

**Программная реализация построения последовательности
псевдослучайных значений непрерывной случайной величины
с заданным законом распределения**

Лаптева Анастасия Игоревна

*Брянский государственный университет имени академика И.Г. Петровского
Студент*

Аннотация

В статье рассмотрены некоторые генераторы последовательности псевдослучайных чисел и приводится программная реализация построения последовательности псевдослучайных значений непрерывной случайной величины.

Ключевые слова: случайные числа, псевдослучайные числа, генераторы последовательности случайных чисел.

**A software implementation of constructing a sequence of pseudo-random
values of a continuous random variable with a predetermined distribution law**

Lapteva Anastasiya Igorevna

*Bryansk State University named by academician I.G. Petrovsky
Student*

Abstract

In the article some generators of a sequence of pseudo-random numbers are considered and a software implementation of constructing the sequence of pseudo-random values of a continuous random variable is given.

Keywords: random numbers, pseudo-random numbers, generators of a sequence of random numbers

В современных криптосистемах находят широкое применение генераторы последовательностей случайных чисел.

В последовательности случайных чисел предсказать следующее число невозможно, в отличие от псевдослучайных чисел, где последующие числа вычисляются по заданной формуле. Откуда следует преимущество псевдослучайных чисел - быстрая генерация, и недостаток — их запас ограничен.

Основными характеристиками случайных чисел являются случайность и непредсказуемость.

Одним из первых методов получения псевдослучайных чисел был метод Д. фон Неймана, в котором каждое последующее псевдослучайное число получалось возведением предыдущего в квадрат и отбрасыванием цифр с обоих концов. Это было ненадежно, и от метода быстро отказались.

Следующим методом стал конгруэнтный способ - один из широко известных методов. Линейный метод предложил в 1949 году Д.Г. Лемером, однако наиболее часто используемым методом генерации последовательностей псевдослучайных чисел является смешанный конгруэнтный метод.

Линейный конгруэнтный метод, в настоящее время, опасный для криптографии, но несмотря на это быстрый. Генерация случайных чисел не может осуществляться без наличия внешнего источника "случайности". В следствии этого стали создаваться алгоритмы генерации псевдослучайных чисел. На самом же деле действительно случайные числа генерируются из непредсказуемых физических процессов.

Одним из востребованных генераторов псевдослучайных последовательностей является генератор, основанный на использовании последовательностей Фибоначчи. Такие генераторы успешно применялись уже в конце 50-х годов прошлого столетия. Наибольшей популярностью они стали пользоваться из-за достаточной величиной периода повторяемости, высокой скоростью генерирования чисел и достаточно хорошими статистическими характеристиками.

Генератор, основанный на свойствах простых чисел Мерсенна и обеспечивающий получение высококачественных псевдослучайных чисел именуется Вихрем Мерсенна, который не зависем, имеет большой период, непредсказуем, разность между последовательными данными в выходной последовательности Вихря Мерсенна пренебрежимо мала.

Далее приводим программную реализацию последовательности псевдослучайных значений случайной величины, заданной равномерным распределением на отрезке; распределением Коши; распределением Лапласа; Распределением Леви и логистическим распределением.

```
unit Unit1;
```

```
interface
```

```
uses
```

```
Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,  
Dialogs, StdCtrls, Buttons, ExtCtrls;
```

```
type
```

```
TForm1 = class(TForm)  
    GroupBox1: TGroupBox;  
    Edit1: TEdit;  
    Label1: TLabel;  
    Edit2: TEdit;  
    Label2: TLabel;  
    RadioGroup1: TRadioGroup;  
    BitBtn1: TBitBtn;  
    Label3: TLabel;
```

```
procedure FormCreate(Sender: TObject);
procedure BitBtn1Click(Sender: TObject);
private
  { Private declarations }
public
  { Public declarations }
end;
type mas1=array[0..257]of real;
      mas2=array[0..256] of real;
const x1 = 3.6541528853610088;
const AA = 4.92867323399e-3; /// area under rectangle
const RAND_MAX=1;
var
  Form1: TForm1;
  a,b:integer;
  stairWidth:mas1;
  stairHeight:mas2;
  z: real;
implementation

{$R *.dfm}
function Uniform(a, b:real):real;
begin
  Uniform:= a + random * (b - a) / RAND_MAX;
end;

function Cauchy( x0, gamma:real):real;
var x,y: real;
begin
  repeat
    x := Uniform(-1,1);
    y := Uniform(-1,1);
  until (x * x + y * y <= 1.0) and (y <> 0.0);
  Cauchy:= x0 + gamma * x / y;
end;

function Laplace( mu, b:real):real;
var E:real;
begin
  E := Exp(1.0 / b);
  if random>0 then Laplace:= mu+E
  else Laplace:= mu-E
end;

function Levy( mu, c:real):real;
var N: real;
begin
  N := Uniform(mu,c);
  Levy:= mu + c / (N * N);
```

end;

```
function Logistic( mu, s:real):real;
```

```
begin
```

```
  Logistic:= mu + s * ln(1.0 / Uniform(0, 1) - 1);
```

```
end;
```

```
procedure TForm1.BitBtn1Click(Sender: TObject);
```

```
var rand:integer;
```

```
begin
```

```
  randomize;
```

```
  a:=strtoint(edit1.Text);
```

```
  b:=strtoint(edit2.Text);
```

```
  case RadioGroup1.ItemIndex of
```

```
    0: begin
```

```
      rand:=trunc(a + random * (b - a) / RAND_MAX);
```

```
      Label3.Caption:=Label3.Caption+radiogroup1.Items[0]+inttostr(rand)+#10#13;
```

```
    end;
```

```
    1: begin
```

```
      rand:=trunc(Cauchy( a,b));
```

```
      Label3.Caption:=Label3.Caption+radiogroup1.Items[1]+inttostr(rand)+#10#13;
```

```
    end;
```

```
    2:
```

```
      begin
```

```
        rand:=trunc(Laplace( a,b));
```

```
        Label3.Caption:=Label3.Caption+radiogroup1.Items[2]+inttostr(rand)+#10#13;
```

```
      end;
```

```
    3:
```

```
      begin
```

```
        rand:=trunc(Levy( a,b));
```

```
        Label3.Caption:=Label3.Caption+radiogroup1.Items[3]+inttostr(rand)+#10#13;
```

```
      end;
```

```
    4:
```

```
      begin
```

```
        rand:=trunc(Logistic( a,b));
```

```
        Label3.Caption:=Label3.Caption+radiogroup1.Items[4]+inttostr(rand)+#10#13;
```

```
      end;
```

```
  end;
```

```
end;
```

```
procedure TForm1.FormCreate(Sender: TObject);
```

```
begin
```

```
  Label3.Caption:="";
```

```
end;
```

```
end.
```

Библиографический список

1. Воробьев Н.Н. Числа Фибоначчи. М.: Наука, 1978. 144 с.

2. Дональд Э. Кнут. Глава 3. Случайные числа // Искусство программирования. М.: Вильямс, 2000. 832 с.
3. Дроздова И. И., Жилин В. В. Генераторы случайных и псевдослучайных чисел [Текст] // Технические науки в России и за рубежом: материалы VII Междунар. науч. конф. (г. Москва, ноябрь 2017 г.). М.: Буки-Веди, 2017. С. 13-16. URL <https://moluch.ru/conf/tech/archive/286/13233/> (дата обращения: 26.12.2017).