

```
1 import matplotlib.pyplot as plt
2 import random
3 import numpy as np
4
5 class Node:
6     def __init__(self, num, transmitStatus = False, numOfPackets = 1):
7         self.num = num
8         self.transmitStatus = transmitStatus
9         self.numOfPackets = numOfPackets
10
11 def RandomStatus(n):
12     if random.randint(1,n) == 1:
13         return True
14     return False
15
16 efficiency = []
17 numOfNodes = 50
18 AveragingNum = 50
19
20 for i in range(1, numOfNodes):
21     countSuccess=0
22     countCollision=0
23     countEmpty=0
24     listOfNodes = []
25     AveragingEff = 0
26     for j in range(AveragingNum):
27         for j in range(i):
28             listOfNodes.append( Node(j+1))
29             NumOfSlots = 100000
30             for k in range(NumOfSlots):
31                 NumTrue = 0
32                 for node in listOfNodes:
33                     if(node.numOfPackets!=0):
34                         node.transmitStatus = RandomStatus(i)
35                         # print(node.num, node.transmitStatus, node.numOfPackets)
36                         if(node.transmitStatus==True):
37                             NumTrue+=1
38                 if(NumTrue==0):
39                     # print("Empty")
40                     countEmpty+=1
41                 elif(NumTrue==1):
42                     for node in listOfNodes:
43                         if(node.transmitStatus==True):
44                             node.numOfPackets-=1
45                             node.transmitStatus=False
46                     # print("Success")
47                     countSuccess+=1
48                 else:
49                     # print("Collision")
50                     countCollision+=1
51             if(countSuccess == len(listOfNodes)):
52                 # print(len(listOfNodes))
53                 NumOfSlots=countSuccess+countEmpty+countCollision
54                 # print(i)
55                 break
56     AveragingEff += (countSuccess/NumOfSlots)
```

```
57     print(i)
58     # plt.plot(i,countSuccess/NumOfSlots)
59     # print("Success:",countSuccess, "Collision",countCollision, "Empty", countEmpty)
60     # print("Efficiency", countSuccess/NumOfSlots)
61     efficiency.append(AveragingEff/AveragingNum)
62
63
64     sum = 0
65     for s in efficiency:
66         sum+=s
67
68     print(sum/len(efficiency))
69
70
71     # x axis values
72     x = np.arange(1,numOfNodes)
73     print(x)
74     # corresponding y axis values
75     y = [2,4,1]
76
77     # plotting the points
78     plt.plot(x, efficiency)
79     # naming the x axis
80     plt.xlabel('Node - axis')
81     # naming the y axis
82     plt.ylabel('Efficiency - axis')
83     plt.axis([1, numOfNodes, 0, 1])
84     plt.grid(linestyle='--')
85     # function to show the plot
86     plt.show()
```