```

```
288675.1345948129
```



In [13]:
```
#R_unfin: 2 Circle
N=288675
A_b_circle= 1-A_c_circle*N
R_conv_unfin=1/(h*A_b_circle)
print('Circle:', R_conv_unfin)
```

```
Circle: 0.036948591030671635
```

In [14]:
```
#R_unfin: 2 Square
A_b_square= 1-A_c_square*N
R_conv_unfin=1/(h*A_b_square)
print('Square:', R_conv_unfin)
```

```
Square: 0.040166490101470595
```

In [15]:
```
#R_unfin: 2 Triangle
A_b_triangle= 1-A_c_triangle*N
R_conv_unfin=1/(h*A_b_triangle)
print('Triangle:', R_conv_unfin)
```

```
Triangle: 0.03265305904956232
```

In [16]:
```
#R_fin: 3 Circle
R_fin_circle=1/(Efficiency_Circle*h*A_f_circle)
print('Circle:', R_fin_circle)
```

```
Circle: 2080.6034474201233
```

In [17]:
```
#R_fin: 3 Square
R_fin_square=1/(Efficiency_Square*h*A_f_square)
print('Square:', R_fin_square)
```

```
Square: 1634.1021263621644
```

In [18]:
```
#R_fin: 3 Triangle
R_fin_triangle=1/(Efficiency_Triangle*h*A_f_triangle)
print('Square:', R_fin_triangle)
```

```
Square: 2818.3169790257502
```

In [19]:
```
#Total Resistance and Efficiency Circle
A_tot_circle=N*A_f_circle+A_b_circle
Surface_Efficiency_circle= 1-(N*A_f_circle/A_tot_circle)*(1-Efficiency_Circle)
print("Surface Efficiency Circle:", Surface_Efficiency_circle)
Total_resistance_circle=1/(Surface_Efficiency_circle*h*A_tot_circle)
print("Total Resistance Circle:", Total_resistance_circle)
```

```
Surface Efficiency Circle: 0.48133632883128197
Total Resistance Circle: 0.006030983463979283
```

In [21]:
```
#Total Resistance and Efficiency Square
A_tot_square=N*A_f_square+A_b_square
Surface_Efficiency_square= 1-(N*A_f_square/A_tot_square)*(1-Efficiency_Square)
print("Surface Efficiency Square:", Surface_Efficiency_square)
Total_resistance_square=1/(Surface_Efficiency_square*h*A_tot_square)
print("Total Resistance Square:", Total_resistance_square)
```

```
Surface Efficiency Square: 0.4697752529599205
Total Resistance Square: 0.0049614736492842254
```

In [22]:
```
#Total Resistance and Efficiency Triangle
A_tot_triangle=N*A_f_triangle+A_b_triangle
Surface_Efficiency_triangle= 1-(N*A_f_triangle/A_tot_triangle)*(1-Efficiency_Triangle)
print("Surface Efficiency Triangle:", Surface_Efficiency_triangle)
Total_resistance_triangle=1/(Surface_Efficiency_triangle*h*A_tot_triangle)
print("Total Resistance Triangle:", Total_resistance_triangle)
```

```
Surface Efficiency Triangle: 0.39868054476924353
Total Resistance Triangle: 0.007515793270600793
```

In [23]:
```
#Heat Transfer Rate Circle
T_gas=35+273.15
T_base=60+273.15
Theta=T_base-T_gas
qt_circle=Theta/Total_resistance_circle
print("Heat Transfer Rate Circle [W]:", qt_circle)
```

```
Heat Transfer Rate Circle [W]: 4145.260909653503
```

In [24]:
```
#Heat Transfer Rate Square
qt_square=Theta/Total_resistance_square
print("Heat Transfer Rate Square [W]:", qt_square)
```

```
Heat Transfer Rate Square [W]: 5038.8255117724275
```

In [25]:
```
#Heat Transfer Rate Triangle
qt_triangle=Theta/Total_resistance_triangle
print("Heat Transfer Rate Triangle [W]:", qt_triangle)
```

```
Heat Transfer Rate Triangle [W]: 3326.328851778219
```

Problem 5

In [26]:
```
D=5*10**-2
u=0.75
T_in=300
k=40
alpha=0.001
th=0.6*10**-3
k_g=0.03
L_c=D/4
h=k_g/th
Bi=h*L_c/k
print(Bi)
```

```
0.015625
```

In [27]:
```python
# Set up the python script
# import required modules

#Numerical Math
import math # math functions
import numpy as np # numerical math
from numpy import *

# Scientific Math
import scipy # scientific computing functions
import scipy.special as sp # scientific computing functions
import scipy.misc as misc

#Symbolic math
import sympy # symbolic math
from sympy import *
from sympy.abc import x

#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)


T = symbols('T', cls=Function)    # define T as a function
a1, a2, Tw0, Tf, Lc, xinf, Tin = symbols('a1 a2 Tw0 Tf Lc xinf Tin', positive = true)   # defi

eq = T(x).diff(x,x) - a1*T(x).diff(x) + a2*(T(x)-(Tf - (Tf-Tw0)*exp(-x/Lc)))        # define
Teq=dsolve(eq)                            # symbolically solve the ode
pprint(Teq)
print(Teq)
```

```
                ⎛    ________⎞             ⎛    ________⎞
                ⎜   /   2    ⎟             ⎜   /   2    ⎟
              x⋅⎝a₁ - ╲╱  a₁  - 4⋅a₂ ⎠   x⋅⎝a₁ + ╲╱  a₁  - 4⋅a₂ ⎠
              ────────────────────────   ────────────────────────      2
                         2                          2                Lc ⋅Tf⋅
T(x) = C₁⋅ℯ                         + C₂⋅ℯ                       - ─────────
                                                                      2
                                                                    Lc ⋅a₂ + 

    -x             -x
    ───            ───
    Lc      2      Lc
a₂⋅ℯ      Lc ⋅Tw₀⋅a₂⋅ℯ
─────── + ─────────── + Tf
           2
Lc⋅a₁ + 1  Lc ⋅a₂ + Lc⋅a₁ + 1
Eq(T(x), C1*exp(x*(a1 - sqrt(a1**2 - 4*a2))/2) + C2*exp(x*(a1 + sqrt(a1**2 - 4*a2))/2) - Lc*
*2*Tf*a2*exp(-x/Lc)/(Lc**2*a2 + Lc*a1 + 1) + Lc**2*Tw0*a2*exp(-x/Lc)/(Lc**2*a2 + Lc*a1 + 1)
+ Tf)
```

In [28]:

```python
C1, C2 = symbols('C1 C2')     # define these as symbols
constants = sympy.solve([Teq.rhs.subs(x,0)-Tw0, Teq.rhs.subs(x,xinf)-Tf ],[C1,C2])  #Solve for
#pprint(constants)
Teq = Teq.subs(constants)  # Substitute constants C1 and C2 into the equation
pprint(Teq)
print(Teq)
```

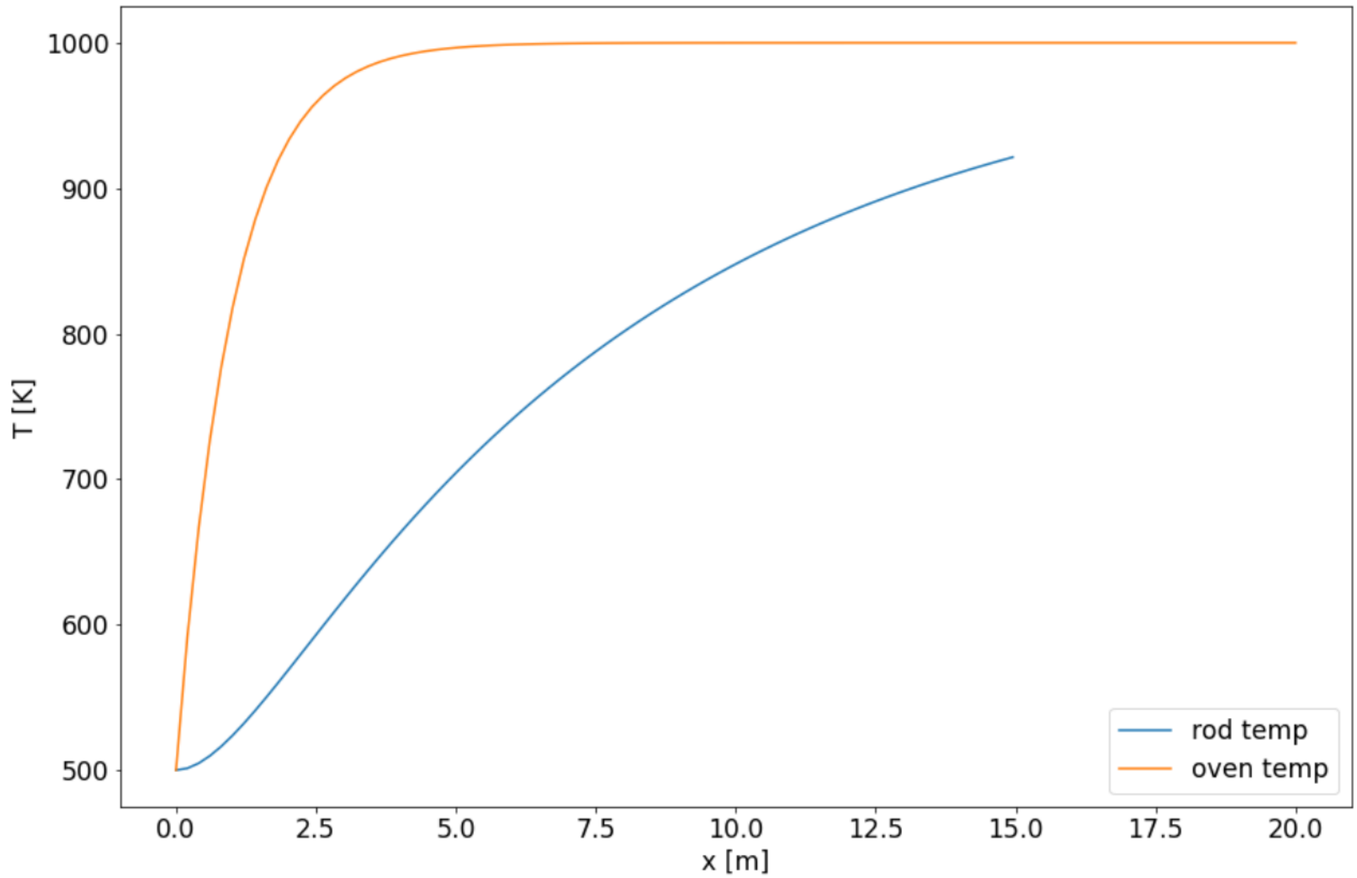
```
                                                                       │
                       -x                        -x                    │
                       ───                       ───                   │
           2           Lc             2          Lc              ⎛     │  2
        Lc ⋅Tf⋅a₂⋅ℯ                Lc ⋅Tw₀⋅a₂⋅ℯ                  ⎜Lc ⋅a₂⋅(Tf - Tw₀) + (
T(x) = - ──────────────────── + ──────────────────── + Tf + ─────────────────────────────
          2                      2
        Lc ⋅a₂ + Lc⋅a₁ + 1    Lc ⋅a₂ + Lc⋅a₁ + 1

                                                  ⎛             ________    ⎞⎞⎞  ⎛
                                                  ⎜            ╱   2        ⎟ │  ⎜
                                          xinf⋅ ⎝Lc⋅a₁ + Lc⋅╲╱  a₁  - 4⋅a₂  + 2⎠ │ x⋅⎝a
                                          ──────────────────────────────────────  ────
                                                         2⋅Lc                 │
Lc⋅Tf⋅a₁ - Lc⋅Tw₀⋅a₁ + Tf - Tw₀)⋅ℯ                                          ⎠⋅ℯ
─────────────────────────────────────────────────────────────────────────────────
                                         ⎛            ________⎞
                                         ⎜           ╱   2    ⎟
                                         ⎜   xinf⋅╲╱  a₁  - 4⋅a₂⎟ ⎛  2
                                         ⎝1 - ℯ                 ⎠⋅⎝Lc ⋅a₂ + Lc⋅a₁ +

          ________⎞              ⎛                ________        ⎞  ⎛
        ╱   2     ⎟              ⎜          ╱   2                 ⎟  ⎜
₁ - ╲╱  a₁  - 4⋅a₂ ⎠             ⎜   a₁   ╲╱  a₁  - 4⋅a₂      1   ⎟  ⎜
──────────────────────  xinf⋅⎜- ── + ──────────────── - ──⎟  │
          2                      ⎝   2            2            Lc⎠ ⎝     2
                       ⋅ℯ                                         - Lc ⋅a₂⋅(Tf - Tw₀
────────────────────────────────────────────────────────────────── + ─────────────────

⎞
1⎠

        ________
       ╱   2                                                    xinf⋅(Lc⋅a₁ + 2)
  xinf⋅╲╱  a₁  - 4⋅a₂                                          ────────────────
  ──────────────────                                                2⋅Lc
          2
)⋅ℯ                  + (-Lc⋅Tf⋅a₁ + Lc⋅Tw₀⋅a₁ - Tf + Tw₀)⋅ℯ
─────────────────────────────────────────────────────────────────────────────────
                           ⎛             ________⎞
                           ⎜            ╱   2    ⎟ ⎛  2                  ⎞
                           ⎜    xinf⋅╲╱  a₁  - 4⋅a₂⎟ ⎜Lc ⋅a₂ + Lc⋅a₁ + 1⎟
                           ⎝1 - ℯ                 ⎠⋅⎝                   ⎠
⎞     ⎛             ________⎞
⎟     ⎜            ╱   2    ⎟
⎟ x⋅ ⎝a₁ + ╲╱  a₁  - 4⋅a₂ ⎠    a₁⋅xinf    xinf
⎟ ──────────────────────── - ─────── - ────
⎟              2                 2          Lc
⎠⋅ℯ                          ⋅ℯ
──────────────────────────────────────────
```

```
Eq(T(x), -Lc**2*Tf*a2*exp(-x/Lc)/(Lc**2*a2 + Lc*a1 + 1) + Lc**2*Tw0*a2*exp(-x/Lc)/(Lc**2*a
2 + Lc*a1 + 1) + Tf + (Lc**2*a2*(Tf - Tw0) + (Lc*Tf*a1 - Lc*Tw0*a1 + Tf - Tw0)*exp(xinf*(L
c*a1 + Lc*sqrt(a1**2 - 4*a2) + 2)/(2*Lc)))*exp(x*(a1 - sqrt(a1**2 - 4*a2))/2)*exp(xinf*(-a
1/2 + sqrt(a1**2 - 4*a2)/2 - 1/Lc))/((1 - exp(xinf*sqrt(a1**2 - 4*a2)))*(Lc**2*a2 + Lc*a1
+ 1)) + (-Lc**2*a2*(Tf - Tw0)*exp(xinf*sqrt(a1**2 - 4*a2)/2) + (-Lc*Tf*a1 + Lc*Tw0*a1 - Tf
+ Tw0)*exp(xinf*(Lc*a1 + 2)/(2*Lc)))*exp(x*(a1 + sqrt(a1**2 - 4*a2))/2)*exp(-a1*xinf/2 - x
inf/Lc)/((1 - exp(xinf*sqrt(a1**2 - 4*a2)))*(Lc**2*a2 + Lc*a1 + 1)))
```

In [ ]:

```
k = 40        #[W/m-K] Thermal conductivity of rod
k_g = 0.03     #[W/m-K] Thermal conductivity of gas
alpha = 0.001     #[m^2/s] diffusivity of rod
u = 0.75      #[m/s] speed of rod
th = 0.6*10**-3   #[m] thickness of gap
D =5*10**-2       #[m] diameter of rod
Lcs = 1           #[m] decay length for oven temp
Tw0s = 500        #[C] Wall inlet temp
Tfs = 1000        #[C] Wall downstream temp
Tins = 300        #[C] inlet temp
L = 20            #[m] lenght for plotting
xinfs = 2000      #[m] a long length to assume x --> infty

a1s = u/alpha
a2s = -4*k_g/(k*D*th)

Tp = Teq.rhs.subs([(a1,a1s),(a2,a2s),(Tw0,Tw0s),(Tf,Tfs),(Lc,Lcs),(xinf,xinfs),(Tin,Tins)])  #
Tp = lambdify(x,Tp,"numpy")
xp = np.linspace(0,L,num=100, dtype=float128)
Tp = Tp(xp)
plt.figure(figsize=(15,10))
plt.plot(xp,Tp,label = 'rod temp')
Tw = (Tfs - (Tfs-Tw0s)*np.exp(-xp/Lcs))
plt.plot(xp,Tw,label='oven temp')
plt.xlabel('x [m]')
plt.ylabel('T [K]')
plt.legend();
```

float128 didn't work on my Jupyter Notebook so I ran this section in Google Colabs using the same code as above


1000
900
800
700
600
500
T [K]
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0
x [m]
rod temp
oven temp

```
In [29]: T_prime=Teq.rhs.diff(x)
         pprint(T_prime)
         print(T_prime)
```

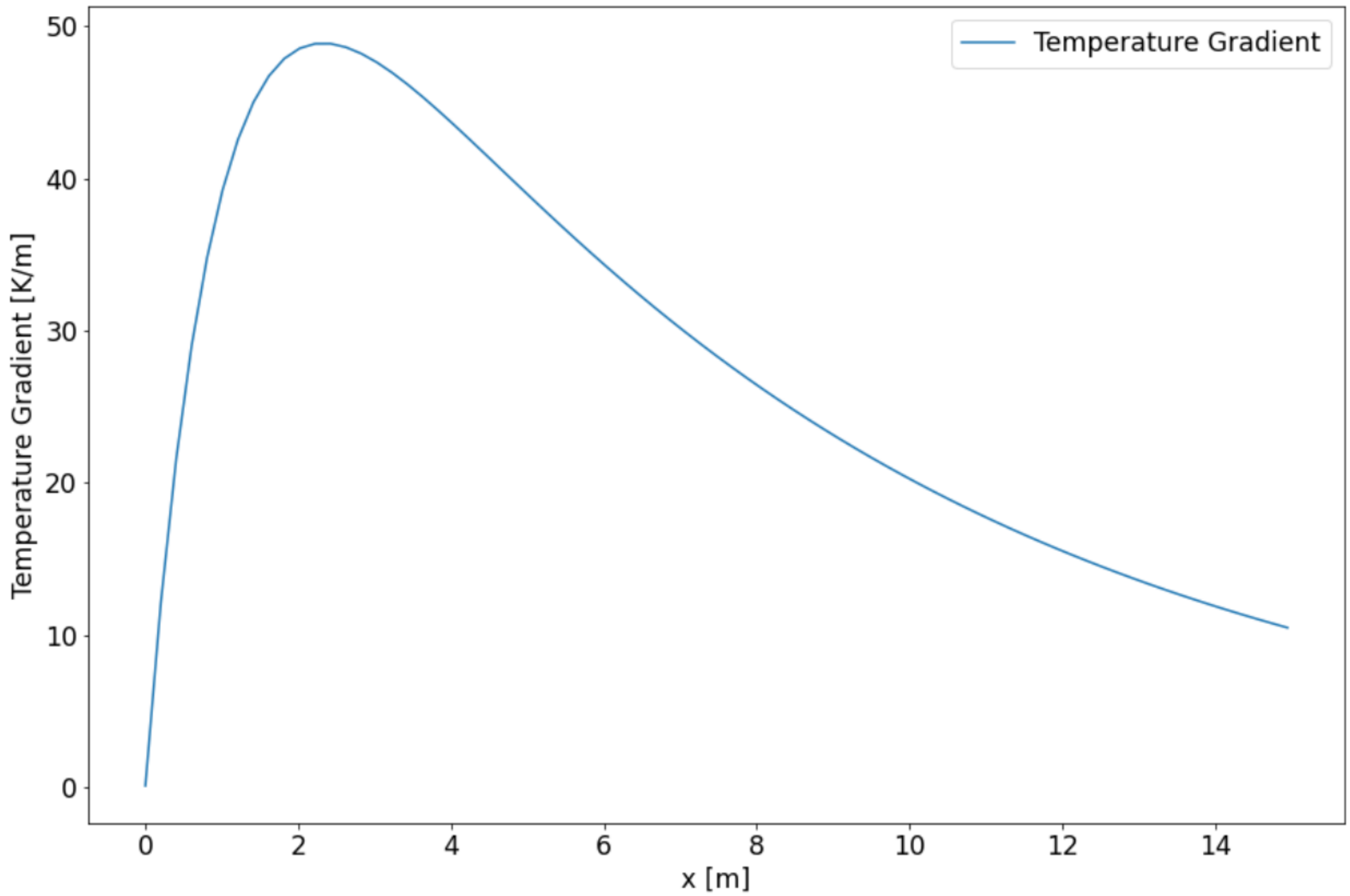
```
                                                        ⎛                          ⎞⎞
                                                        ⎜          ________________⎟⎟
              -x                    -x                  ⎜         ╱   2            ⎟⎟
              ───                   ───                 ⎜  a₁    ╲╱  a₁   - 4⋅a₂   ⎟⎟   2
              Lc                    Lc                 ⎛⎜  ── -  ──────────────────⎟⎟⋅⎝Lc ⋅a₂⋅(Tf -
  Lc⋅Tf⋅a₂⋅ℯ           Lc⋅Tw₀⋅a₂⋅ℯ                     ⎝2  2            2          ⎠
────────────────── - ────────────────── + ──────────────────────────────────────────────────────
  2                    2
Lc ⋅a₂ + Lc⋅a₁ + 1   Lc ⋅a₂ + Lc⋅a₁ + 1

                                                              ⎛
                                                              ⎜             ________________
                                                              ⎜            ╱   2
                                                       xinf⋅  ⎝Lc⋅a₁ + Lc⋅╲╱  a₁   - 4⋅a₂   +
                                                       ──────────────────────────────────────
                                                                        2⋅Lc
 Tw₀) + (Lc⋅Tf⋅a₁ - Lc⋅Tw₀⋅a₁ + Tf - Tw₀)⋅ℯ
────────────────────────────────────────────────────────────────────────────────────────────
                                              ⎛                            ⎞
                                              ⎜          ________________  ⎟
                                              ⎜         ╱   2              ⎟⎛  2                  ⎞
                                              ⎜ xinf⋅╲╱  a₁   - 4⋅a₂      ⎟⎜                     ⎟
                                              ⎝1 - ℯ                       ⎠⋅⎝Lc ⋅a₂ + Lc⋅a₁ + 1⎠

 ⎞⎞      ⎛                          ⎞        ⎛                                          ⎞
 ⎟⎟      ⎜      ________________    ⎟        ⎜          ________________                ⎟ ⎛
 ⎟⎟      ⎜     ╱   2                ⎟        ⎜         ╱   2                            ⎟ ⎜
2⎠⎟  x⋅  ⎝a₁ - ╲╱  a₁   - 4⋅a₂      ⎠  xinf⋅⎜⎜  a₁   ╲╱  a₁   - 4⋅a₂       1  ⎟       ⎜       ╱
──⎟  ──────────────────────────────       ⎜⎜- ── + ──────────────────── - ──⎟       ⎜ a₁  ╲╱
  ⎟               2                       ⎝⎝  2             2              Lc⎠      ⎜⎜── + ────
  ⎠⋅ℯ                               ⋅ℯ                                              ⎝2
────────────────────────────────────────────────────────────────────────────────── + ───────────

                          ⎛
 ________________       ⎞ ⎜                       ________________
   2                    ⎟ ⎜                      ╱   2
a₁   - 4⋅a₂             ⎟ ⎜              xinf⋅╲╱  a₁   - 4⋅a₂
                        ⎟ ⎜      2       ─────────────────────────
────────────────        ⎟ ⎜                          2
      2                 ⎠⋅⎝- Lc ⋅a₂⋅(Tf - Tw₀)⋅ℯ                        + (-Lc⋅Tf⋅a₁ + Lc⋅Tw₀⋅
────────────────────────────────────────────────────────────────────────────────────────────
                                                              ⎛                           ⎞
                                                              ⎜          ________________ ⎟
                                                              ⎜         ╱   2             ⎟⎛
                                                              ⎜ xinf⋅╲╱  a₁   - 4⋅a₂     ⎟⎜
                                                              ⎝1 - ℯ                      ⎠⋅⎝

                                   ⎞   ⎛                              ⎞
                                   ⎟   ⎜          ________________    ⎟
                                   ⎟   ⎜         ╱   2                ⎟
                 xinf⋅(Lc⋅a₁ + 2)  ⎟ x⋅⎝a₁ + ╲╱  a₁   - 4⋅a₂          ⎠   a₁⋅xinf   xinf
                 ────────────────  ⎟  ──────────────────────────────── - ─────── - ────
                       2⋅Lc        ⎟                  2                     2        Lc
a₁ - Tf + Tw₀)⋅ℯ                   ⎠⋅ℯ                                ⋅ℯ
```

```
  2                        ⎞⎞
Lc ⋅a₂ + Lc⋅a₁ + 1⎠⎠
Lc*Tf*a2*exp(-x/Lc)/(Lc**2*a2 + Lc*a1 + 1) - Lc*Tw0*a2*exp(-x/Lc)/(Lc**2*a2 + Lc*a1 + 1) +
(a1/2 - sqrt(a1**2 - 4*a2)/2)*(Lc**2*a2*(Tf - Tw0) + (Lc*Tf*a1 - Lc*Tw0*a1 + Tf - Tw0)*exp(x
inf*(Lc*a1 + Lc*sqrt(a1**2 - 4*a2) + 2)/(2*Lc)))*exp(x*(a1 - sqrt(a1**2 - 4*a2))/2)*exp(xinf
*(-a1/2 + sqrt(a1**2 - 4*a2)/2 - 1/Lc))/((1 - exp(xinf*sqrt(a1**2 - 4*a2)))*(Lc**2*a2 + Lc*a
1 + 1)) + (a1/2 + sqrt(a1**2 - 4*a2)/2)*(-Lc**2*a2*(Tf - Tw0)*exp(xinf*sqrt(a1**2 - 4*a2)/2)
+ (-Lc*Tf*a1 + Lc*Tw0*a1 - Tf + Tw0)*exp(xinf*(Lc*a1 + 2)/(2*Lc)))*exp(x*(a1 + sqrt(a1**2 -
4*a2))/2)*exp(-a1*xinf/2 - xinf/Lc)/((1 - exp(xinf*sqrt(a1**2 - 4*a2)))*(Lc**2*a2 + Lc*a1 +
1))
```

In [ ]:

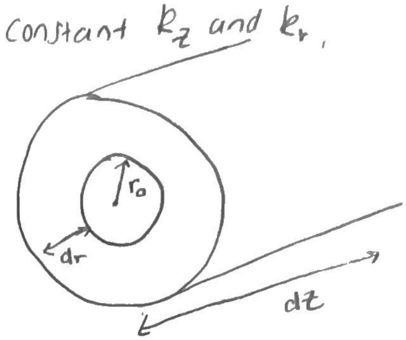
```
 = 40        #[W/m-K] Thermal conductivity of rod
_g = 0.03     #[W/m-K] Thermal conductivity of gas
lpha = 0.001     #[m^2/s] diffusivity of rod
 = 0.75      #[m/s] speed of rod
h = 0.6*10**-3   #[m] thickness of gap
 =5*10**-2       #[m] diameter of rod
cs = 1           #[m] decay length for oven temp
w0s = 500        #[C] Wall inlet temp
fs = 1000        #[C] Wall downstream temp
ins = 300        #[C] inlet temp
 = 20            #[m] lenght for plotting
infs = 2000      #[m] a long length to assume x --> infty

1s = u/alpha
2s = -4*k_g/(k*D*th)

T = T_prime.subs([(a1,a1s),(a2,a2s),(Tw0,Tw0s),(Tf,Tfs),(Lc,Lcs),(xinf,xinfs),(Tin,Tins)])  # 
T = lambdify(x,dT,"numpy")
p = np.linspace(0,L,num=100, dtype=float128)
T = dT(xp)
lt.figure(figsize=(15,10))
lt.plot(xp,dT,label = 'rod temp')
lt.legend();
```

again, float128 did not work in Jupyter so I ran the above code in Google Colab


Temperature Gradient
Temperature Gradient [K/m]
x [m]
0
10
20
30
40
50
0
2
4
6
8
10
12
14

Problem 1:

Assumptions: Steady State, 2D (r and z directions), constant k_z and k_r, Spatially varying heat generation Rate $\dot{q}$



① $\underbrace{\cancel{\rho C_p \frac{\partial T}{\partial t}}}_{0 (SS)} = \frac{1}{r}\frac{\partial}{\partial r}\left(\underbrace{k_r}_{\text{constant } k_r} r \frac{\partial T}{\partial r}\right) + \underbrace{\cancel{\frac{1}{r^2}\frac{\partial}{\partial \theta}\left(k_\theta \frac{\partial T}{\partial \theta}\right)}}_{\text{2D in } r \,\&\, z \text{ directions}} + \frac{\partial}{\partial z}\left(\underbrace{k_z}_{\text{constant } k_z} \frac{\partial T}{\partial z}\right) + \dot{q}$

$$\boxed{0 = \frac{k_r}{r}\frac{d}{dr}\left(r\frac{dT}{dr}\right) + k_z\left(\frac{d^2T}{dz^2}\right) + \dot{q}(r,z)}$$

② $q_r = q_r'' A = \boxed{-k_r \frac{\partial T}{\partial r} r_o (2\pi) dz = q_r}$

③ $q_{r+dr} = q''_{r+dr} A = q''_r r_o (2\pi) dz + \frac{\partial}{\partial r}\left(q''_r r (2\pi) dz\right) dr = \boxed{-k_r \frac{\partial T}{\partial r} r_o (2\pi) dz + \frac{\partial}{\partial r}\left(-k_r \frac{dT}{dr} r (2\pi) dz\right) dr}$

$q_{r+dr} \leftarrow q_r + \frac{\partial}{\partial r}(q_r) dr$

④ $q_r - q_{r+dr} = \underbrace{-k_r \frac{\partial T}{\partial r} r (2\pi) dz}_{\text{cancels}} - \left(\underbrace{-k_r \frac{\partial T}{\partial r} r (2\pi) dz}_{\text{cancels}} + \frac{\partial}{\partial r}\left(-k_r \frac{\partial T}{\partial r} r (2\pi) dz\right) dr\right)$

$q_r - q_{r+dr} = -\frac{\partial}{\partial r}\left(-\underbrace{k_r}_{\text{constant}} \frac{\partial T}{\partial r} r\right) \underbrace{(2\pi) dz dr}_{2\pi \underline{r} dz dr = \text{volume}}$

$$\boxed{q_r - q_{r+dr} = k_r \frac{\partial}{\partial r}\left(r \frac{\partial T}{\partial r}\right)(2\pi) dz dr}$$

⑤

$q_z = q_z'' A$

$q_z = -k_z \frac{\partial T}{\partial z} r_o (2\pi) dr$

$q_{z+dz} = -k_z \frac{\partial T}{\partial z} r_o (2\pi) dr + \frac{\partial}{\partial z}\left(-k_z \frac{\partial T}{\partial z} r_o (2\pi) dr\right) dz$

$q_z - q_{z+dz} = -k_z \frac{\partial T}{\partial z} r_o (2\pi) dr - \left(-k_z \frac{\partial T}{\partial z} r_o (2\pi) dr + \frac{\partial}{\partial z}\left(-k_z \frac{\partial T}{\partial z} r_o (2\pi) dr\right) dz\right)$

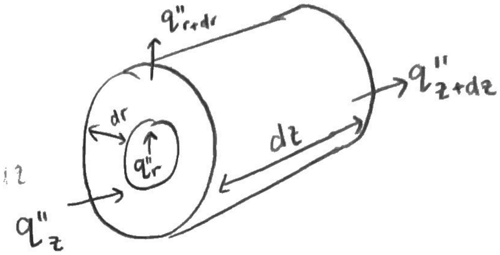
$q_z - q_{z+dz} = -\underbrace{\frac{\partial}{\partial z}}_{\text{constant}}\left(-k_z \frac{dT}{dz}\right) \underbrace{r_o (2\pi) dz dr}_{\text{volume}}$

$$\boxed{q_z - q_{z+dz} = k_z \frac{\partial}{\partial z}\left(\frac{\partial T}{\partial z}\right) r_o (2\pi) dz dr}$$

→

(5)

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



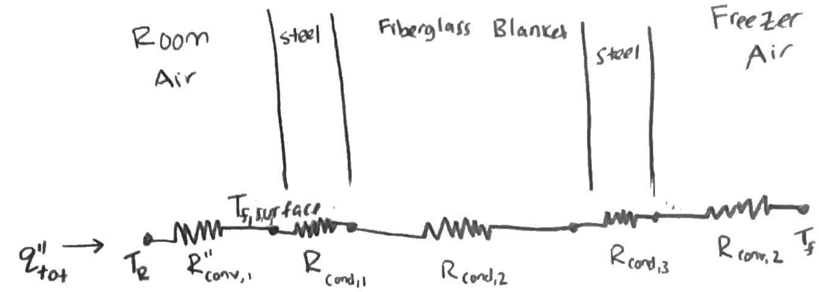
$$q_r + q_z - q_{r+dz} - q_{z+dz} + \dot{q}\,\forall = \rho C_p \frac{dT}{dt}\forall$$

$$(q_r - q_{r+dr}) + (q_z + q_{z+dz}) + \dot{q}\, r_0 dr(2\pi)dz = \rho c_p \frac{dT}{dt}\, r dr_0 (2\pi) dz$$

$$\left[\left(k_r \frac{\partial}{\partial r}\left(r\frac{\partial T}{\partial r}\right)\underbrace{(2\pi)dzdr}_{\frac{\forall}{r_0}}\right) + \left(k_z \frac{\partial}{\partial z}\left(\frac{\partial T}{\partial z}\right)\underbrace{r_0(2\pi)dzdr}_{\forall}\right) + \dot{q}\,\underbrace{r_0 dr(2\pi)dz}_{\forall} = \rho C_p \frac{\partial T}{\partial t}\underbrace{r_0 dr(2\pi)dz}_{\forall}\right.$$

$$\boxed{\frac{k_r}{r_0}\frac{\partial}{\partial r}\left(r\frac{\partial T}{\partial r}\right) + k_z\frac{\partial}{\partial z}\left(\frac{\partial T}{\partial z}\right) + \dot{q} = \rho C_p \frac{dT}{dt}}$$

Problem 2:
Assumptions



(1) $R''_{conv,1} = \frac{1}{h_r}$

$R''_{cond,1} = \frac{th_w}{k_w}$

$R_{cond,2} = \frac{th_b}{k_b}$

$R_{cond,3} = \frac{th_w}{k_w}$

$R_{conv,2} = \frac{1}{h_f}$

$$R''_{tot} = R''_{conv,1} + R''_{cond,1} + R''_{cond,2} + R''_{cond,3} + R''_{conv,2}$$

$$q''_{net} = \frac{(T_R - T_f)}{R''_{tot}}$$

* Jupyter Notebook *

(2) $R''_{tot} = 0.367$

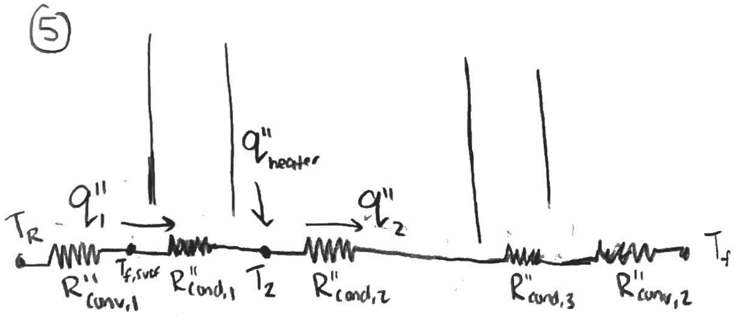
$q''_{tot} = 81.67 \ \frac{W}{m^2}$

* Jupyter Notebook *

(4) $q''_{net} = \frac{(T_R - T_{f,surface})}{R''_{conv,1}}$ $\rightarrow T_{f,surface} = -[(q''_{net} R''_{conv,1}) - T_R]$

$T_{f,surface} = 11.83°C$ * Jupyter Notebook *

(5)



$\dot{E}_{in} = \dot{E}_{out}$

$q''_1 + q''_{heater} = q''_2$

$$q''_1 = \frac{T_R - \overset{15°C}{T_{freezer\ surface}}}{R_{conv,1}}$$

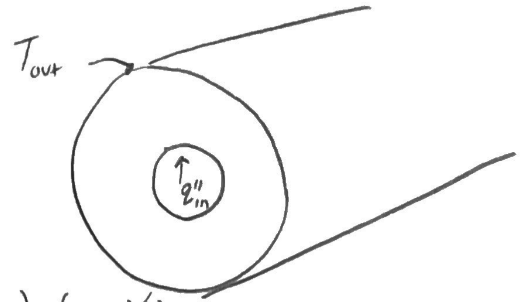
Solve for T_2 $\rightarrow q''_1 = \frac{T_{freezer\ surface} - T_2}{R''_{cond,1}}$

Solve for q''_2 $\rightarrow q''_2 = \frac{T_2 - T_f}{R''_{cond,2} + R_{cond,3} + R_{conv,2}}$

$$q''_{heater} = q''_2 - q''_1$$

Problem 3

Assumptions: 1D, steady state, constant k



Constant k_r

(1) $\underbrace{\rho C_p \cancel{\frac{dT}{dt}}}_{0\ (SS)} = \frac{1}{r}\frac{\partial}{\partial r}\left(k_r r \frac{\partial T}{\partial r}\right) + \underbrace{\frac{1}{r^2}\frac{\partial}{\partial \theta}\left(k_\theta \cancel{\frac{\partial T}{\partial \theta}}\right)}_{0\ (1D)} + \underbrace{\frac{\partial}{\partial z}\left(k_z \cancel{\frac{\partial T}{\partial z}}\right)}_{0\ (1D)} + \dot{q}$

$$0 = \frac{k_r}{r}\frac{\partial}{\partial r}\left(r\frac{\partial T}{\partial r}\right) + \dot{q}$$

$$r\frac{dT}{dr} = \frac{-\dot{q}}{2k_r}r^2 + C_1 \rightarrow -k_r\frac{dT}{dr} = \frac{\dot{q}r}{2} - k_r\frac{C_1}{r}$$

(2)

$$T(r) = -\frac{\dot{q}}{4k_r}r^2 + C_1\ln(r) + C_2$$

(3) Boundary Conditions

Constant Surface Heat flux

$$-k\left.\frac{dT}{dr}\right|_{r=r_{in}} = q''_{in} = \frac{\dot{q}r_{in}}{2} - \frac{k_r C_1}{r} \qquad -\frac{r_{in}}{k_r}\left[q''_{in} - \frac{\dot{q}r_{in}}{2}\right] = C_1$$

$$T(r=r_{out}) = T_{out} = -\frac{\dot{q}}{4k_r}r_{out}^2 + \left[\left(-\frac{r_{in}}{k_r}\right)\left(q''_{in} - \frac{\dot{q}r_{in}}{2}\right)\right]\ln(r_{out}) + C_2$$

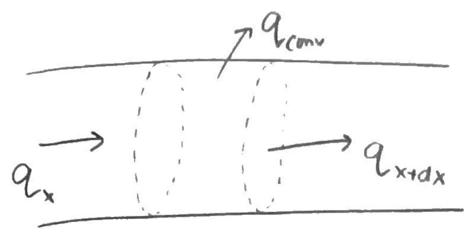
(4) $\underbrace{C_2 = T_{out} + \frac{\dot{q}}{4k_r}r_{out}^2 - \left[\left(-\frac{r_{in}}{k_r}\right)\left(q''_{in} - \frac{\dot{q}r_{in}}{2}\right)\right]}\ln(r_{out})$

$$\boxed{T(r) = -\frac{\dot{q}}{4k_r}r^2 + \left[\left(-\frac{r_{in}}{k_r}\right)\left(q''_{in} - \frac{\dot{q}r_{in}}{2}\right)\right]\ln(r) + \left[T_{out} + \frac{\dot{q}}{4k_r}r_{out}^2 - \left(\left(-\frac{r_{in}}{k_r}\right)\left(q''_{in} - \frac{\dot{q}r_{in}}{2}\right)\right)\ln(r_{out})\right]}$$

(5) * Jupyter Notebook *

Problem 5

① $Bi = \frac{hL_c}{k} \qquad L_c = \frac{r_o}{2} = \frac{D}{4}$



② $\dot{E}_{in} - \dot{E}_{out} + \cancel{\dot{E}_{gen}}^{\,0} = \dot{E}_{st}$

$$q_x - q_{x+dx} - q_{conv} = \frac{\partial}{\partial t}(\rho C_p \overset{A_c dx}{\forall} T(x))$$

$$q_x - \left(q_x + \frac{\partial}{\partial x}(q_x)dx\right) - h_{conv}(dA_s)(T(x) - T_w(x)) = \rho C_p A_c dx \underset{u}{\frac{dx}{dt}} \frac{dT}{dx}$$

$$-\frac{\partial}{\partial x}(q_x)dx - h_{conv}(\pi D dx)(T(x) - T_w(x)) = \rho C_p \frac{\pi D^2}{4} dx\, u \frac{dT}{dx}$$

$$-\frac{\partial}{\partial x}\left(-k \frac{\partial T}{\partial x} \frac{\pi D^2}{4}\right)\cancel{dx} - h_{conv}(\cancel{\pi D}\,\cancel{dx})(T(x) - T_w(x)) = \rho C_p \frac{\pi D^2}{4}\cancel{dx}\, u \frac{dT}{dx}$$

$$-\frac{\partial}{\partial x}\left(-k \frac{\partial T}{\partial x} \frac{D}{4}\right) - h_{conv}(T(x) - T_w(x)) = \rho C_p \frac{D}{4} u \frac{dT}{dx} \to \rho C_p = \frac{k}{\alpha}$$

$$\frac{kD}{4}\left(\frac{d^2T}{dx^2}\right) - h_{conv}(T(x) - T_w(x)) = \frac{k}{\alpha}\frac{D}{4} u \frac{dT}{dx}$$

$$\frac{kD}{4}\left(\frac{d^2T}{dx^2}\right) - \frac{k}{\alpha}\frac{D}{4} u \frac{dT}{dx} - \frac{k_g}{th}(T(x) - T_w(x)) = 0$$

$$\frac{d^2T}{dx^2} - \frac{u}{\alpha}\frac{dT}{dx} - \frac{4k_g}{kD\,th}(T(x) - T_w(x)) = 0$$

$a_1 = \frac{u}{\alpha}$

$a_2 = -\frac{4k_g}{kD\,th}$

B.C.

$T(x=0) - T_{wo} = 0$

$T(x=\infty) - T_f = 0$

* Jupyter Notebook *

In [ ]: