Министерство образования Республики Беларусь

Учреждение образования

«Брестский Государственный технический университет»

Кафедра ИИТ

**Лабораторная работа №3**

По дисциплине «Криптографические методы защиты информации»

Тема: «Генерирование равномерно распределенных псевдослучайных последовательностей»

**Выполнил:**

Студент 2 курса

Группы ИИ-23

Романюк А.П.

**Проверил:**

Хацкевич А. С.

Брест 2024

**Цель:** Освоить основные алгоритмы программного генерирования равномерно распределенных псевдослучайных последовательностей

**Ход работы:**

**Вариант 1**



#include <iostream>

#include <vector>

#include <cmath>

#include <cstdlib>

class FibonacciRandomGenerator {

private:

std::vector<double> sequence;

int a;

int b;

public:

FibonacciRandomGenerator(int a, int b) {

int length = std::max(a, b);

this->a = a;

this->b = b;

for (int i = 0; i < length; i++) {

double random\_number = static\_cast<double>(rand()) / RAND\_MAX;

this->sequence.push\_back(random\_number);

}

}

double Next() {

if (sequence[sequence.size() - a] >= sequence[sequence.size() - b]) {

sequence.push\_back(sequence[sequence.size() - a] - sequence[sequence.size() - b]);

}

else {

sequence.push\_back(sequence[sequence.size() - a] - sequence[sequence.size() - b] + 1);

}

return sequence.back();

}

std::vector<int> generateSeq(int n, int max) {

std::vector<int> sequence;

for (int i = 0; i < n; i++) {

sequence.push\_back(Next() \* max);

}

return sequence;

}

};

int main() {

FibonacciRandomGenerator generator(55, 24);

for (auto num : generator.generateSeq(10,256)) {

std::cout << num << " ";

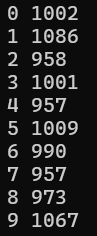
}

return 0;

}

**Случайная последовательность: **

**Распределение при генерации 10000 чисел от 0 до 9:**

****

**Вывод:** в ходе лабораторной работы я научился шифровать и сжимать информацию.