Practical-3

AIM: Implement a program to perform Kolmogorov Smirnov (KS) test.

THEORY:

Kolmogorov–Smirnov test a very efficient way to determine if two samples are significantly different from each other. It is usually used to check the uniformity of random numbers. Uniformity is one of the most important properties of any random number generator and Kolmogorov–Smirnov test can be used to test it.

The Kolmogorov–Smirnov test may also be used to test whether two underlying one-dimensional probability distributions differ. It is a very efficient way to determine if two samples are significantly different from each other.

The Kolmogorov–Smirnov statistic quantifies a distance between the empirical distribution function of the sample and the cumulative distribution function of the reference distribution, or between the empirical distribution functions of two samples.

To use the test for checking the uniformity of random numbers, we use the CDF (cumulative distribution function) of U[0, 1].

F(x)= x for 0<=x<=1

Empirical CDF, Sn(x)= (number of R1, R2…Rn < x)/N array of random numbers, the random numbers must be in the range of [0, 1].

PROGRAM:

```
#include<bits/stdc++.h>
using namespace std;
class KS
{
        private:
                float numbers[20];
                float D,tabulatedD;
                float Dplusmax,Dminusmax;
                float Dplus[20],Dminus[20];
                float ratio[20],ratiominus[20];
                int i,j,n;



        public:
                void getdata() //to get the random numbers
                {
                        cout<<"How many numbers?:"<<endl;
                        cin>>n;
                        cout<<"Enter "<<n<<" numbers"<<endl;
                        for(i=0;i<n;i++)
```

```
                    {
                            cout<<"Enter "<<i+1<<" number:"<<endl;
                            cin>>numbers[i];
                    }
            }
            void BubbleSort() // arrange the number in increasing order
            {
                    int i,j;
                    float temp;
                    for(i=0;i<n-1;i++)
                    {
                            for(j=0;j<n-i-1;j++)
                            {
                                    if(numbers[j]>numbers[j+1])
                                    {
                                            temp=numbers[j];
                                            numbers[j]=numbers[j+1];
                                            numbers[j+1]=temp;
                                    }
                            }
                    }
                    cout<<"The numbers in ascending order is:"<<endl;
                    for(i=0;i<n;i++)
                    {
                            cout<<setprecision(2)<<numbers[i]<<" ";
                    }

            }
            void calculate() // find D+, D-
            {
                    for(i=0;i<n;i++)
                    {
                            int j;
                            j=i+1;
                            ratio[i]=(float)j/n;
                            ratiominus[i]=(float)i/n;
                            Dplus[i]=ratio[i]-numbers[i];
                            Dminus[i]=numbers[i]-ratiominus[i];


                    }
            }
            void display() // display the tabulated format and find D
            {
                    cout<<endl;
```

```cpp
cout<<endl;
cout<<setw(10)<<"i";
for(i=1;i<=n;i++)
{
        cout<<setw(10)<<i;

}
cout<<endl;
cout<<setw(10)<<"R(i)";
for(i=0;i<n;i++)
{
        cout<<setw(10)<<numbers[i];
}
cout<<endl;
cout<<setw(10)<<"i/n";

for(i=0;i<n;i++)
{
        cout<<setw(10)<<setprecision(2)<<ratio[i];
}
cout<<endl;
cout<<setw(10)<<"D+";
for(i=0;i<n;i++)
{
        cout<<setw(10)<<setprecision(2)<<Dplus[i];
}
cout<<endl;
cout<<setw(10)<<"D-";
for(i=0;i<n;i++)
{
        cout<<setw(10)<<setprecision(2)<<Dminus[i];
}
cout<<endl;
Dplusmax=Dplus[0];
Dminusmax=Dminus[0];
for(i=1;i<n;i++)
{

        if(Dplus[i]>Dplusmax)
        {
                Dplusmax=Dplus[i];
        }
        if(Dminus[i]>Dminusmax)
        {
```

```
                              Dminusmax=Dminus[i];
                    }
          }
          cout<<"D+ max: "<<Dplusmax<<endl;
          cout<<"D- max: "<<Dminusmax<<endl;
          cout<<"D =max("<<Dplusmax<<", "<<Dminusmax<<") =";
          if(Dplusmax>Dminusmax)
          {
                    D=Dplusmax;
          }
          else
          {
                    D=Dminusmax;
          }
          cout<<D;
          cout<<endl;

     }
void conclusion() // asking tabulated D and comparing it with D(calculated)
{
          cout<<"Enter the tabulated value:"<<endl;
          cin>>tabulatedD;

          if(D<tabulatedD)
          {
                    cout<<"The test is accepted."<<endl;
          }
          else
          {
                    cout<<"The test is rejected."<<endl;
          }
}
};


int main() //main function
{
     KS ks1; //object of KS class
     ks1.getdata(); //function calls
     ks1.BubbleSort();
     ks1.calculate();
     ks1.display();
     ks1.conclusion();
     return(0);
```

}

OUTPUT:

```
    "C:\Users\Ankit Goyal\OneDrive\Documents\labs\8th Sem Lab\SSM\Kolmogorov-Smirnov(KS-Test).exe"
How many numbers?:
5
Enter 5 numbers
Enter 1 number:
0.12
Enter 2 number:
0.25
Enter 3 number:
0.90
Enter 4 number:
0.65
Enter 5 number:
0.44
The numbers in ascending order is:
0.12 0.25 0.44 0.65 0.9

            i          1          2          3          4          5
         R(i)        0.12       0.25       0.44       0.65       0.9
          i/n         0.2        0.4        0.6        0.8         1
           D+        0.08       0.15       0.16       0.15       0.1
           D-        0.12       0.05       0.04       0.05       0.1
D+ max: 0.16
D- max: 0.12
D =max(0.16, 0.12) =0.16
Enter the tabulated value:
0.563
The test is accepted.

Process returned 0 (0x0)   execution time : 211.950 s
Press any key to continue.
```