МИНОБРНАУКИ РОССИИ

федеральное государственное бюджетное образовательное учреждение

высшего образования

«Череповецкий государственный университет»

**Лабораторная работа № 5**

**Построение генераторов случайных чисел с заданными функциями распределения для моделирования процессов различной природы**

**Выполнил:**

студент гр. 1ИВТпб-01-31оп

Климов А.Г.  
**Проверил:** преподаватель

Юдина О.В.  
Отметка о зачете:

Череповец

2018 год

***Построение генераторов случайных чисел с заданными функциями распределения для моделирования процессов различной природы***

***Цель работы:*** изучить алгоритмы построения генераторов базовой последовательности, а также генераторов с заданными функциями распределения.

***1. Задание:***

1.1. Разработать генератор случайных чисел как подпрограмму вычисления 10 случайных чисел в интервале [0;1] мультипликативным конгруэнтным методом RANDOM1. (Данную подпрограмму используйте для разработки программ в п.1.2 и 1.3.)

1.2. Разработать программу реализации алгоритма, формирующего последовательность из 10 случайных величин с равномерным распределением на отрезке [a; b]. Границы отрезка a и b задаются по варианту. Результат представить в виде 10-и значений Yi.

1.3. Разработать программу реализации алгоритма, формирующего последовательность из 10 случайных величин с показательным распределением, величина λ задается по варианту. Результат представить в виде 10-и значений Ti.

***2. Порядок выполнения работы:***

2.1.Конгруэнтные методы используют фундаментальное соотношение конгруэнтности, выражаемое формулой (1):

ni+1 = (λ ni +μ)(mod m) (1)

где ni – i-й элемент последовательности, λ , μ, m – неотрицательные целые числа.

Выражение z = y(mod m) означает, что z есть остаток деления (y/m).

*Мультипликативный конгруэнтный алгоритм* задает последовательность неотрицательных целых чисел {ni}, не превосходящих m, по формуле:

ni+1 = λ ni (mod m) (2)

Это частный случай формулы (1) при μ=0.

Рассмотрим выбор параметров для десятичной ЭВМ, чтобы полученная последовательность имела максимально возможный период. Формулу (2) рассмотрим в виде:

ni+1 = λ ni (mod 10d) (3)

Для десятичных машин модуль m выбирается 10d, где d – число десятичных цифр в машинном слове. Максимальный период p = 5 \* 10 d-2.

Чтобы построить последовательность с таким периодом, нужно:

1. Выбрать n0 – любое нечетное число и не делящееся на 5,
2. Вычислить λ: λ = 200\*t ± z, где t – любое целое число, z взято из набора:

{3, 11, 13,19,21, 27,29, 37, 53, 59, 61, 67, 69, 77, 83, 91}.

1. Вычислить λ• n0. Взять d младших разрядов полученного числа в качестве n1, остальные отбросить.
2. Поставить десятичную запятую слева от найденного числа ni . Полученную дробь считать за xi, т.е.  ( i = 1, 2,…).
3. Вычислить очередное псевдослучайное число ni+1 как d правых разрядов произведения λ ni и вернуться к п.4.

Пример: Пусть d =4. Мультипликативный генератор определит 5\*10 4-2 = 500 различных чисел.

1. Возьмем n0 = 5379.
2. Для t = 0 и z = 91 можно взять λ = ± 91. Пусть λ = 91.
3. λ n0 = 91 \* 5379 = 00489489. Отсюда n1 = 9489, х1 = 0,9489.
4. λ n1 = 91 \* 9489 = 00863499. Отсюда n2 = 3499, х2 = 0,3499.
5. λ n2 = 91 \* 3499 = 00318409. Отсюда n3 = 8409, х3 = 0,8409 и т.д.

n4 = 5219, n5 = 4929, n6 = 8539…Младшие разряды не напоминают случайные.

Мультипликативная процедура обеспечивает полный период высшему разряду генерируемых чисел. Если эксперимент позволяет ограничиться выборкой, состоящей из чисел, включающих меньше значащих цифр, чем укладывается в машинное слово, надо использовать высшие порядки чисел мультипликативного генератора.

Для построения последовательности в практической работе возьмите два значащих разряда полученных чисел: 0,53; 0,94; 0,34; 0,84 и т.д.

2.2. Имитация случайных величин, заданных различными функциями распределения, может быть выполнена на основе генератора равномерно распределенных случайных чисел в интервале [0;1], построенного конгруэнтным методом в п.6.1.

Равномерное распределение случайной величины Х представляет собой непрерывную функцию плотности вероятности, постоянную внутри интервала от a до b и равную 0 вне этого интервала. Функция распределения: .

Для имитации равномерного распределения на интервале [a; b] используется обратное преобразование функции плотности:

**Xi = a + (b-a) \* ni** , где 0 ≤ ni ≤ 1. (4)

ni задается с помощью подпрограммы вычисления случайных чисел в интервале [0; 1], a и b задаются на входе.

x1 = a + (b-a)\*n1, x2 = a + (b-a)\*n2, …

Получаем последовательность случайной величины xi , равномерно распределенной на интервале [a; b]. Последовательность представьте как значения Yi.

2.3. Показательное распределение. Случайная величина Х имеет показательное распределение с параметром λ. Функция распределения f(x): F(x) = 1 - λ e -λ x, х ≥0, λ≥0.

 или  (5)

где ni задается с помощью подпрограммы вычисления случайных чисел в интервале [0; 1], λ задаётся на входе.

Полученную последовательность из 10-ти чисел представьте как значения величины Ti.

***3. Содержание отчета:***

Отчет должен содержать результаты выполнения п.п. 5.1. - 5.3.

***4. Контрольные вопросы:***

8.1. Какие Вы можете назвать способы генерации случайных чисел?

8.2. Что такое генератор случайных чисел?

8.3. Какие Вы знаете методы для построения генераторов случайных чисел?

8.4. Какие Вы знаете функции распределения?

**ПРИЛОЖЕНИЕ**

Таблица 2.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Вариант** | **1** | **2** | **3** | **4** | **5** | **6** | **7** | **8** | **9** | **10** |
| a | 1 | 1 | 2 | 2 | 3 | 4 | 4 | 5 | 5 | 3 |
| b | 2 | 3 | 3 | 4 | 4 | 6 | 7 | 8 | 7 | 5 |
| λ  (для показательного распределения) | 11 | 13 | 53 | 59 | 91 | 77 | 83 | 27 | 29 | 19 |

**Текст программы**

// ConsoleApplication1.cpp : Defines the entry point for the console application.

//

#include "stdafx.h"

#include <iostream>

void print(double m[])

{

for (int i = 0; i < 10; i++)

{

std::cout << m[i] << std::endl;

}

}

void random1(double m[], int nn, int tt, int zz, int dd)

{

for (int i = 0; i < 10; i++)

{

int l = 200 \* tt + zz; // лямбда

nn = (l \* nn) % (int)pow(10.0, dd);

m[i] = nn / pow(10, dd);

}

}

void random2(double m[], int aa, int bb, double m1[])

{

for (int i = 0; i < 10; i++)

{

m[i] = aa + (bb - aa) \* m1[i];

}

}

void random3(double m[], double ll, double m1[])

{

for (int i = 0; i < 10; i++)

{

m[i] = -log(1 - m1[i]) / ll;

}

}

int main()

{

setlocale(LC\_ALL, "Russian");

double result1[10];

double result2[10];

double result3[10];

/\*Конгруэнтный метод\*/

printf("Конгруэнтный метод\n");

int n = 99; // n0 - любое нечетное число и не делящееся на 5

int t = 1; // любое целое число

int z = 59; // число из набора: {3, 11, 13, 19, 21, 27, 29, 37, 53, 59, 61, 67, 69, 77, 83, 91}

int d = 2; // d младших разрядов полученного числа в качестве n1

random1(result1, n, t, z, d);

print(result1);

/\*Равномерное распределение \*/

printf("\nРавномерное распределение\n");

int a = 3, b = 4; // интервал

random2(result2, a, b, result1);

print(result2);

/\*Показательное распределение\*/

printf("\nПоказательное распределение\n");

double l1 = 91.0; // параметр лямбда

random3(result3, l1, result1);

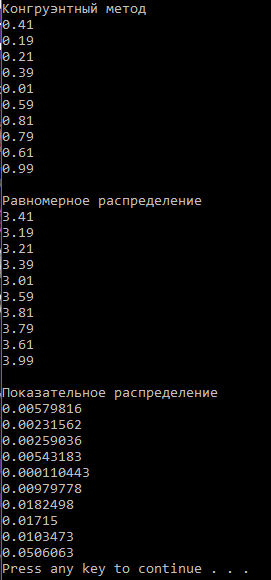
print(result3);

system("pause");

return 0;

}

**Тестирование**



**Ответы на контрольные вопросы**

8.1. Какие Вы можете назвать способы генерации случайных чисел?

1. Линейный конгуэнтный генератор псевдослучайных чисел (ЛКГ ПСЧ)
2. Нелинейные конгуэнтные генераторы псевдослучайных чисел (НКГ ПСЧ)
3. Физические датчики случайных процессов
4. Биологический датчик случайных чисел
5. Физические датчики шума

8.2. Что такое генератор случайных чисел?

Генератор псевдослучайных чисел (ГПСЧ, англ. pseudorandom number generator, PRNG) — алгоритм, порождающий последовательность чисел, элементы которой почти независимы друг от друга и подчиняются заданному распределению (обычно равномерному)

8.3. Какие Вы знаете методы для построения генераторов случайных чисел?

* Аппаратные способы построения генераторов случайных чисел
* Программные способы построения генераторов случайных чисел

8.4. Какие Вы знаете функции распределения?

* Дискретные распределения
* Непрерывные распределения
* Абсолютно непрерывные распределения