Министерство образования и науки РФ
Государственное образовательное учреждение
высшего профессионального образования

Тульский государственный университет

КАФЕДРА АВТОМАТИКИ И ТЕЛЕМЕХАНИКИ

ГЕНЕРАЦИЯ СЛУЧАЙНЫХ ЧИСЕЛ

Лабораторная работа № 1
по курсу «Структуры и алгоритмы обработки данных»

Вариант № 4

Выполнил: студент группы 220601 _____ Белым А.А.
(подпись)
Проверил: д. ф.-м.н, проф.каф. АТМ _____ Двоенко С.Д.
(подпись)

Тула 2013

Цель работы

Изучение способов генерации последовательности случайных чисел. Написание программы, демонстрирующей изученные принципы.

Задание

Написать программу, генерирующую случайные числа, используя аддитивный генератор чисел.

Теоретическая справка

Случайной величиной называется величина, которая в результате опыта может принять то или иное значение, причем неизвестно заранее, какое именно.

Но в некотором смысле такого объекта, как случайное число, просто нет. Скажем, двойка – это случайное число? Скорее можно говорить о последовательности независимых случайных чисел с определенным законом распределения, и это означает, грубо говоря, что каждое число было получено самым произвольным образом, без всякой связи с другими членами последовательности, и что у него есть определенная вероятность оказаться в любом заданном интервале.

Равномерным называется такое распределение, при котором каждое число равновероятно. Обычно, если специально не оговорено что-либо иное, имеют в виду равномерные распределения.

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

Современная информатика широко использует псевдослучайные числа в самых разных приложениях — от метода Монте-Карло и имитационного моделирования до криптографии. При этом от качества используемых ГПСЧ напрямую зависит качество получаемых результатов. Это обстоятельство подчёркивает известный афоризм Роберта Р. Кавью из ORNL: «генерация случайных чисел слишком важна, чтобы оставлять её на волю случая».

Источники настоящих случайных чисел найти трудно. Физические шумы, такие как детекторы событий ионизирующей радиации, дробовой шум в резисторе или космическое излучение могут быть такими источниками. Однако применяются такие устройства в приложениях сетевой безопасности редко. Сложности также вызывают грубые атаки на подобные устройства.

Криптографические приложения используют для генерации случайных чисел особенные алгоритмы. Эти алгоритмы заранее определены и, следовательно, генерируют последовательность чисел, которая теоретически не может быть статистически случайной. В то же время, если выбрать хороший алгоритм, полученная численная последовательность будет проходить большинство тестов на случайность. Такие числа называют псевдослучайными числами.

Альтернативным решением является создание набора из большого количества случайных чисел и опубликование его в некотором словаре, называемом «одноразовым блокнотом». Тем не менее, и такие наборы обеспечивают очень ограниченный источник чисел по сравнению с тем количеством, которое требуется приложениям сетевой безопасности. Хотя данные наборы действительно обеспечивают статистическую случайность, они не достаточно случайны, так как злоумышленник может получить копию словаря.

Можно рассматривать генераторы случайных чисел вида

$$X_{n+1}=(X_n+X_{n-k}) \bmod m,$$

где k — достаточно большое число. Такие генераторы были предложены Грином, Смитом и Клемом. При соответствующем выборе X0, X1, ..., Xk эта формула может стать источником хороших случайных чисел.

Длина периода такого генератора не намного больше m. В статье Грина, Смита и Клема говорится, что при $k \leq 15$ последовательность не удовлетворяет тесту «проверка интервалов» (один из тестов, оценивающий критерии случайности последовательности, т.е. показывает — достаточно ли случайна последовательность), хотя при k=16 тест проходит нормально.

Схема алгоритма

На рисунке 1 представлена схема алгоритма получения случайных чисел с помощью аддитивного генератора.

Рисунок 1 – Схема алгоритма аддитивного ГСЧ

На рисунке 2 представлена схема алгоритма сдвига влево элементов массива.

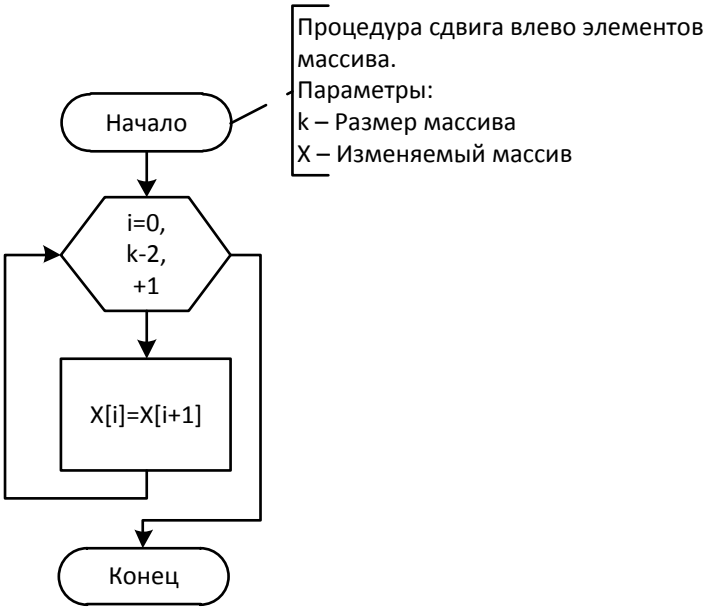

Рисунок 2 – Схема алгоритма сдвига влево элементов массива

Инструкция пользователю

Данная программа генерирует случайные числа с помощью аддитивного генератора.

Для работы программы введите модуль случайных чисел m (числа генерируются в диапазоне [0..m-1]), количество параметров генерации, и сами параметры генерации. После этого для генерации каждого следующего числа нужно нажать клавишу <Enter>, а для выхода – сочетание <Ctrl>+<C>.

Инструкция программисту

1. `void shift(int *X,int k)`

Процедура сдвига влево элементов массива.

Параметры:

k – Размер массива

X – Изменяемый массив

2. `int additive_random(const int m,const int k, int* X)`

Процедура получения случайных чисел аддитивным методом.

Параметры:

m – Модуль

k – Количество параметров

X – Массив параметров генерации

Текст программы

Далее представлен текст программы на языке C++, реализующей аддитивный генератор случайных чисел.

```
#include <iostream>
using namespace std;
void shift(int *X,int k){
    for(int i=0;i<k-1;++i){
            X[i]=X[i+1];
    };
}

int additive_random(const int m,const int k, int* X){
      int Xn=(X[k-1]+X[0])%m;
        shift(X,k);
      X[k-1]=Xn;
      return Xn;
}

int main(){
```

```
    int k,m;
      cout<<"Программа генерирует случайные числа в диапазоне [0,m-1]."<<endl;
      cout<<"Введите m."<<endl;
      cin>>m;
      cout<<"Введите k - количество параметров генератора."<<endl;
      cin>>k;
    int *X=new int[k];
      for(int i=0;i<k;++i){
          cout<<"Введите параметр номер "<<i<<"."<<endl;
          cin>>X[i];
      }
      for(;;){
        cout<<additive_random(m,k,X)<<endl;
          cout<<"Нажмите <Enter> для генерации следующего числа."<<endl;
          cin.get();
    };
    return 0;
}
```

Тестовый пример

На рисунке 3 представлен пример работы программы, реализующей аддитивный генератор случайных чисел.

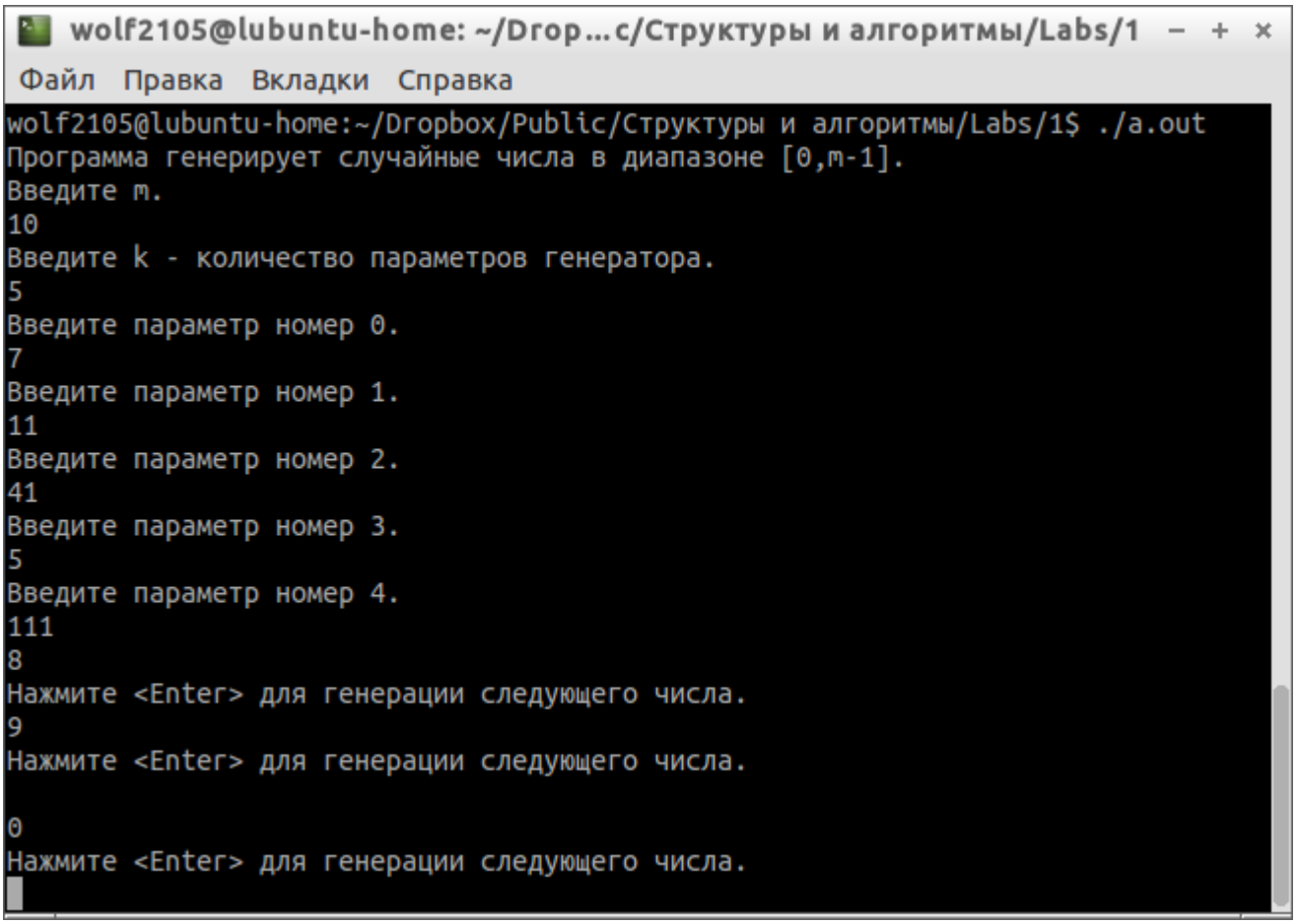

Рисунок 3— Пример работы программы с аддитивным ГСЧ

Вывод

В данной работе я познакомился с простейшими генераторами псевдослучайных чисел. Также была написана программа, реализующая

аддитивный генератор. Главной характеристикой генератора является длина периода – длина неповторяющейся последовательности генерируемых чисел. Чем больше длина периода, тем лучше генератор.