 МОСКОВСКИЙ АВИАЦИОННЫЙ ИНСТИТУТ

НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ

УНИВЕРСИТЕТ

Институт №8

«Компьютерные науки и прикладная математика» Кафедра №806

«Вычислительная математика и программирование»

Отчет по сложной задаче №6 из лабораторной работы № 2,

по учебной дисциплине

«Параллельные и распределенные вычисления»

Выполнил:

Студент 1-го курса

Гр. М80-114M-22

Гордеев Н. М.

.

(подпись, дата) .

Принял:

Семенов С. А.

.

(подпись, дата) .

Москва 2022

Оглавление

[1. Постановка задачи 2](#_Toc122004301)

[2. Метод решения 2](#_Toc122004302)

[3. Результат работы программы 3](#_Toc122004303)

[4. Вывод: 3](#_Toc122004304)

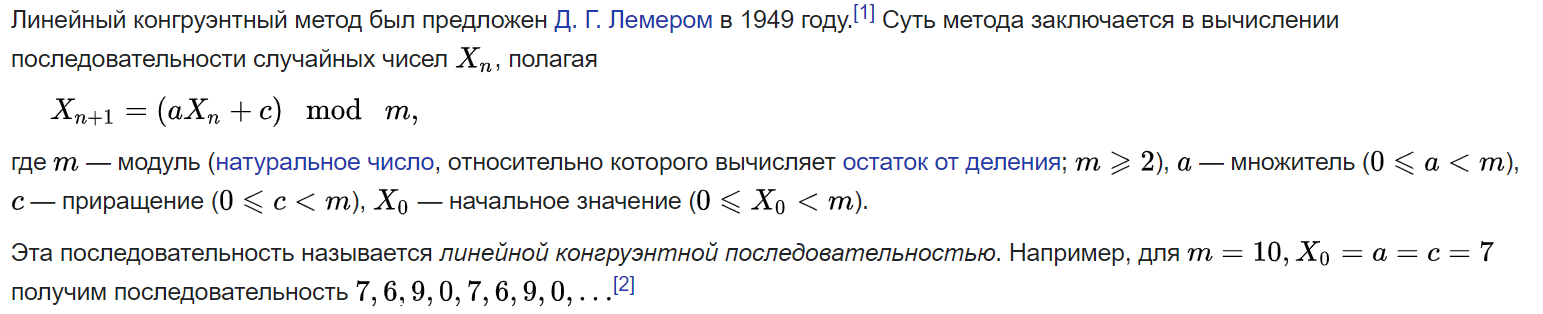
[5. Код программы 4](#_Toc122004305)

# 1. Постановка задачи

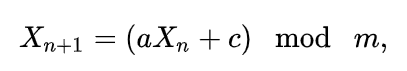
Вариант № 6. Генерация случайных чисел

# 2. Метод решения

Используем немного усовершенствованный Линейный конгруэнтный метод.

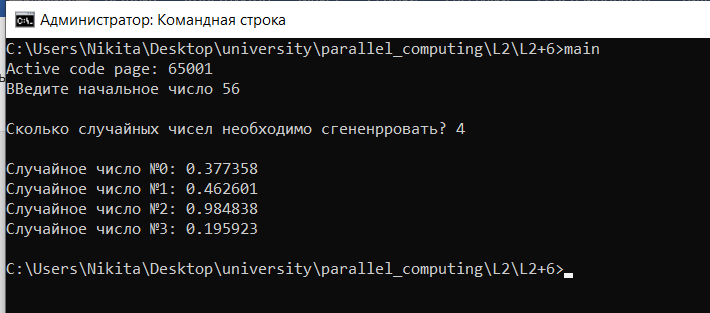


В нашем случае , будет вычисляется следующим образом:



= / m

# 3. Результат работы программы



ВВедите начальное число 56

Сколько случайных чисел необходимо сгененрровать? 4

Случайное число №0: 0.377358

Случайное число №1: 0.462601

Случайное число №2: 0.984838

Случайное число №3: 0.195923

# 4. Вывод:

Познакомился с основами программирования на cuda. Алгоритм очень простой, ничего нового для себя не подчеркнул.

# 5. Код программы

#include "cuda\_runtime.h"      // CUDA runtime

#include "device\_launch\_parameters.h"

#include <stdio.h>

#include <stdlib.h>

// Фуккция, которая выполняется на GPU

\_\_global\_\_ void Go(float x,float \*rez,int a,int m,int c,int n){

    \*rez = (float)((int)(a\*x + c) % m) / (float)m;

}

int main(){

    system("chcp 65001"); // Для Русских букв в консоли

    float x;

    int a = 593456;

    int m = 5936;

    int c = 532673;

    int n;

    printf("ВВедите начальное число ");

    scanf("%f", &x); // ввод  переменной a с клавиатуры

    printf("\n");

    printf("Сколько случайных чисел необходимо сгененрровать? ");

    scanf("%d", &n); // ввод  переменной a с клавиатуры

    printf("\n");

    float \*dev\_rez; // Адрес масива на GPU

    // Выделить память на GPU и сохранить её адрес в переменную

    cudaMalloc((void\*\*)&dev\_rez, sizeof(float));

    for(int i = 0; i < n; i++){

        Go<<<1,1>>> (x,dev\_rez,a,m,c,n);

        // Копировать данные по адресу dev\_mass, размером N\*sizeof(int) в адрес mass\_GPU. Копирование с GPU на устройство

        cudaMemcpy(&x, dev\_rez, sizeof(float), cudaMemcpyDeviceToHost);

        printf("Случайное число №%d: %f \n",i,x);

    }

    return 0;

}