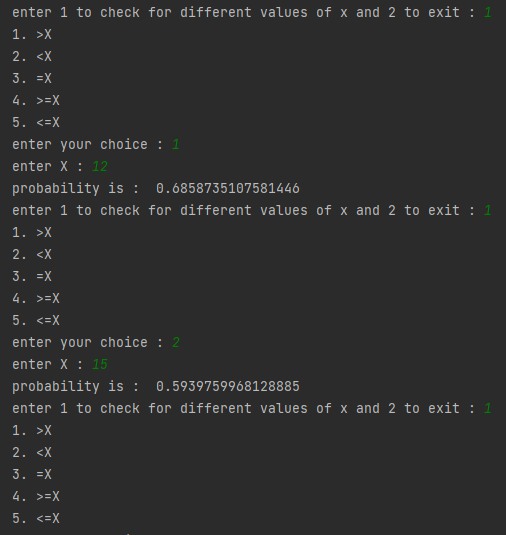
Tutorial Assignment: Distributions

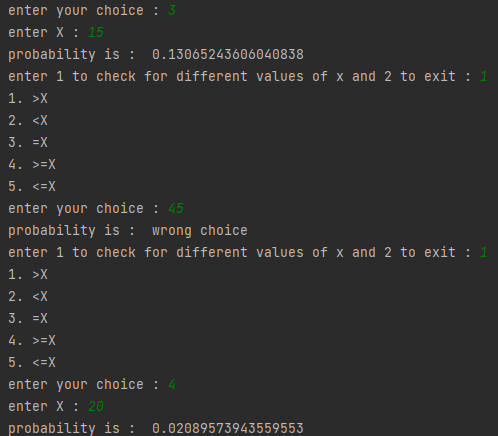
PTDA

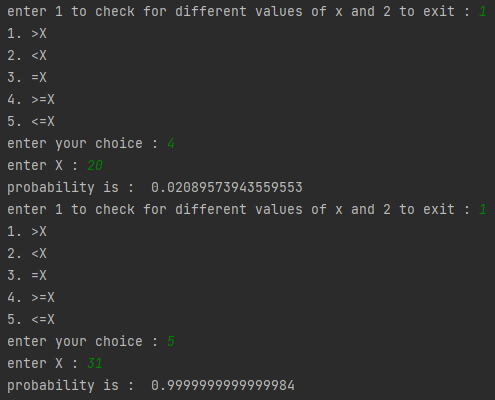
Question-1 :

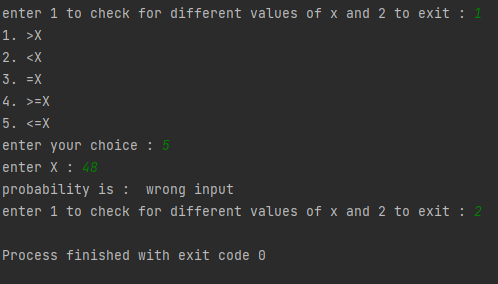
import matplotlib.pyplot as plt  
def binomialCoeff(n, k):  
 C = [0 for i in range(k + 1)]  
 C[0] = 1  
  
 for i in range(1, n + 1):  
 j = min(i, k)  
 while (j > 0):  
 C[j] = C[j] + C[j - 1]  
 j -= 1  
  
 return C[k]  
  
n=31  
p=0.447  
q=1-p  
  
def GetPMF():  
 dic={}  
 for i in range(0,n+1):  
 temp=binomialCoeff(n,i)\*pow(p,i)\*pow(q,n-i)  
 dic[i]=temp  
 return dic  
  
def GetCDF(d):  
 dic={}  
 t=0  
 for i,j in d.items():  
 t+=j  
 dic[i]=t  
 return dic  
  
# a  
PMF=GetPMF()  
CDF=GetCDF(PMF)  
  
#c  
def GetProb(PMF,CDF):  
 print('1. >X\n2. <X\n3. =X\n4. >=X\n5. <=X')  
 option=int(input('enter your choice : '))  
 if(option>5 or option<1):  
 return "wrong choice"  
 x=int(input('enter X : '))  
 if(x>31 or x<0):  
 return "wrong input"  
 if(option==1):  
 return 1-CDF[x]  
 if(option==2):  
 if(x==0):  
 return 0  
 return CDF[x-1]  
 if(option==3):  
 return PMF[x]  
 if(option==4):  
 if(x==0):  
 return 1  
 return 1-CDF[x-1]  
 return CDF[x]  
  
while(1):  
 n=int(input('enter 1 to check for different values of x and 2 to exit : '))  
 if(n!=1):  
 break  
 p=GetProb(PMF,CDF)  
 print('probability is : ',p)  
  
#graphs  
X=[i for i,j in PMF.items()]  
Y=[j for i,j in PMF.items()]  
  
plt.plot(X,Y,color="green")  
plt.xlabel("PMF")  
plt.ylabel("probability")  
plt.show()  
  
XCDF=[i for i,j in CDF.items()]  
YCDF=[j for i,j in CDF.items()]  
  
plt.plot(XCDF,YCDF,color="red")  
plt.xlabel("number of students")  
plt.ylabel("CDF")  
plt.show()

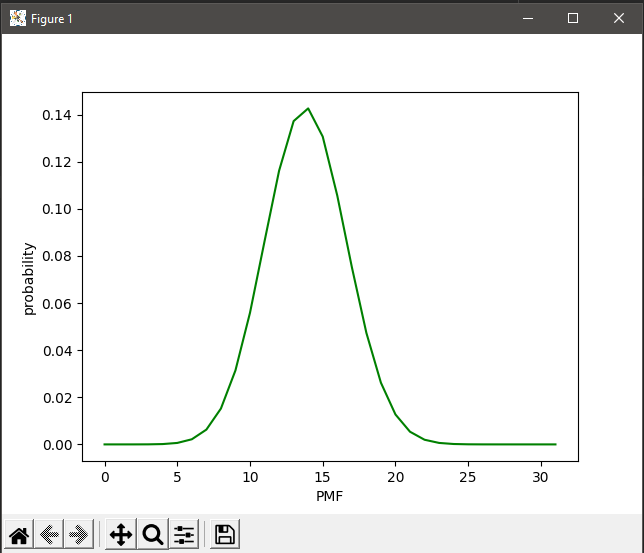
OUTPUT:

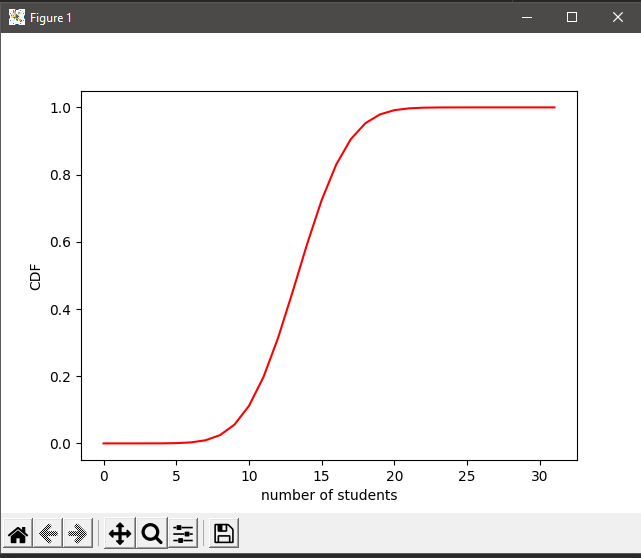












Question-2 :

import matplotlib.pyplot as plt  
import math  
n= 500  
p=0.001  
q=1-p  
lambdaa=n\*p  
  
def GetPMF():  
 dic={}  
 for i in range(0,n+1):  
 temp=pow(lambdaa,i)\*pow(math.e,-lambdaa)  
 for j in range(1,i+1):  
 temp/=j  
 dic[i]=temp  
 return dic  
  
def GetCDF(d):  
 dic={}  
 t=0  
 for i,j in d.items():  
 t+=j  
 dic[i]=t  
 return dic  
  
PMF=GetPMF()  
CDF=GetCDF(PMF)  
  
NoDef=pow(100\*p,0)\*pow(math.e,-100\*p)  
AtleastOneDef=1-NoDef  
print('No Defective : ',NoDef)  
print('Atleast One Defective : ',AtleastOneDef)  
x=int(input('enter X : '))  
print('probability of ',x ,' is ', PMF[x])  
  
#graphs  
X=[i for i,j in PMF.items()]  
Y=[j for i,j in PMF.items()]  
  
plt.plot(X,Y,color="green")  
plt.xlabel("PMF")  
plt.ylabel("probability")  
plt.show()  
  
XCDF=[i for i,j in CDF.items()]  
YCDF=[j for i,j in CDF.items()]  
  
plt.plot(XCDF,YCDF,color="red")  
plt.xlabel("number of students")  
plt.ylabel("CDF")  
plt.show()

OUTPUT:

