CSC446 ASS3

YINGCHAOYU V00830514

1. Named random.java

Output is:

```
E:\csc446>java random
X0 is 5.52
the null hypothesis is not rejected since X0<16.9

E:\csc446>java random
X0 is 13.42
the null hypothesis is not rejected since X0<16.9

E:\csc446>java random
X0 is 9.02
the null hypothesis is not rejected since X0<16.9

E:\csc446>java random
X0 is 11.959999999999999
the null hypothesis is not rejected since X0<16.9
```

2.

A. Kolmogorv method:

Named q2.java

```
E:\csc446>java q2
Random numbers in java:
0.048207132492562144
0.7349731062737864
0.823012118762264
0.3978771627093538
0.5732833506127893
0.8769321430099684
0.6558575862542575
0.089503583333761399
0.6178006265337461
0.3545755421217991


D= 0.17328335061278932
H0 is not rejected since d<D(0.05,10)=0.410
```

B.Chi-Square test:

Named q2b.java

Assume a=0.05 , the test case n=10;

```
E:\csc446>java q2b
After generating 1000 random numbers in java:
X0 is 14.179999999999998
the null hypothesis is not rejected since X0<X(0.05,9)=16.9

E:\csc446>java q2b
After generating 1000 random numbers in java:
X0 is 15.280000000000001
the null hypothesis is not rejected since X0<X(0.05,9)=16.9

E:\csc446>java q2b
After generating 1000 random numbers in java:
X0 is 4.159999999999999
the null hypothesis is not rejected since X0<X(0.05,9)=16.9
```

3.

	A	B	C	D	E	F
1	i	0	1	2	3	4
2	X0	7	13	15	5	7
3	X1	8	8	8	8	8
4	X2	7	1	7	1	7
5	X3	8	8	8	8	8

Since c=0; Max period=16/4=4;
So we conclude that if X0 is odd and a=3+8k, then the maximum period is achieved, otherwise the period is less than maximum period.

4.
CDF:
F(x)=e^(2x)/2 x<0
= 1-e^(-2x)/2 x>0
Let F(x)=R ,we get:

X=ln(2R)/2 0<R=<1/2
= -ln(2-2R)/2 1/2 <R <1

5. We get i=1; m=3; N=20; M=5;
Define R(X,Y)=RX*RY;
P13=1/6(R(1,4)+R(4,7)+R(7,10)+R(10,13)+R(13,16)+R(16,19))-0.25
=1.216918/6-0.25

=-0.04718

Ơp13=0.11785

Z0= P13/Ơp13=-0.4

Since z0.025=1.96 and Z0 is between -z0.025 and z0.025, the hypothesis is not rejected;

6.

Named q6.c

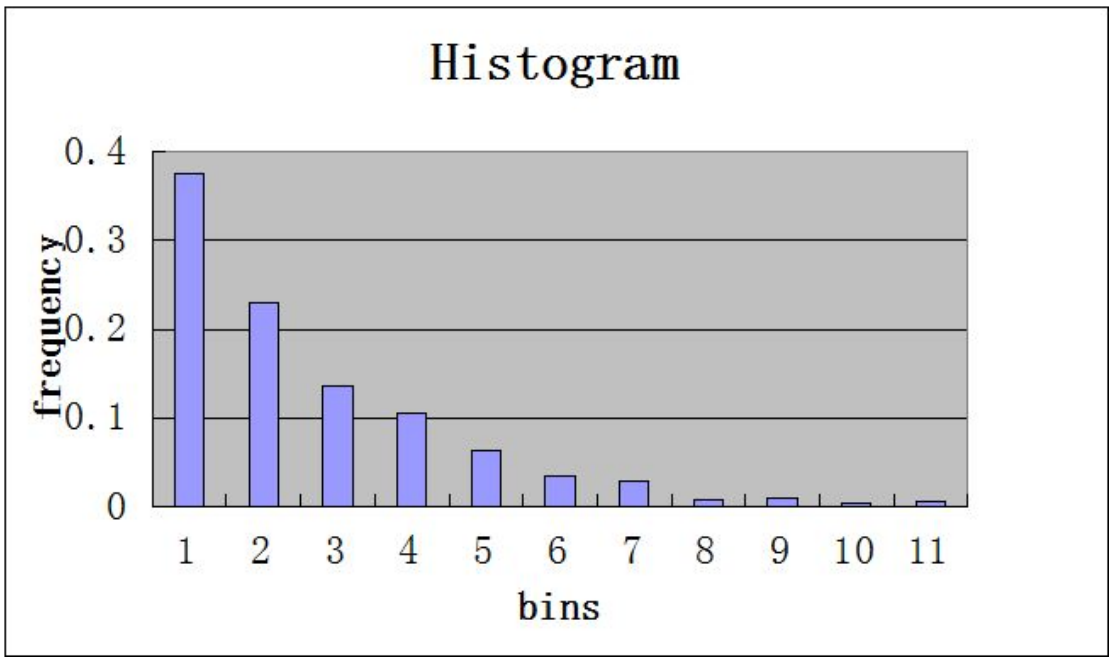
7.

Named q7.c (worked in some software):

```
int main()
{
int n=3;
int k=0;
int i;
int j;
double x[1000],r[1000];
int g[11];
srand((unsigned)time(NULL));
    for(i=0;i<1000;i++){
r[i]=rand()/(float)(RAND_MAX);
    //printf("%lf ",r[i]);
x[i]=-log(1-r[i]);
// printf("X[%d] is %lf.  ",i,x[i]);
    }
    for(j=0;j<11;j++){
        for(i=0;i<1000;i++){
            if(j==10){
                if(2*x[i]>=j){
                    g[j]++;
                }
            }
            else{
            if(2*x[i]>=j &&2*x[i]<j+1 ){
                g[j]++;
            }
            }
        }
        printf("%d ",g[j]);
    }
    return 0;
}
```

```
376 229 136 106 64 35 29 7 9 4 5
```

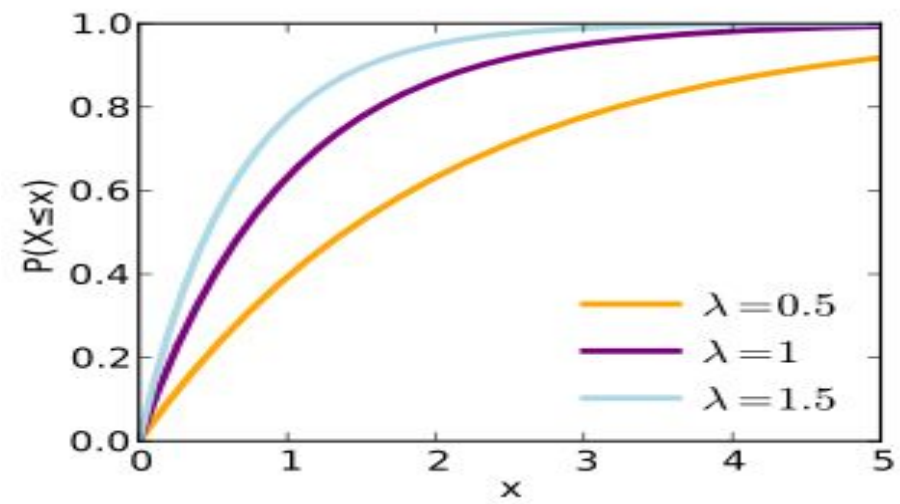
The data are: 376 229 136 106 64 35 29 7 9 4 5 in 11bins.



The mid point of bins are slowly decreasing from 1-11 bins.

Actual CDF:

exponential CDF:



Conclusion: They are similar in the CDF.

8.

Procedures:

Step1: Generate R~U[0,1]

Step2: If R>=p, accept X=R

Step3: If R<p, reject R, return to Step1