Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования «Рязанский государственный радиотехнический университет имени В. Ф. Уткина»

Кафедра ВПМ

Отчёт по лабораторной работе №1

«Изучение базовых генераторов псевдослучайных чисел»

Выполнил:

ст. гр. 9413

Романов А. Р.

Проверил:

Проф. каф. ВПМ

Овечкин Г. В.

Рязань 2023

Задание: составить и отладить программу (подпрограмму)

генерирования псевдослучайных чисел с равновероятным

распределением на интервале [0;1). Вариант задания выбирается из

таблицы 1, в которой указаны тип генератора случайных чисел,

начальные условия и пр. Для заданных объема выборки и числа

участков разбиения интервала [0;1) построить гистограмму частот и

статистическую функцию распределения, получить программным

способом оценки математического ожидания, дисперсии, второго и

третьего моментов.





Код программы

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace lab1

{

public partial class Form1 : Form

{

private int \_Mm = 4096 \* 4;

private int \_Y0 = 3971;

private int \_Y1 = 1013;

int V = 2000;

int N = 21;

double MX = 0;

double DX = 0;

double m2 = 0;

double m3 = 0;

public Form1()

{

InitializeComponent();

}

private void button1\_Click(object sender, EventArgs e)

{

double[] x = values();

printDiagramm(x);

printStaticFunc(x);

MX = MatOzh(x);

DX = Disp(x,MX);

m2 = Moments(x, 2);

m3 = Moments(x, 3);

label1.Text = "Мат. ожидание = " + Convert.ToString(MX);

label2.Text = "Дисперсия = " + Convert.ToString(DX);

label3.Text = "Второй момент = " + Convert.ToString(m2);

label4.Text = "Третий момент = " + Convert.ToString(m3);

}

public double MatOzh(double[] x)

{

double M = 0;

for (int i = 0;i < x.Length; i++)

{

M += x[i];

}

return M/x.Length;

}

public double Disp(double[] x, double Mx)

{

double D = 0;

for (int i = 0; i < x.Length; i++)

{

D += x[i] \* x[i];

}

D /= x.Length;

return (D - Mx \* Mx) \* x.Length / (x.Length - 1);

}

public double Moments(double[] x, int degree)

{

double sum = 0;

for (int i = 0; i < x.Length; i++)

{

sum += Math.Pow(x[i], degree);

}

return sum / x.Length;

}

public double[] values()

{

double[] x = new double[V];

for (int i = 0; i < V; i++)

{

x[i] = Rnd();

}

return x;

}

public double Rnd()

{

int у = (\_Y0 + \_Y1) % \_Mm;

\_Y0 = \_Y1;

\_Y1 = у;

return (double)у / \_Mm;

}

public int countLessValueFromRnds(double[] x, double value)

{

int count = 0;

for (int i = 0; i < x.Length - 1; i++)

{

if (x[i] < value)

{

count++;

}

}

return count;

}

public void printDiagramm(double[] x)

{

chart1.ChartAreas[0].AxisX.Minimum = 0;

chart1.ChartAreas[0].AxisX.Maximum = 1;

chart1.ChartAreas[0].AxisY.Minimum = 0;

double[] xData = new double[V];

double[] yData = new double[V];

for (int i = 0; i < N; i++)

{

double count = 0;

for (int j = 0; j < x.Length; j++)

{

if (x[j] >= i \* ((double)1 / N) && x[j] < (i + 1) \* ((double)1 / N))

{

count += 1;

}

}

xData[i] = i \* ((double)1 / N);

yData[i] = count / (V \* ((double)1 / N));

}

chart1.Series[0].Points.DataBindXY(xData, yData);

chart1.Series[0].Name = "Гистограмма";

}

public void printStaticFunc(double[] x)

{

chart2.ChartAreas[0].AxisX.Minimum = 0;

chart2.ChartAreas[0].AxisX.Maximum = 1;

chart2.ChartAreas[0].AxisY.Minimum = 0;

chart2.ChartAreas[0].AxisY.Maximum = 1;

int count = 1000;

double[] xs = new double[1000];

double[] ys = new double[1000];

double s = 0;

for (int i = 0; i < 1000; i++)

{

double y = (double)countLessValueFromRnds(x, s) / V;

xs[i] = s;

ys[i] = y;

s += 0.001;

}

chart2.Series[0].Points.DataBindXY(xs, ys);

chart2.Series[0].Name = "Статическая функция";

}

}

}

Результат выполнения программы:

