PROBABILITY AND STATISTICS LAB

ETIE-254

MAHARAJA AGRASEN
INSTITUTE OF TECHNOLOGY
MAIT

Faculty Name: Dr. K.C. Tripathi

Submitted by : Khushi
Rollno : 03114813120
Semester : 4th
Branch : ITE
Group : 2

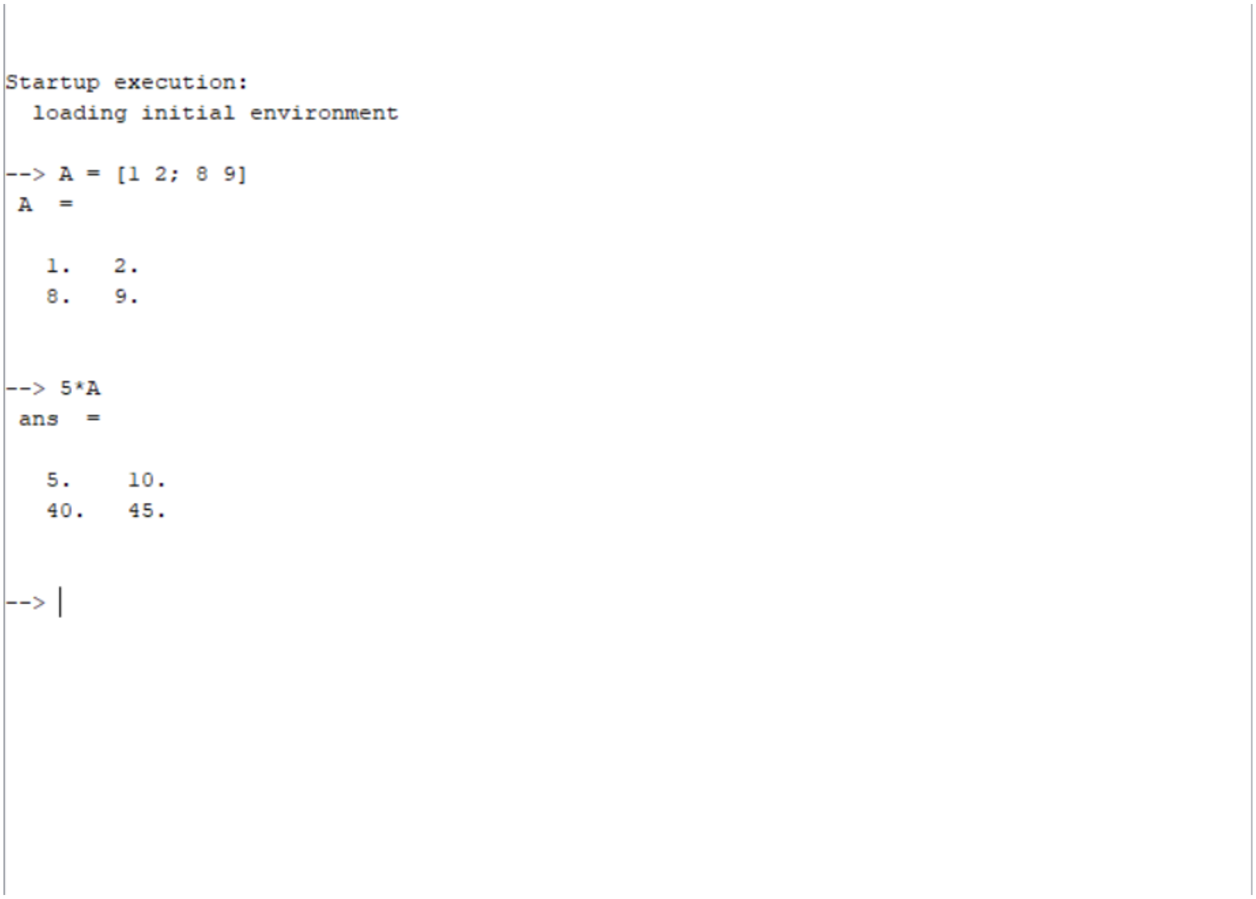
PRACTICAL RECORD

S.No.	Experiment	Date of Submission	Teacher’s Signature	Marks/ Remarks
1.	Installation of Scilab and demonstration of simple programming concepts like marix multiplication (scalar and vector), loop, conditional statements and plotting.			
2.	Program for demonstration of theoretical probability limits.			
3.	Program to plot normal distributions and exponential distributions for various parametric values.			
4.	Fitting of binomial distributions for given n and p.			
5.	Fitting of binomial distributions after computing mean and variance.			

EXPERIMENT - 1

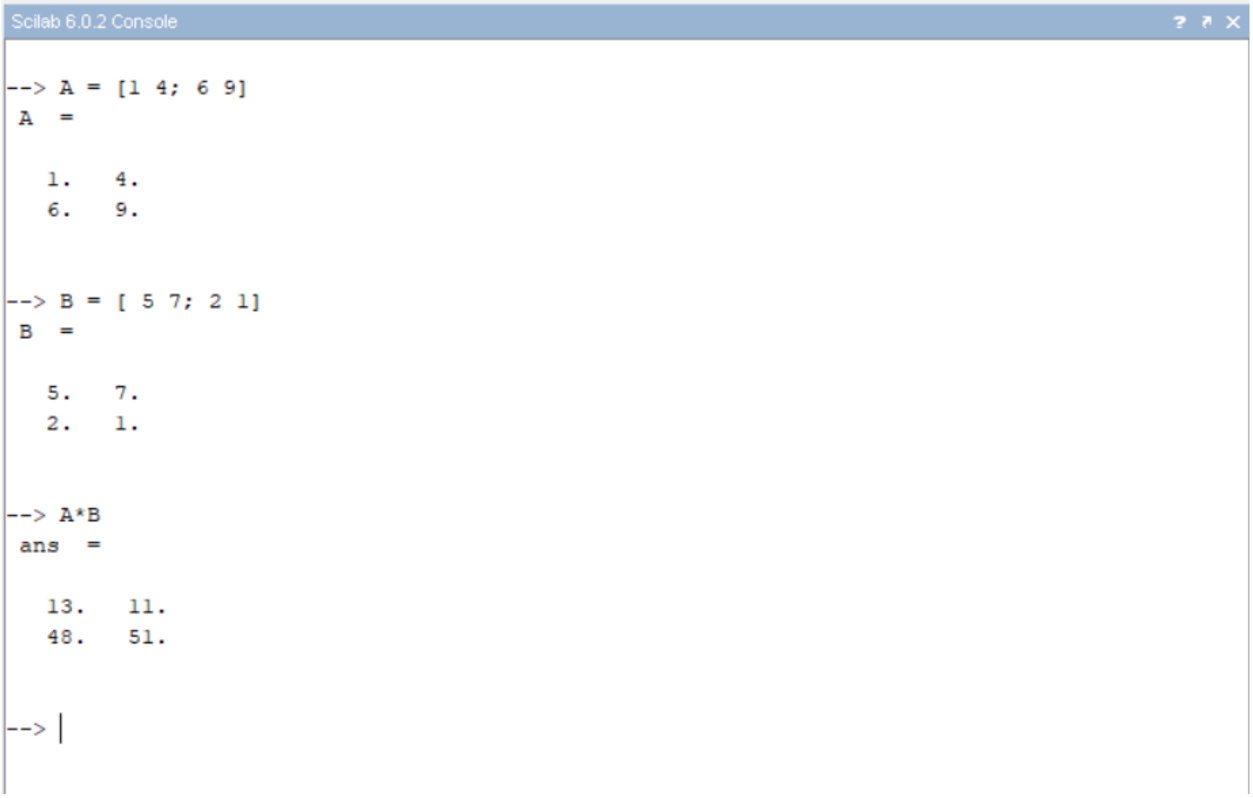
AIM : Installation of Scilab and demonstration of simple programming concepts like marix multiplication (scalar and vector), loop, conditional statements and plotting.

i) Multiplication of a matrix with a scalar.



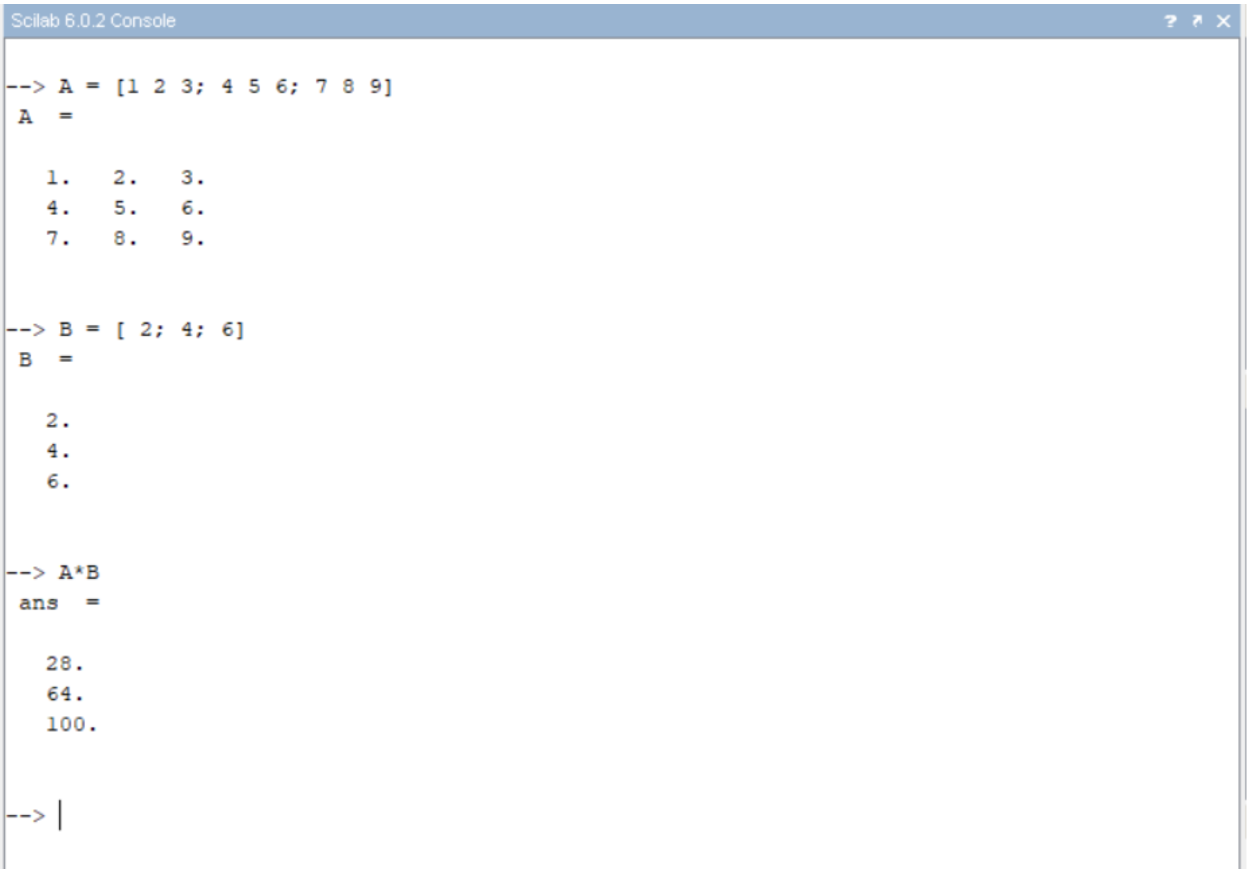
```
Startup execution:
  loading initial environment

--> A = [1 2; 8 9]
 A  =

   1.   2.
   8.   9.


--> 5*A
 ans  =

   5.    10.
   40.   45.


--> |
```

ii) Multiplication of a matrix with a matrix (Scalar Product)



```
--> A = [1 4; 6 9]
 A  =

   1.   4.
   6.   9.


--> B = [ 5 7; 2 1]
 B  =

   5.   7.
   2.   1.


--> A*B
 ans  =

   13.   11.
   48.   51.


-->
```

iii) Multiplication of a matrix with a matrix (Vector Product)



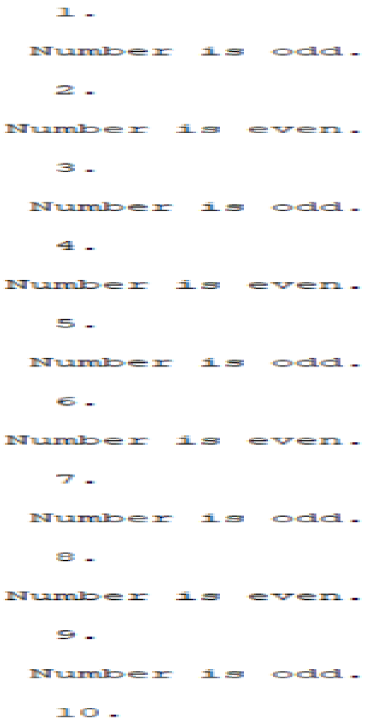
```
--> A = [1 2 3; 4 5 6; 7 8 9]
 A  =

   1.   2.   3.
   4.   5.   6.
   7.   8.   9.


--> B = [ 2; 4; 6]
 B  =

   2.
   4.
   6.


--> A*B
 ans  =

   28.
   64.
   100.


--> 
```

iv) Loop 100 times and print if the number is odd or even.

Code :

```
i=1;
while(i<=100)
disp(i)
    if modulo(i,2)==0
    then
    disp('The number is even.');
 else
    disp('The number is odd.');
     end
 i=i+1;
 End
```

Output:



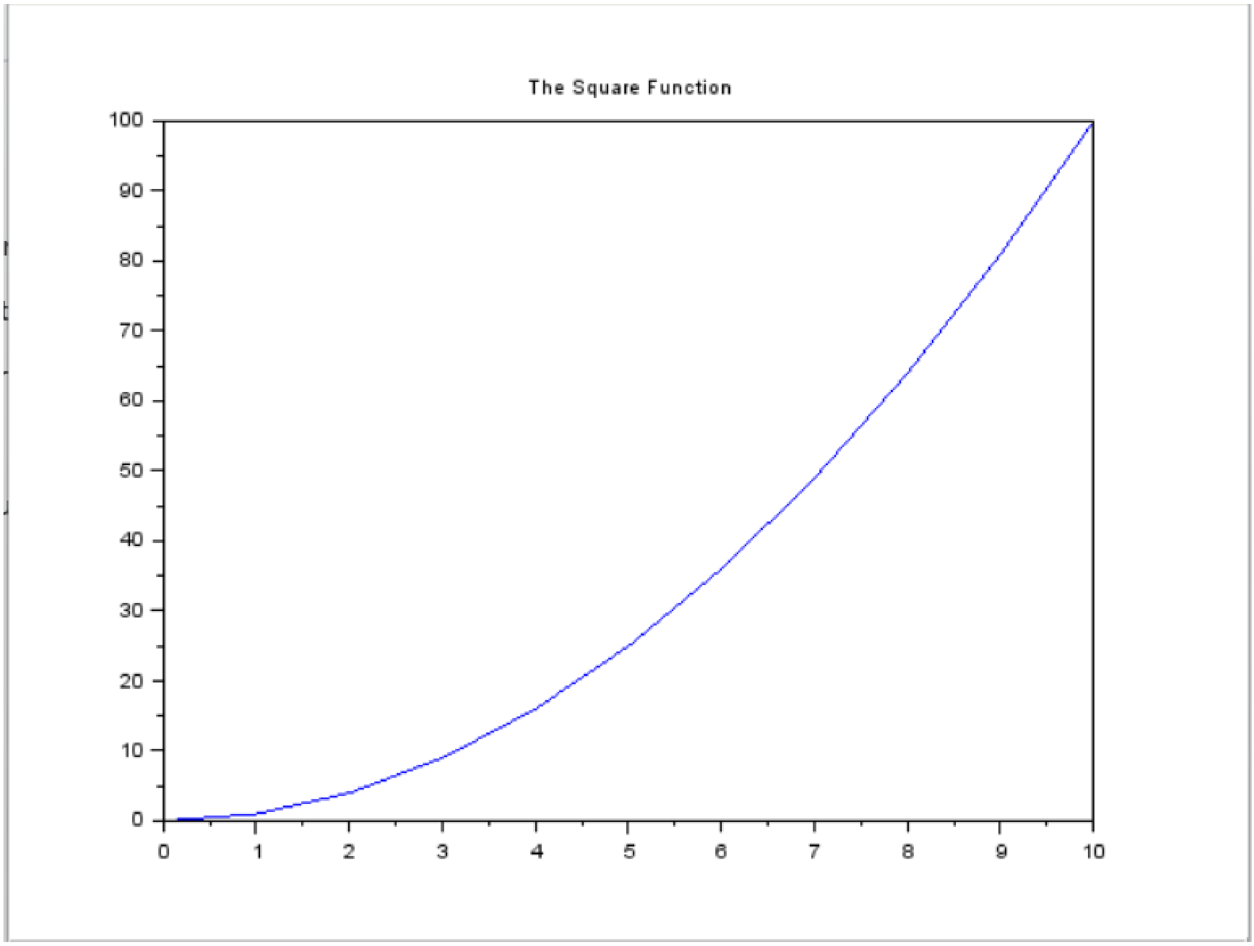
```
     1.
   Number is odd.
     2.
 Number is even.
     3.
   Number is odd.
     4.
 Number is even.
     5.
   Number is odd.
     6.
 Number is even.
     7.
   Number is odd.
     8.
 Number is even.
     9.
   Number is odd.
     10.
```

v) Plot (2D) The Square Function

Code:

```
//Plot 2d the square function
clc
x = [0 1 2 3 4 5 6 7 8 9 10];
y = [0 1 4 9 16 25 36 49 64 81 100];
plot (x,y);
title('The Square Function')
```

OUTPUT:



EXPERIMENT - 2

AIM: Program for demonstration of theoretical probability limits.

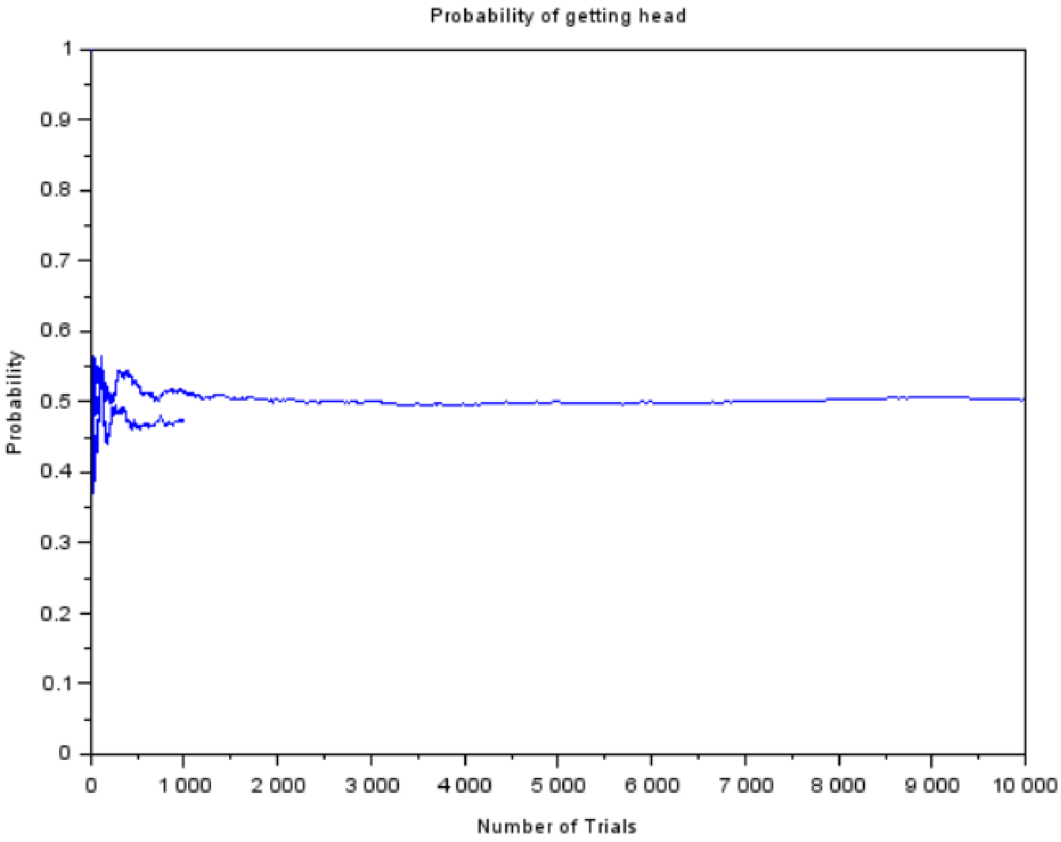
Code:

```
clc
clear all; clf;
```

```
a1 = 10;
counth = 0;
for i=1:a1
    x = round(rand(1));
    if (x==1)
        counth = counth+1;
end
p(i)= counth/i;
end
plot(1:a1,p)
xlabel('Number of Trials');
ylabel('Probability');
title('Probability of getting head');

a1 = 100;
counth = 0;
for i=1:a1
    x=round(rand(1));
    if(x==1)
        counth = counth+1;
end
p(i)= counth/i;
end
plot(1:a1,p)
xlabel('Numeber of Trials');
ylabel('Probability');
title('Probability of getting head');

a1 = 1000;
counth = 0;
for i=1:a1
    x=round(rand(1));
    if(x==1)
        counth = counth+1;
```

```
end
p(i)= counth/i;
end
plot(1:a1,p)
xlabel('Number of Trials');
ylabel('Probability');
title('Probability of getting head');

a1 = 10000;
counth = 0;
for i=1:a1
    x=round(rand(1));
    if(x==1)
        counth = counth+1;
end
p(i)= counth/i;
end
plot(1:a1,p)
xlabel('Number of Trials');
ylabel('Probability');
title('Probability of getting head');
```

OUTPUT:



EXPERIMENT - 3

AIM: Program to plot normal distributions and exponential distributions for various parametric values.

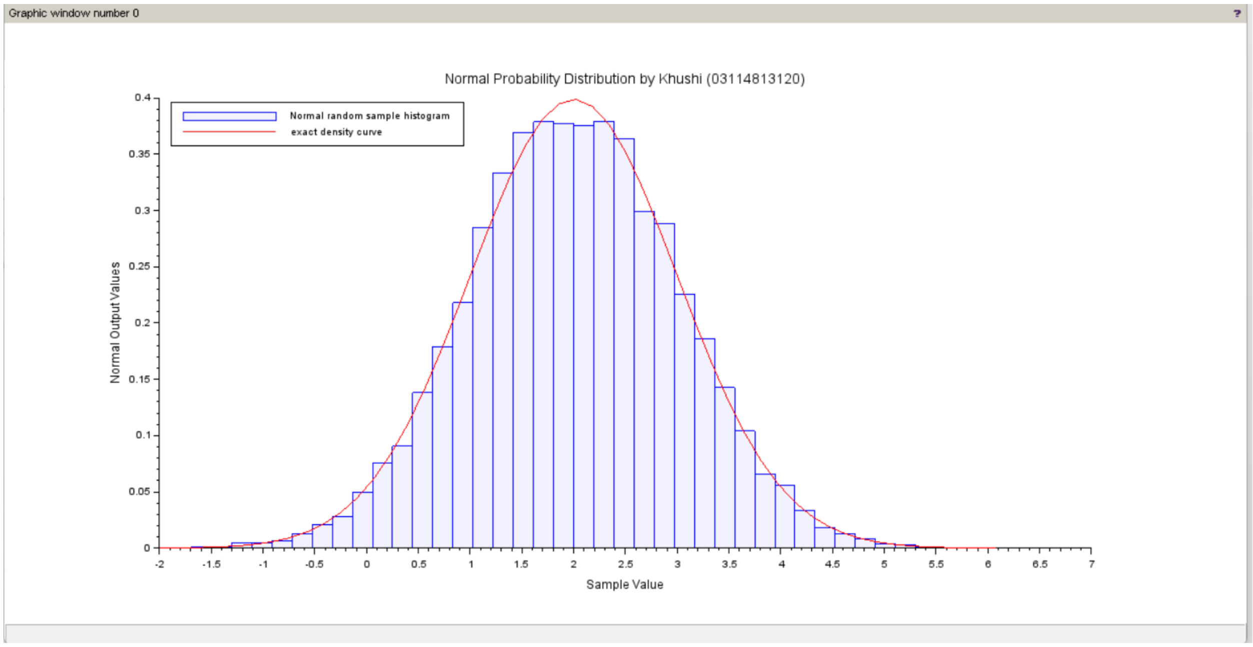
EXPERIMENT - 3

AIM : Program to plot normal distributions and exponential distributions for various parametric values.

Code for Normal Distribution :

```
clc;
clear all;
clf();
m=2;
sd=1;
vari=sd^2;
x=grand(10000,1,"nor", m,sd);
xmax= max(x);
clf();
histplot(40,x,style=2)
x = linspace(-10,max(xmax),100)';
plot2d(x,(1/(sqrt(2*%pi*vari))*exp((-0.5)*(x-m)^2/vari)),strf="000", style=5)
xlabel('Sample Value');
ylabel('Normal Output Values');
title('Normal Probability Distribution by Khushi (03114813120)');
legend(["Normal random sample histogram" "exact density curve"],2);
```

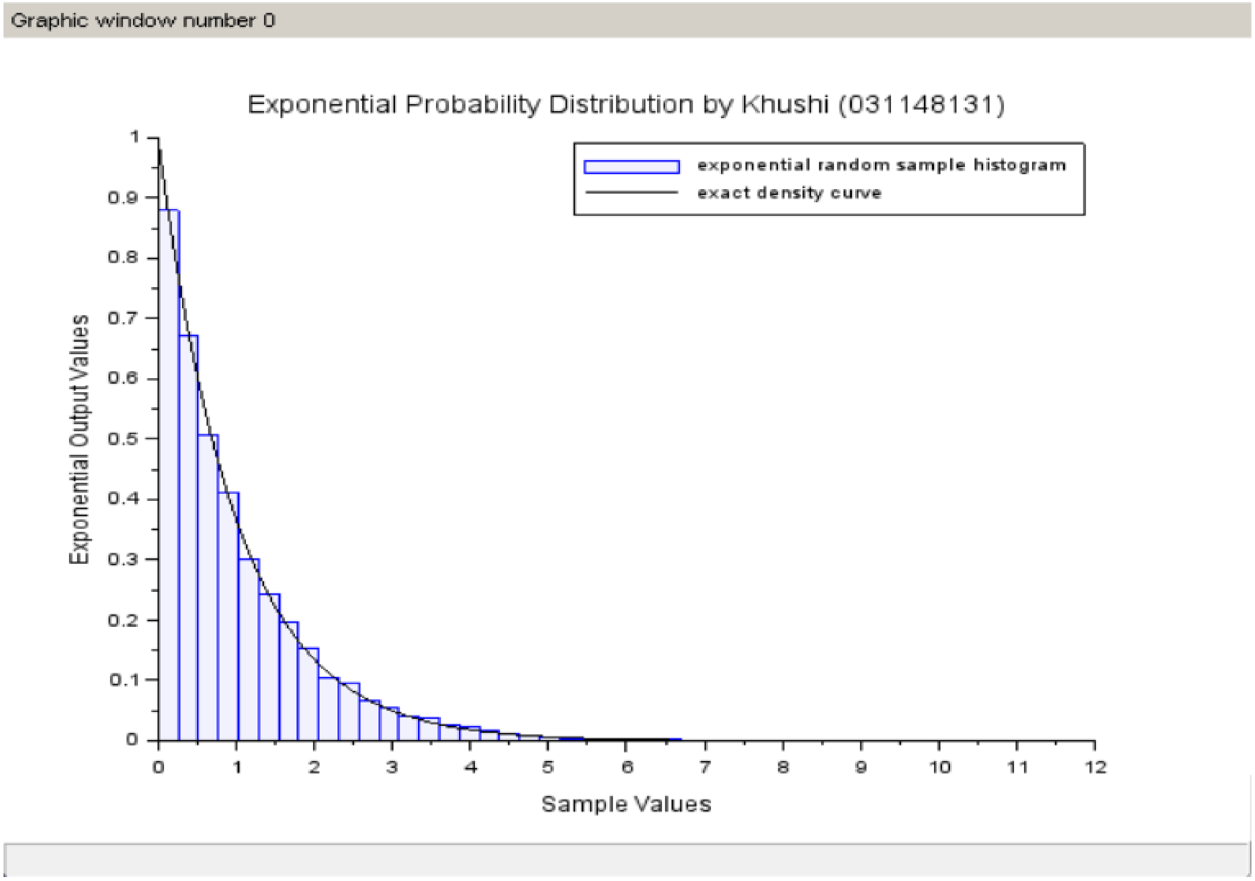
OUTPUT :


Graphic window number 0
Normal Probability Distribution by Khushi (03114813120)
Normal random sample histogram
exact density curve
Normal Output Values
Sample Value
0.4
0.35
0.3
0.25
0.2
0.15
0.1
0.05
0
-2
-1.5
-1
-0.5
0
0.5
1
1.5
2
2.5
3
3.5
4
4.5
5
5.5
6
6.5
7

Code for Exponential Distribution :

```
clc
alpha=1
x=grand(10000,1,"exp",alpha)
Xmax = max(x) ;
histplot (40 ,x, style =2)
x = linspace (0,max(Xmax),100);
plot2d (x , alpha*exp(-alpha*x))
legend (['exponential random sample histogram' ' exact density curve' ]) ;
xtitle('Exponential Probability Distribution by Khushi (031148131)','Sample
Values','Exponential Output Values')
```

OUTPUT :



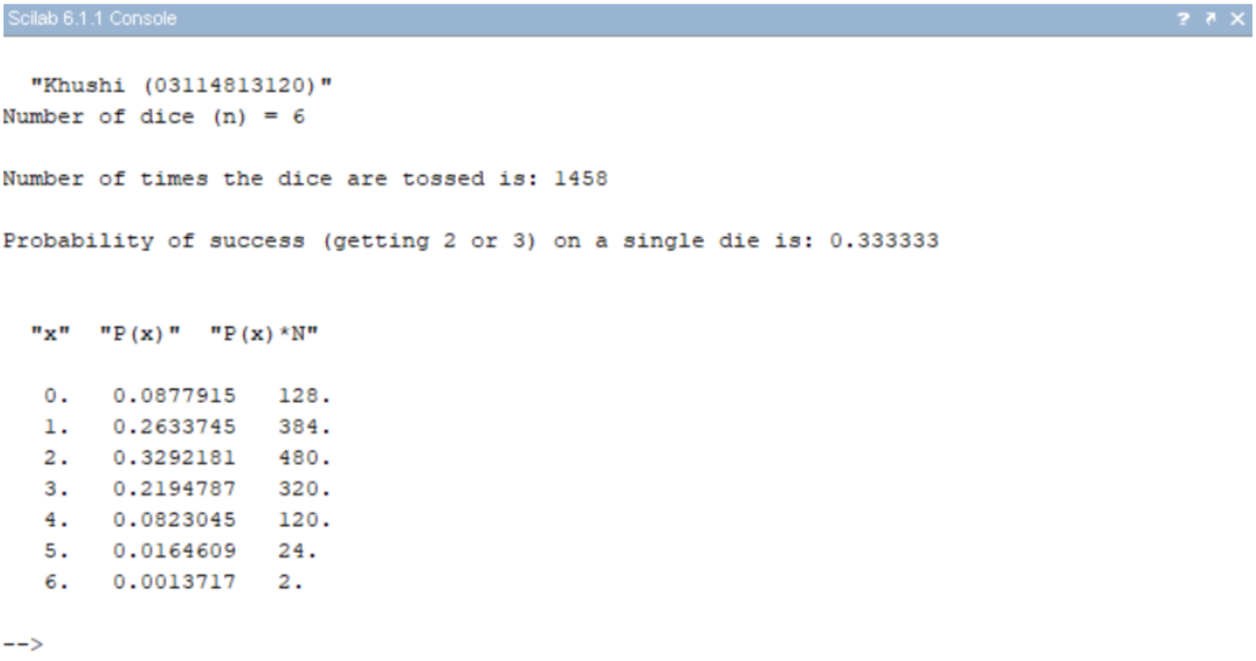
EXPERIMENT - 4

AIM : Fitting of binomial distributions for given n and p.

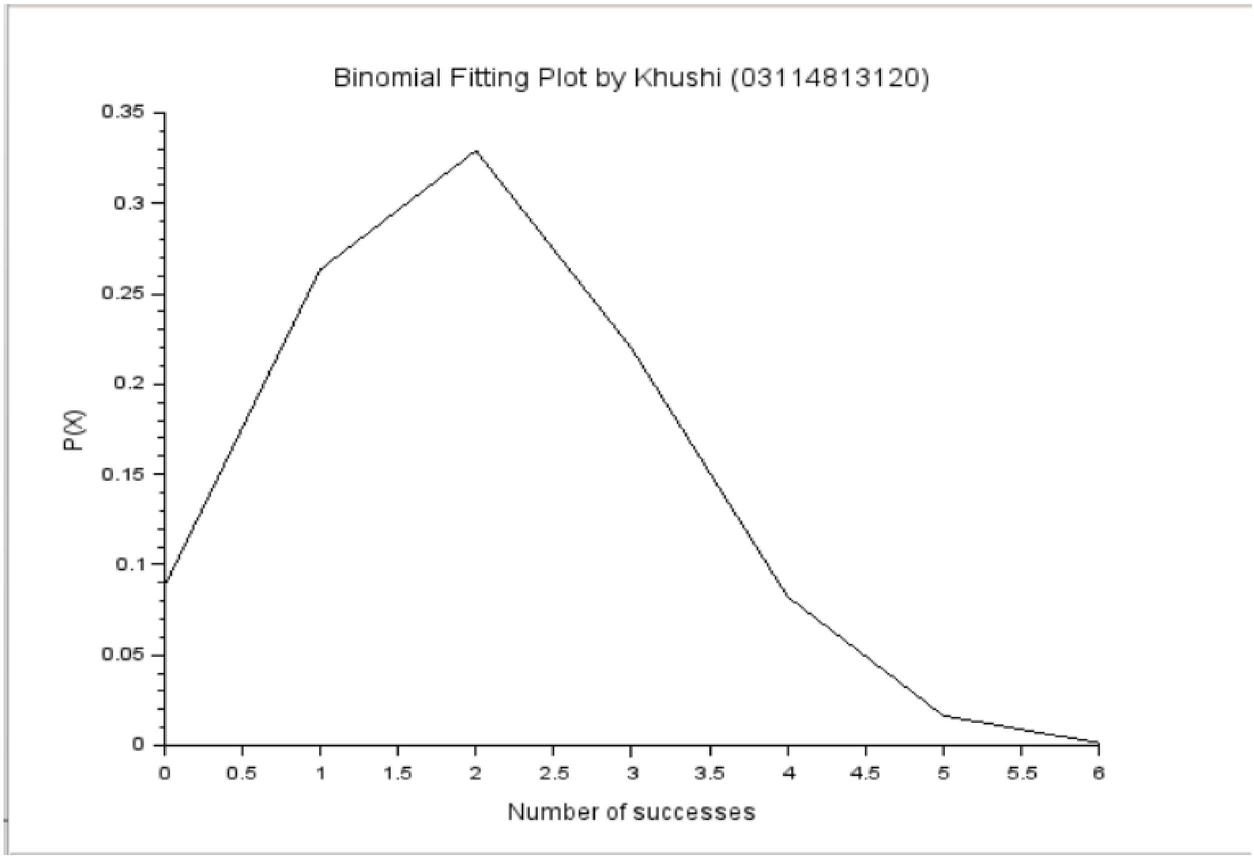
Code:

```
clc
disp('Khushi (03114813120)')
n=6;
p=1/3;
N=1458;
mprintf('Number of dice (n) = %i \n \n',n)
mprintf('Number of times the dice are tossed is: %i \n \n',N)
mprintf('Probability of success (getting 2 or 3) on a single die is: %f
\n \n',p)
for i=0:n
x(1,i+1)=i
ans= nchoosek(n,i)*(p^i)*((1-p)^(n-i));
prob(1,i+1)=ans
expected(1,i+1)=ans*N;
end
disp(['x' 'P(x)' 'P(x)*N'],[x' prob' expected'])
plot2d(x,prob)
xtitle('Binomial Fitting Plot by Khushi (03114813120)','Number of
successes','P(X)')
```

OUTPUT :



```
  "Khushi (03114813120)"
Number of dice (n) = 6

Number of times the dice are tossed is: 1458

Probability of success (getting 2 or 3) on a single die is: 0.333333


  "x"   "P(x)"   "P(x)*N"

   0.   0.0877915   128.
   1.   0.2633745   384.
   2.   0.3292181   480.
   3.   0.2194787   320.
   4.   0.0823045   120.
   5.   0.0164609   24.
   6.   0.0013717   2.

-->
```



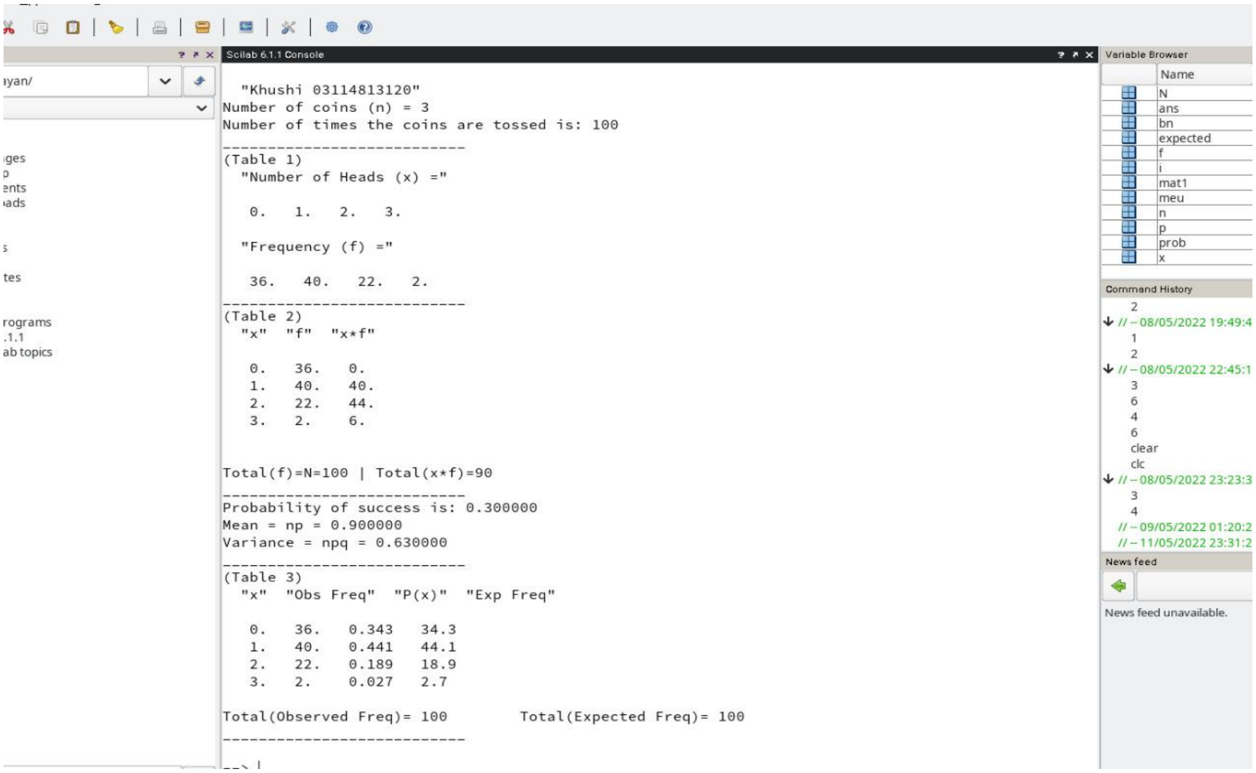
EXPERIMENT - 5

AIM : Fitting of binomial distributions after computing mean and variance.

Code:

```
clc
disp('khushi (03114813120)')
n=3;
x=[0 1 2 3];
f=[36 40 22 2];
a='hello';
mat1=x.*f
meu=sum(mat1)/sum(f)
p=meu/n;
N=100;
mprintf('Number of coins (n) = %i \n',n)
mprintf('Number of times the coins are tossed is: %i \n',N)
mprintf('       \n(Table 1)')

disp('Number of Heads (x) =', x, 'Frequency (f) =',f)

mprintf('             \n(Table 2)')
disp(['x' 'f' 'x*f'],[x' f' mat1'],)

mprintf('Total(f)=N=%i | Total(x*f)=%i\n',sum(f),sum(mat1))

mprintf('             \n')

mprintf('Probability of success is: %f \n',p)
mprintf('Mean = np = %f \n',n*p)
mprintf('Variance = npq = %f \n',n*p*(1-p))
mprintf('         \n(Table 3)')
for i=0:3
ans= nchoosek(n,i)*(p^i)*((1-p)^(n-i));
prob(1,i+1)=ans;
expected(1,i+1)=ans*N;
end
disp(['x' 'Obs Freq' 'P(x)' 'Exp Freq'],[x'  prob' expected'])
mprintf('\nTotal(Observed Freq)= %i \t Total(Expected Freq)= %.0f',sum(f),sum(expected))
disp('Khushi (03114813120)')
plot(x,prob)
xtitle('Binomial Fitting Plot by Khushi (03114813120)','Number of successes','P(X)')
```

OUTPUT :



```
  "Khushi 03114813120"
Number of coins (n) = 3
Number of times the coins are tossed is: 100
---------------------------
(Table 1)
  "Number of Heads (x) ="

   0.   1.   2.   3.

  "Frequency (f) ="

   36.   40.   22.   2.
---------------------------
(Table 2)
  "x"  "f"  "x*f"

   0.   36.   0.
   1.   40.   40.
   2.   22.   44.
   3.   2.    6.


Total(f)=N=100 | Total(x*f)=90
---------------------------
Probability of success is: 0.300000
Mean = np = 0.900000
Variance = npq = 0.630000
---------------------------
(Table 3)
  "x"  "Obs Freq"  "P(x)"  "Exp Freq"

   0.   36.   0.343   34.3
   1.   40.   0.441   44.1
   2.   22.   0.189   18.9
   3.   2.    0.027   2.7

Total(Observed Freq)= 100        Total(Expected Freq)= 100
---------------------------

-->
```

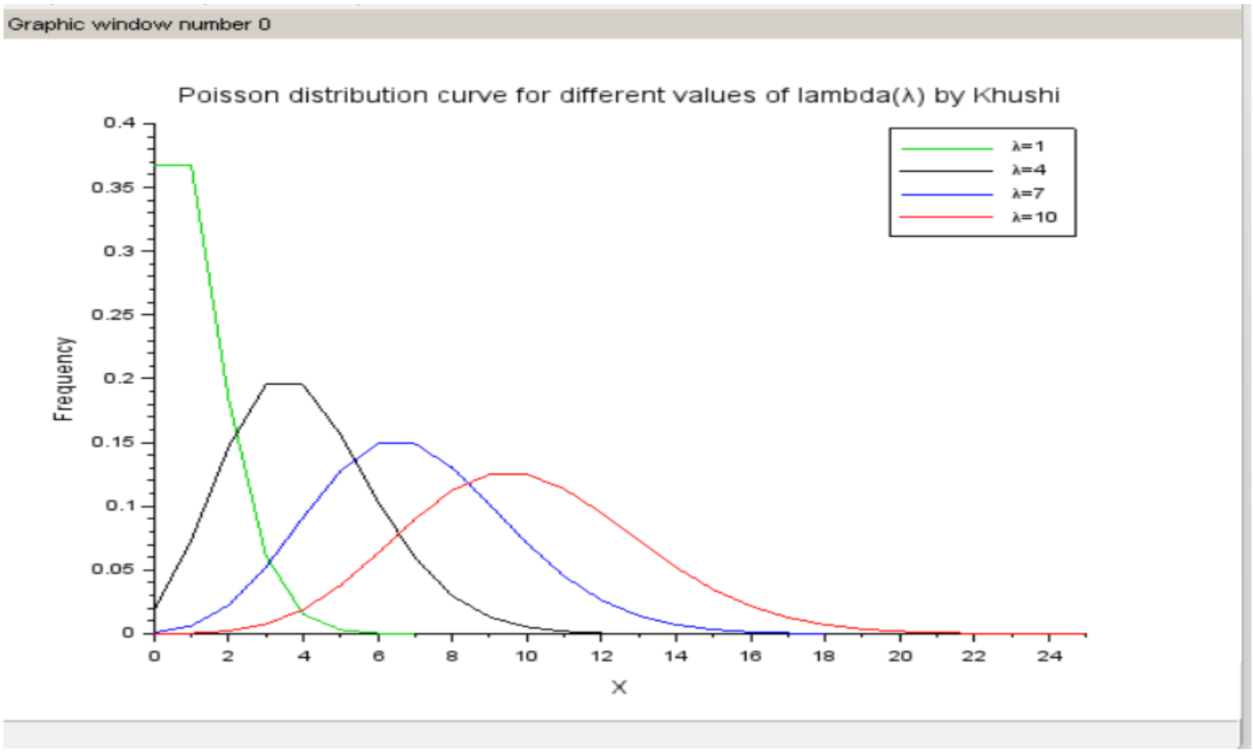
EXPERIMENT - 6

AIM : : Fitting of Poisson distributions for different values of lambda (using grand function).

Code:

```
clear
clc
lambda= [1 4 7 10]
s=[15 1 2 5]
scf();
for i=1:4
    mtlb_hold('on')
    x = grand(10000,1,"poi",lambda(i))
    xmax=max(x)
    x = linspace (0,max(xmax),xmax+1)
    plot2d (x, exp(-lambda(i)).*((lambda(i))^x)./factorial(x), style=s(i))
end
xtitle("Poisson distribution curve for different values of lambda(λ) by Khushi","X","Frequency");
legend(['λ=1' 'λ=4' 'λ=7' 'λ=10'])
```

OUTPUT :



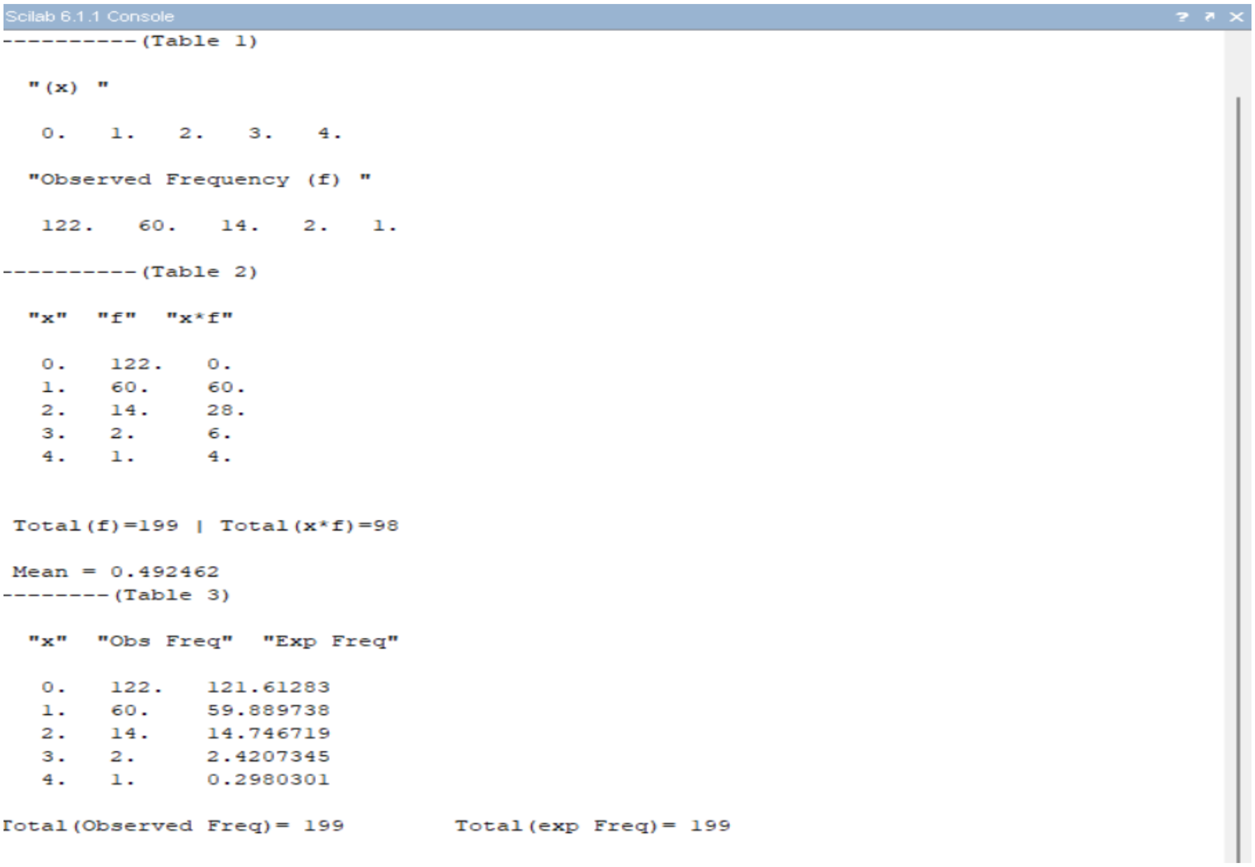
EXPERIMENT - 7

AIM : Fitting of Poisson distributions after computing mean.

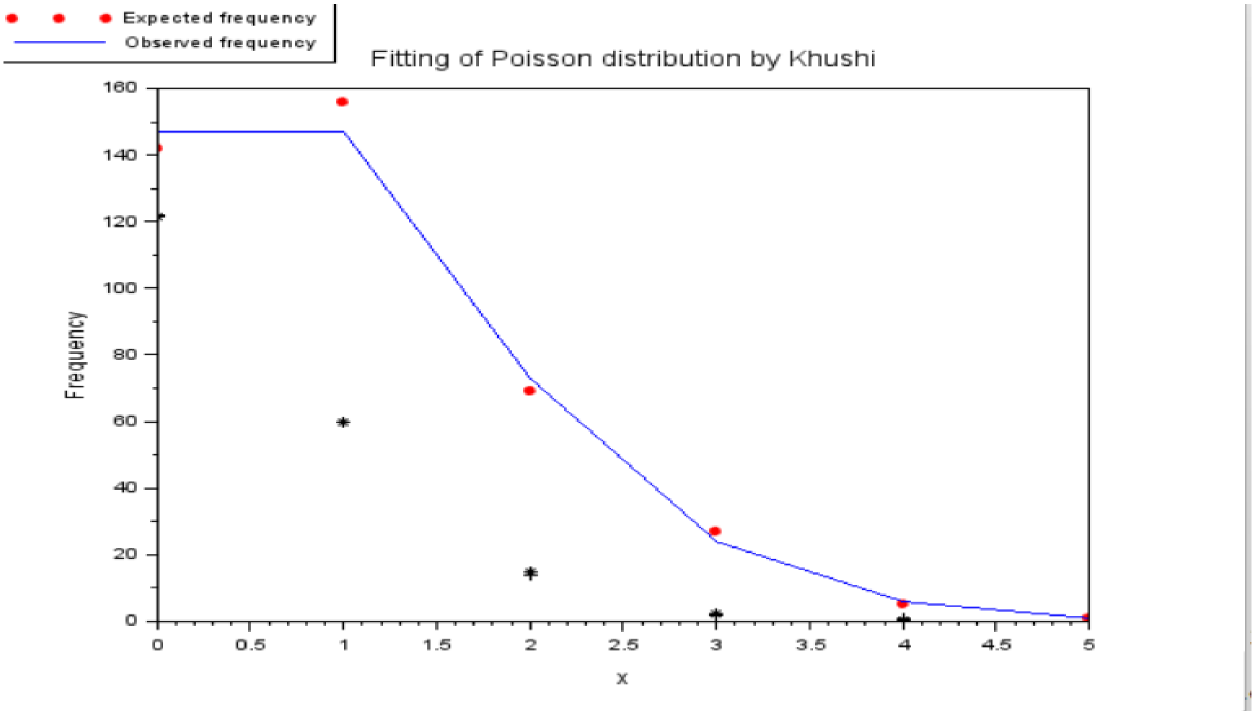
Code:

```
clc;
disp('Khushi (03114813120)')
 x=[0,1,2,3,4];
f=[122,60,14,2,1];
m=x.*f;
u=sum(m)/sum(f);
N =sum(f)
mprintf('----------(Table 1)        \n')
disp('(x) ', x, 'Observed Frequency (f) ',f);
mprintf(' \n----------(Table 2)    \n')
disp(['x' 'f' 'x*f'],[x' f' m'],)
mprintf(' Total(f)=%i | Total(x*f)=%i\n \n',sum(f),sum(m))
mprintf(' Mean = %f \n',u)
mprintf('--------(Table 3)          \n')
for i= 0:4
thf =(N*exp(-u).*(u.^i)./gamma(i+1));
thfr(i+1)=thf;
end
disp(['x' 'Obs Freq' 'Exp Freq'],[x' f' thfr])
mprintf('\nTotal(Observed Freq)= %i \t Total(exp Freq)= %.0f',sum(f),sum(thfr))
mprintf('\n       \n')
plot(x,thfr,"* black");
plot(x,f,"+ black");
xlabel('x');
ylabel('Frequency');
title("Fitting of Poisson distribution by Khushi");

legend (['Expected frequency' ' Observed frequency'],5)
```

OUTPUT :



```
-----------(Table 1)

  "(x)  "

   0.    1.    2.    3.    4.

  "Observed Frequency (f) "

   122.    60.    14.    2.    1.

-----------(Table 2)

  "x"   "f"   "x*f"

   0.    122.    0.
   1.    60.     60.
   2.    14.     28.
   3.    2.      6.
   4.    1.      4.


 Total(f)=199 | Total(x*f)=98

 Mean = 0.492462
---------(Table 3)

  "x"   "Obs Freq"   "Exp Freq"

   0.    122.    121.61283
   1.    60.     59.889738
   2.    14.     14.746719
   3.    2.      2.4207345
   4.    1.      0.2980301

Total(Observed Freq)= 199        Total(exp Freq)= 199
```



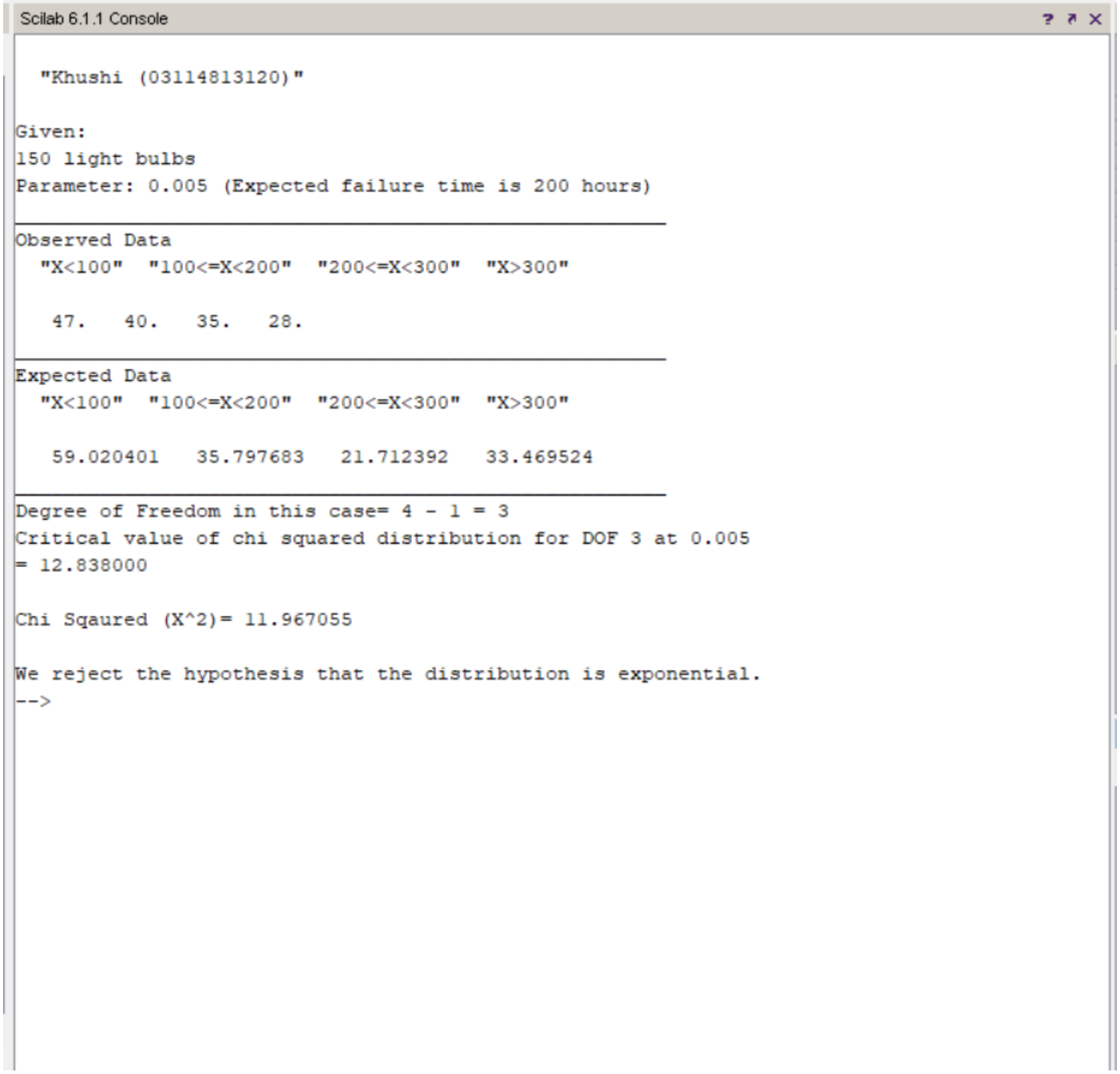
EXPERIMENT - 8

AIM : Testing Goodwill of Fit using a test based on Chi Square Distribution.

Code:

```
clear
clc
format("v",10)
disp('Khushi (03114813120)')
x=[100 200 300 0]
obs=[47 40 35 28]
mprintf('\nGiven: \n150 light bulbs \n')
mprintf('Parameter: 0.005 (Expected failure time is 200 hours)\n')
mprintf('________________________________________________________________\n')
mprintf('Observed Data')
disp(['X<100' '100<=X<200' '200<=X<300' 'X>300'],obs)
prev=0
for i=1:length(x)
if i~=length(x)
p(1,i)=1-exp(-0.005*x(i))-prev
k=1-exp(-0.005*x(i))
prev=k
else
p(i)=exp(-0.005*x(i-1))
end
end
expected=150*p
chi2=sum(((expected-obs)^2)./expected)
mprintf('________________________________________________________________\n')
mprintf('Expected Data')
disp(['X<100' '100<=X<200' '200<=X<300' 'X>300'], expected)
mprintf('________________________________________________________________\n')
mprintf('Degree of Freedom in this case= %i - 1 = %i', length(obs), length(obs)-1)
critical=12.838
mprintf('\nCritical value of chi squared distribution for DOF %i at 0.005\n= %f', length(obs)-1,
critical)
mprintf('\n\nChi Sqaured (X^2)= %f',chi2)
if chi2<critical then
mprintf('\n\nWe reject the hypothesis that the distribution is exponential.')
else
mprintf('\n\nWe accept the hypothesis that the distribution is exponential.')
end
```

OUTPUT:



```
  "Khushi (03114813120)"

Given:
150 light bulbs
Parameter: 0.005 (Expected failure time is 200 hours)
______________________________________________________
Observed Data
  "X<100"  "100<=X<200"  "200<=X<300"  "X>300"

   47.    40.    35.    28.
______________________________________________________
Expected Data
  "X<100"  "100<=X<200"  "200<=X<300"  "X>300"

   59.020401   35.797683   21.712392   33.469524
______________________________________________________
Degree of Freedom in this case= 4 - 1 = 3
Critical value of chi squared distribution for DOF 3 at 0.005
= 12.838000

Chi Sqaured (X^2)= 11.967055

We reject the hypothesis that the distribution is exponential.
-->
```

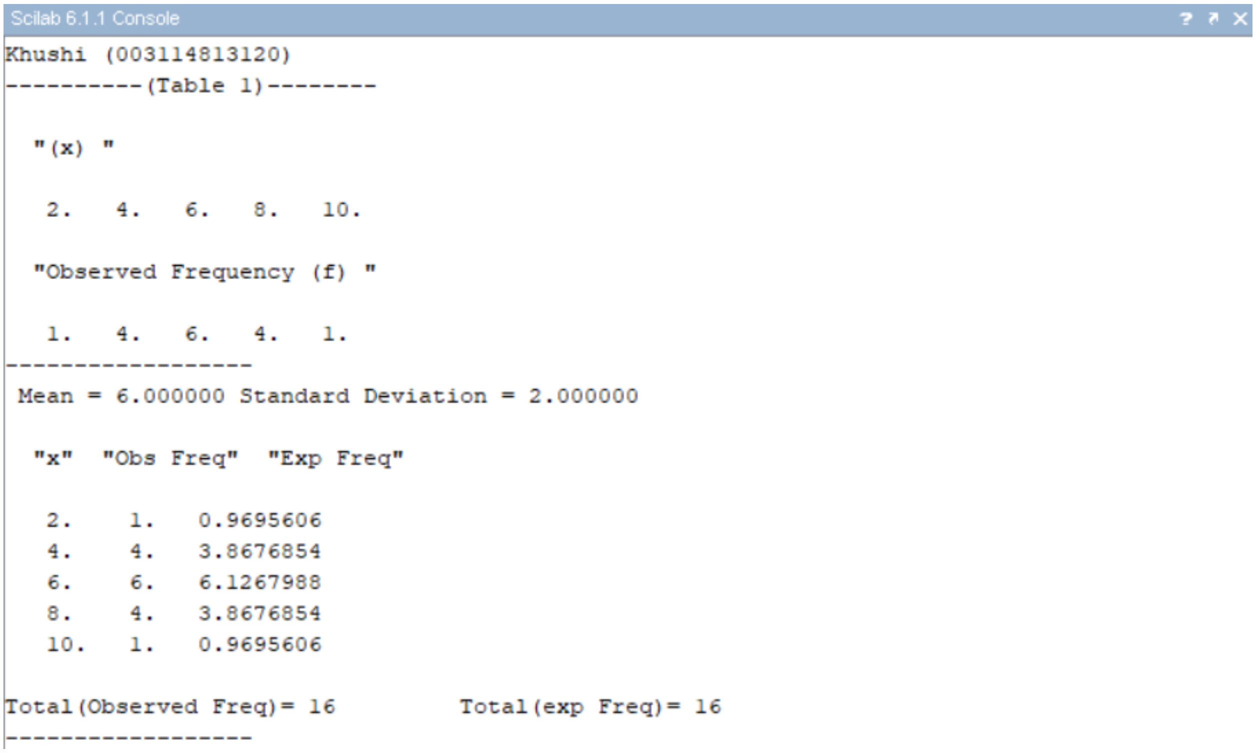
EXPERIMENT - 8

AIM : Fitting of normal distribution when parameters are not given

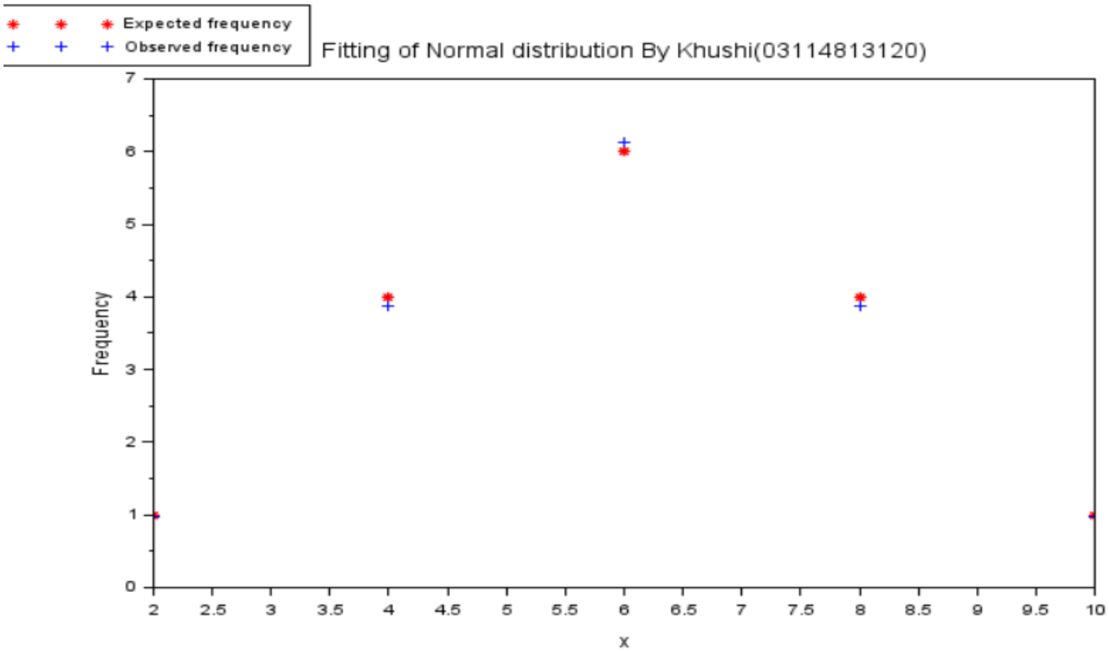
Code:

```
clear
clc
x=[2,4,6,8,10];
f=[1,4,6,4,1];
m=x.*f;
u=sum(m)/sum(f);
N =sum(f)
mprintf('Khushi (003114813120)\n')
mprintf('----------(Table 1)--------\n')
disp('(x) ', x, 'Observed Frequency (f) ',f);
mprintf('------------------\n')
// to find standard deviation
a= f.*((x-u)^2)
b= sum(a);
sd= sqrt(b/(N));
mprintf(' Mean = %f Standard Deviation = %f \n',u,sd);
x1 =x-1;
x2 =x+1;
for i=1:5
 z=[0];
z=(cdfnor("PQ",x2(i),u,sd)- cdfnor("PQ",x1(i),u,sd));
Pz(1)=z;
Exf(i)=N*z;
end
disp(['x' 'Obs Freq' 'Exp Freq'],[x' f' Exf])
mprintf('\nTotal(Observed Freq)= %i \t Total(exp Freq)= %.0f',sum(f),sum(Exf))
mprintf('\n------------------\n')
plot(x,f,"*r");
plot(x,Exf,"+ b");
xlabel('x');
ylabel('Frequency');
title("Normal distribution");
legend (['Expected frequency' ' Observed frequency'],5)
```

OUTPUT:



```
Khushi (003114813120)
----------(Table 1)--------

  "(x) "

   2.   4.   6.   8.   10.

  "Observed Frequency (f) "

   1.   4.   6.   4.   1.
------------------
 Mean = 6.000000 Standard Deviation = 2.000000

  "x"  "Obs Freq"  "Exp Freq"

   2.    1.    0.9695606
   4.    4.    3.8676854
   6.    6.    6.1267988
   8.    4.    3.8676854
   10.   1.    0.9695606

Total(Observed Freq)= 16         Total(exp Freq)= 16
------------------
```


✱ ✱ ✱ Expected frequency
+ + + Observed frequency
Fitting of Normal distribution By Khushi(03114813120)
Frequency
x