Учреждение образования

Белорусский государственный университет информатики

и радиоэлектроники

Факультет компьютерных технологий ИИТ

Лабораторная работа №3

Моделирование

Разработка, отладка и исследование  
 программной модели дискретно-стохастической СМО

Выполнил: студенты гр.280512

Богданов Н.В.

Пилипчик А.О.

Проверил: преподаватель Лашкевич Е.М.

Минск, 2015

**Задание**

Для СМО заданной конфигурации построить имитационную модель и исследовать ее (разработать алгоритм и написать имитирующую программу, предусматривающую сбор и статистическую обработку данных для получения оценок заданных характеристик СМО). Распределение интервалов времени между заявками во входном потоке и интервалов времени обслуживания – геометрическое с соответствующим параметром (ρ, π1, π2). Если ρ не задано, то входной поток – регулярный (с указанным в обозначении источника числом тактов между заявками).

Lоч – средняя длина очереди;

1) Исходные данные

Р-схема

π

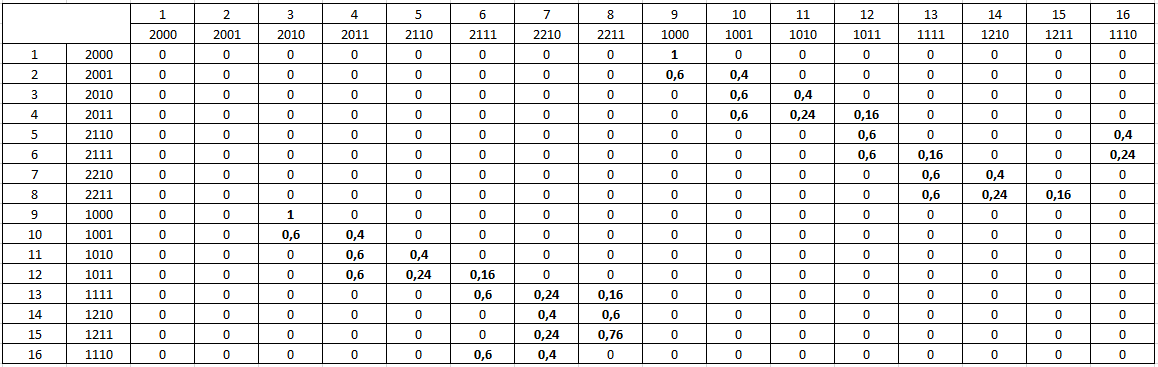
2

|  |  |  |  |
| --- | --- | --- | --- |
| *ρ* | *π*1 | *π*2 | Цель исследования |
| - | 0,4 | 0,4 | Lоч |

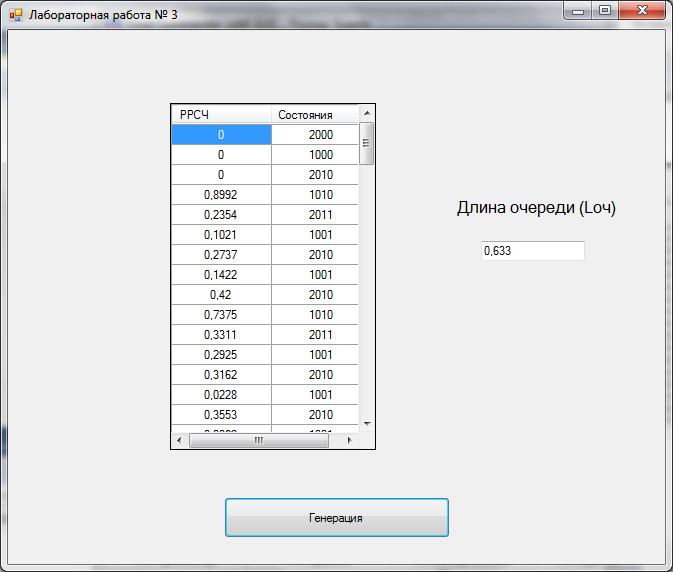
π

2

2)Таблица переходов



3)Скиншот программы



4)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 Lab3

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

dataGridView1.RowCount = 100;

dataGridView1.Rows[0].Cells[0].Value = 0;

dataGridView1.Rows[1].Cells[0].Value = 0;

dataGridView1.Rows[2].Cells[0].Value = 0;

dataGridView1.Rows[0].Cells[1].Value = 2000;

dataGridView1.Rows[1].Cells[1].Value = 1000;

dataGridView1.Rows[2].Cells[1].Value = 2010;

}

Random rand = new Random();

string sost = "2010";

double gen = 0;

int row1 = 2;

int row2 = 2;

double dlina = 0;

double[] sost2010 = { 0, 0.6, 1 };

double[] sost2011 = { 0, 0.6, 0.84, 1 };

double[] sost2110 = { 0, 0.6, 1 };

double[] sost2111 = { 0, 0.6, 0.76, 1 };

double[] sost2210 = { 0, 0.6, 1 };

double[] sost2211 = { 0, 0.6, 0.84, 1 };

double[] sost1001 = { 0, 0.6, 1 };

double[] sost1010 = { 0, 0.6, 1 };

double[] sost1011 = { 0, 0.6, 0.84, 1 };

double[] sost1111 = { 0, 0.6, 0.84, 1 };

double[] sost1210 = { 0, 0.4, 1 };

double[] sost1211 = { 0, 0.24, 1 };

double[] sost1110 = { 0, 0.6, 1 };

string Perehod(double r, string sost)

{

if (sost == "1000")

{

return "2010";

}

if (sost == "2010")

{

if (r > sost2010[0] && r < sost2010[1])

return "1001";

else return "1010";

}

if (sost == "2011")

{

if (r > sost2011[0] && r < sost2011[1])

return "1001";

if (r >= sost2011[1] && r < sost2011[2])

return "1010";

else return "1011";

}

if (sost == "2110")

{

dlina += 0.059;

if (r > sost2110[0] && r < sost2110[1])

return "1011";

else return "1110";

}

if (sost == "2111")

{

dlina += 0.086;

if (r > sost2111[0] && r < sost2111[1])

return "1011";

if (r >= sost2111[1] && r < sost2111[2])

return "1111";

else return "1110";

}

if (sost == "2210")

{

dlina += 0.088;

if (r > sost2210[0] && r < sost2210[1])

return "1111";

else return "1210";

}

if (sost == "2211")

{

dlina += 0.03;

if (r > sost2211[0] && r < sost2211[1])

return "1111";

if (r >= sost2211[1] && r < sost2211[2])

return "1210";

else return "1211";

}

if (sost == "1001")

{

if (r > sost1001[0] && r < sost1001[1])

return "2010";

else return "2110";

}

if (sost == "1010")

{

if (r > sost1010[0] && r < sost1010[1])

return "2011";

else return "2110";

}

if (sost == "1011")

{

if (r > sost1011[0] && r < sost1011[1])

return "2011";

if (r >= sost1011[1] && r < sost1011[2])

return "2110";

else return "2111";

}

if (sost == "1111")

{

dlina += 0.046;

if (r > sost1111[0] && r < sost1111[1])

return "2111";

if (r >= sost1111[1] && r < sost1111[2])

return "2210";

else return "2211";

}

if (sost == "1210")

{

dlina += 0.038;

if (r > sost1210[0] && r < sost1210[1])

return "2210";

else return "2211";

}

if (sost == "1211")

{

dlina += 0.0048;

if (r > sost1211[0] && r < sost1211[1])

return "2210";

else return "2211";

}

if (sost == "1110")

{

dlina += 0.07;

if (r > sost1110[0] && r < sost1110[1])

return "2111";

else return "2210";

}

return "error";

}

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

{

gen = rand.NextDouble();

row1++;

row2++;

dataGridView1.Rows[row1].Cells[0].Value = Math.Round(gen, 4);

sost = Perehod(gen, sost);

dataGridView1.Rows[row2].Cells[1].Value = sost;

textBox1.Text = dlina.ToString();

}

}

}