**Assignment 1**

**import** **matplotlib.pyplot** **as** **plt**

n\_values = []

best\_case\_comp = []

avg\_case\_comp = []

worst\_case\_comp = []

**def** linSearch(arr, keys):

comparison\_data = []

*#For every key, linsearch will be performed*

**for** key **in** keys:

no\_of\_comp = 0

flg = 0

*#Linsearch algorithm*

**for** elem **in** arr:

no\_of\_comp += 1;

**if** (elem == key):

comparison\_data.append(no\_of\_comp)

flg = 1

**break**;

**if**(flg == 0):

comparison\_data.append(no\_of\_comp)

**return** comparison\_data

*#Looping For All Ranges*

**for** i **in** range(10,105,5):

*#Saving n values in array*

n\_values.append(i)

*#Array will have elements 1 to i in every case ,where i = 10,15,20,25...*

arr\_of\_elements = list(range(1,i+1));

*#Keys one more than no. of elements in array*

key\_to\_search = arr\_of\_elements + [i+1];

*#Comparision data for current input range*

res = linSearch(arr\_of\_elements, key\_to\_search)

best\_case\_comp.append(min(res))

worst\_case\_comp.append(max(res))

avg\_case\_comp.append((sum(res)/len(res)))

*#Plotting Data*

plt.figure(figsize = (9,7))

plt.plot(n\_values,best\_case\_comp,'g', linestyle="dashed",markerfacecolor='yellow',markersize=5, marker='o')

plt.plot(n\_values,worst\_case\_comp,'r',linestyle="dashed",markerfacecolor='yellow',markersize=5, marker='o')

plt.plot(n\_values,avg\_case\_comp,'b',linestyle="dashed",markerfacecolor='yellow',markersize=5, marker='o')

plt.xlabel('**\n**Input Size ( N )',fontsize=14)

plt.ylabel('No. of Comparisions**\n**',fontsize=14)

plt.title('**\n\n**Graph of Input Size vs No. of Comparisions**\n**',fontsize=18, fontweight='bold')

ax = plt.gca()

legend = ax.legend(['Best Case', 'Worst Case', ' Avg, Case'],fontsize=14)

