ANSWER TO THE QUESTION NO. 3

```
In [16]:  import numpy as np
          import matplotlib.pyplot as plt
```

Initialization Variable

```
In [17]:  Tinitial = 0
          Tfinal = 10
          delT = 0.1
```

```
In [18]:  arraySize = int ((Tfinal - Tinitial)/delT)
          arraySize
```

```
Out[18]:  100
```

```
In [19]:  x = np.zeros(int(arraySize))
          y = np.zeros(int(arraySize))
          theta = np.ones(int(arraySize))
          x, y , theta
```

```
Out[19]:  (array([0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
                  0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
                  0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
                  0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
                  0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
                  0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.]),
           array([0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
                  0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
                  0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
                  0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
                  0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
                  0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.]),
           array([1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1.,
                  1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1.,
                  1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1.,
                  1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1.,
                  1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1.,
                  1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1.]))
```

```
In [20]:  r = 0.1
          l = 1
          wl = np.ones(100)
          wr = np.ones(100)
          wl , wr
```

```
Out[20]: (array([1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1.,
                  1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1.,
                  1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1.,
                  1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1.,
                  1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1.,
                  1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1.]),
          array([1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1.,
                  1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1.,
                  1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1.,
                  1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1.,
                  1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1.,
                  1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1., 1.]))
```

Condition

```
In [21]: for i in range(100):

             wr[i] = 1
             if i>=5 and i<=15:
                 wr[i] = 12
             elif i>=20 and i<=40:
                 wr[i] = 12
             elif i>=40 and i<=60:
                 wl[i] = 12
             elif i>=60 and i<=80:
                 wl[i] = 12


         wl, wr
```

```
Out[21]: (array([ 1.,  1.,  1.,  1.,  1.,  1.,  1.,  1.,  1.,  1.,  1.,  1.,  1.,
                   1.,  1.,  1.,  1.,  1.,  1.,  1.,  1.,  1.,  1.,  1.,  1.,
                   1.,  1.,  1.,  1.,  1.,  1.,  1.,  1.,  1.,  1.,  1.,  1.,
                   1.,  1., 12., 12., 12., 12., 12., 12., 12., 12., 12., 12.,
                  12., 12., 12., 12., 12., 12., 12., 12., 12., 12., 12., 12.,
                  12., 12., 12., 12., 12., 12., 12., 12., 12., 12., 12., 12.,
                  12., 12., 12.,  1.,  1.,  1.,  1.,  1.,  1.,  1.,  1.,  1.,
                   1.,  1.,  1.,  1.,  1.,  1.,  1.,  1.,  1.]),
          array([ 1.,  1.,  1.,  1.,  1., 12., 12., 12., 12., 12., 12., 12., 12.,
                  12., 12., 12.,  1.,  1.,  1.,  1., 12., 12., 12., 12., 12., 12.,
                  12., 12., 12., 12., 12., 12., 12., 12., 12., 12., 12., 12.,
                  12., 12.,  1.,  1.,  1.,  1.,  1.,  1.,  1.,  1.,  1.,  1.,  1.,
                   1.,  1.,  1.,  1.,  1.,  1.,  1.,  1.,  1.,  1.,  1.,  1.,
                   1.,  1.,  1.,  1.,  1.,  1.,  1.,  1.,  1.,  1.,  1.,  1.,
                   1.,  1.,  1.,  1.,  1.,  1.,  1.,  1.,  1.,  1.,  1.,  1.,
                   1.,  1.,  1.,  1.,  1.,  1.,  1.,  1.,  1.]))
```

Main Program

```
In [22]: for i in range(99):
             x[i+1] =  x[i] + ((r/2) *(wl[i]+wr[i])*(np.cos(theta[i]))) *delT
             y[i+1] =  y[i] + ((r/2) *(wl[i]+wr[i])*(np.sin(theta[i]))) * delT
             theta[i+1] =  theta[i] + ((r/l) *(wr[i]-wl[i]))* delT


         x, y, theta
```

Out[22]:
```
(array([ 0.        ,  0.00540302,  0.01080605,  0.01620907,  0.02161209,
         0.02701512,  0.06213477,  0.09103776,  0.11337474,  0.12887568,
         0.13735322,  0.13870489,  0.13291433,  0.12005156,  0.10027205,
         0.07381489,  0.04099989,  0.03503433,  0.02906876,  0.0231032 ,
         0.01713763, -0.02163854, -0.06590717, -0.11513315, -0.16872145,
        -0.22602429, -0.28634902, -0.34896643, -0.41311963, -0.47803314,
        -0.5429223 , -0.60700274, -0.66949988, -0.72965825, -0.78675067,
        -0.84008703, -0.8890226 , -0.93296586, -0.97138564, -1.00381751,
        -1.02986946, -1.04922657, -1.06165485, -1.08101195, -1.1070639 ,
        -1.13949578, -1.17791555, -1.22185881, -1.27079438, -1.32413074,
        -1.38122317, -1.44138154, -1.50387867, -1.56795911, -1.63284827,
        -1.69776178, -1.76191498, -1.8245324 , -1.88485712, -1.94215997,
        -1.99574826, -2.04497424, -2.08924287, -2.12801905, -2.16083404,
        -2.1872912 , -2.20707071, -2.21993349, -2.22572404, -2.22437238,
        -2.21589483, -2.20039389, -2.17805692, -2.14915392, -2.11403427,
        -2.07312249, -2.02691311, -1.9759647 , -1.92089312, -1.86236405,
        -1.801085  , -1.73779669, -1.7278686 , -1.71794051, -1.70801243,
        -1.69808434, -1.68815625, -1.67822817, -1.66830008, -1.65837199,
        -1.64844391, -1.63851582, -1.62858774, -1.61865965, -1.60873156,
        -1.59880348, -1.58887539, -1.5789473 , -1.56901922, -1.55909113]),
 array([0.        , 0.00841471, 0.01682942, 0.02524413, 0.03365884,
        0.04207355, 0.09676916, 0.15498958, 0.21603104, 0.27915568,
        0.34360047, 0.40858642, 0.47332798, 0.53704257, 0.59896002,
        0.65833189, 0.7144405 , 0.72246621, 0.73049192, 0.73851763,
        0.74654334, 0.79871046, 0.8463055 , 0.88875315, 0.92554031,
        0.95622229, 0.98042823, 0.99786552, 1.00832339, 1.01167543,
        1.00788111, 0.9969863 , 0.9791227 , 0.95450623, 0.92343446,
        0.88628297, 0.84350085, 0.79560524, 0.74317509, 0.68684416,
        0.62729337, 0.56524257, 0.5014418 , 0.439391  , 0.37984021,
        0.32350928, 0.27107913, 0.22318351, 0.18040139, 0.14324991,
        0.11217814, 0.08756167, 0.06969807, 0.05880326, 0.05500894,
        0.05836097, 0.06881884, 0.08625614, 0.11046207, 0.14114406,
        0.17793121, 0.22037886, 0.26797391, 0.32014103, 0.37624964,
        0.4356215 , 0.49753896, 0.56125355, 0.62599511, 0.69098105,
        0.75542584, 0.81855049, 0.87959195, 0.93781236, 0.99250798,
        1.04301764, 1.0887308 , 1.12909489, 1.16362199, 1.19189475,
        1.21357141, 1.22838995, 1.22958707, 1.2307842 , 1.23198132,
        1.23317844, 1.23437556, 1.23557269, 1.23676981, 1.23796693,
        1.23916405, 1.24036117, 1.2415583 , 1.24275542, 1.24395254,
        1.24514966, 1.24634678, 1.24754391, 1.24874103, 1.24993815]),
 array([1.  , 1.  , 1.  , 1.  , 1.  , 1.  , 1.11, 1.22, 1.33, 1.44, 1.55,
        1.66, 1.77, 1.88, 1.99, 2.1 , 2.21, 2.21, 2.21, 2.21, 2.21, 2.32,
        2.43, 2.54, 2.65, 2.76, 2.87, 2.98, 3.09, 3.2 , 3.31, 3.42, 3.53,
        3.64, 3.75, 3.86, 3.97, 4.08, 4.19, 4.3 , 4.41, 4.52, 4.41, 4.3 ,
        4.19, 4.08, 3.97, 3.86, 3.75, 3.64, 3.53, 3.42, 3.31, 3.2 , 3.09,
        2.98, 2.87, 2.76, 2.65, 2.54, 2.43, 2.32, 2.21, 2.1 , 1.99, 1.88,
        1.77, 1.66, 1.55, 1.44, 1.33, 1.22, 1.11, 1.  , 0.89, 0.78, 0.67,
        0.56, 0.45, 0.34, 0.23, 0.12, 0.12, 0.12, 0.12, 0.12, 0.12, 0.12,
        0.12, 0.12, 0.12, 0.12, 0.12, 0.12, 0.12, 0.12, 0.12, 0.12, 0.12,
        0.12]))
```

In [23]:
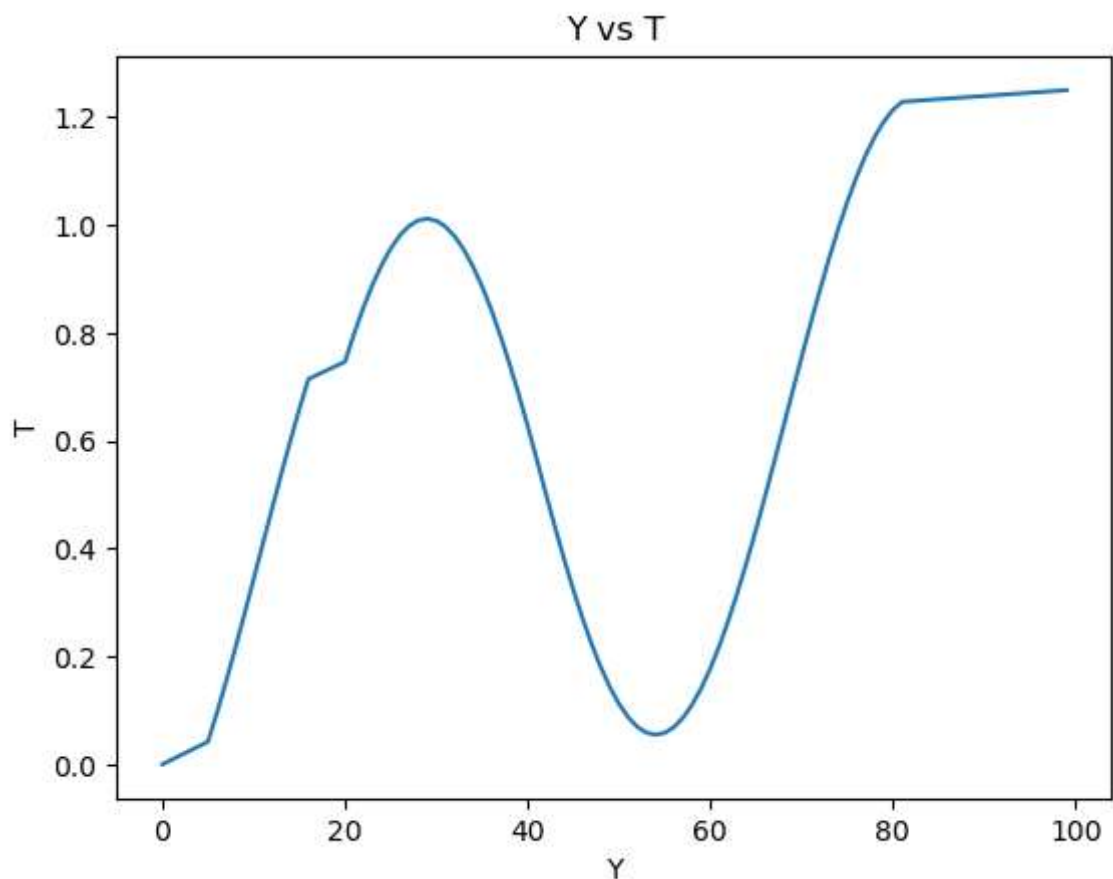```python
t= np.arange(100)

plt.plot(t, x)
plt.title('X vs T')
plt.xlabel('X')
plt.ylabel('T')
```
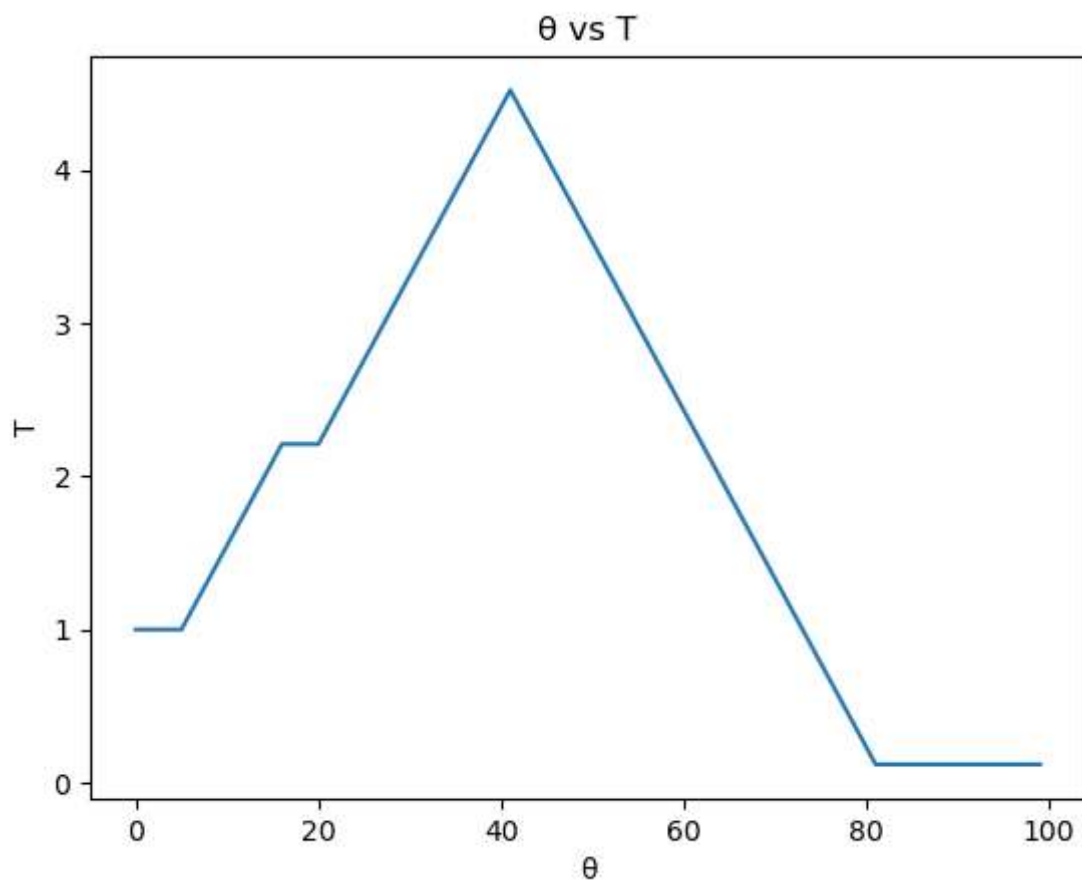
Out[23]:
```
Text(0, 0.5, 'T')
```

## X vs T
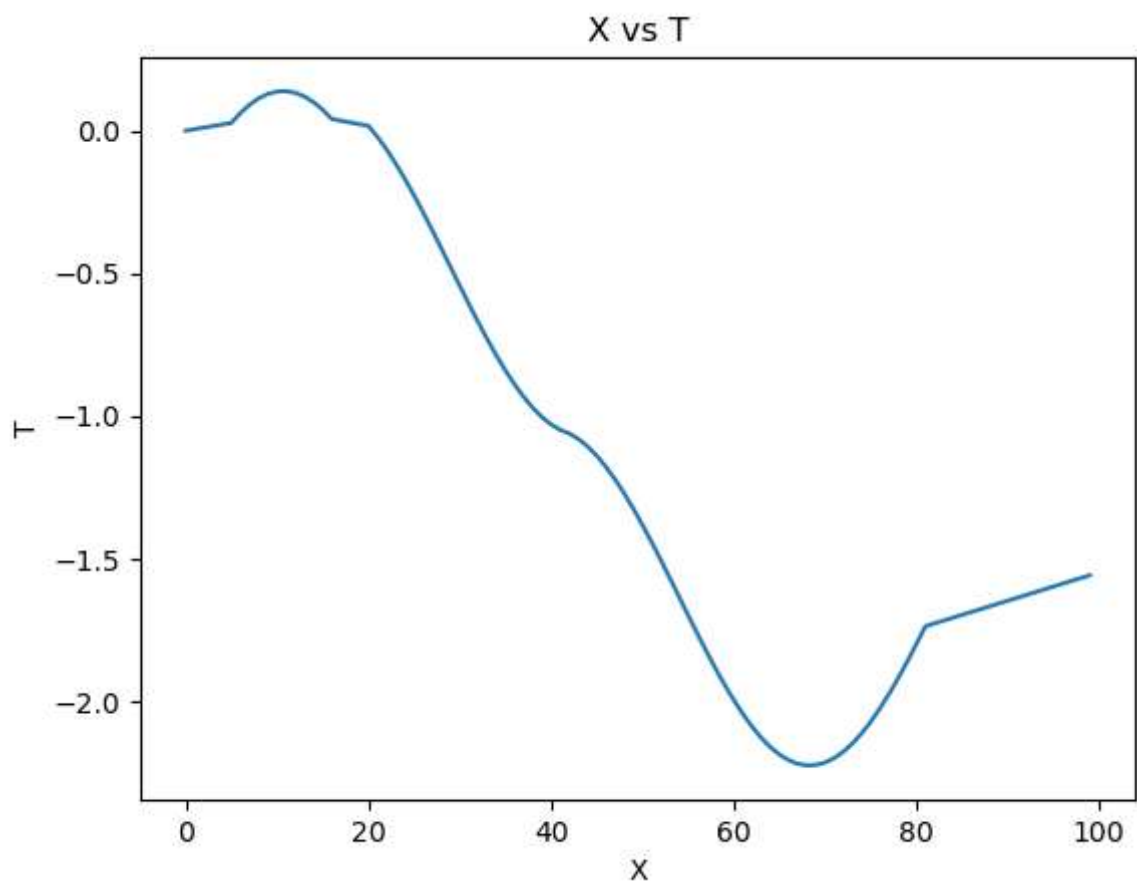


```
In [24]:  plt.plot(t, y)
          plt.title('Y vs T')
          plt.xlabel('Y')
          plt.ylabel('T')
```

Out[24]: Text(0, 0.5, 'T')

Y vs T

In [25]:
```python
plt.plot(t, theta)
plt.title('θ vs T')
plt.xlabel('θ')
plt.ylabel('T')
```

Out[25]: Text(0, 0.5, 'T')

In [26]:
```python
plt.plot(x,y)
plt.title('X vs Y')
plt.xlabel('X')
plt.ylabel('Y')
```

Out[26]: Text(0, 0.5, 'Y')

X vs Y