МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ "ЛЬВІВСЬКА ПОЛІТЕХНІКА"

Інститут **КНІТ**
Кафедра **ПЗ**

ЗВІТ

До лабораторної роботи №1
З дисципліни: *“Безпека програм та даних”*
На тему: *“Створення генератора псевдовипадкових чисел”*

Лектор:
доцент каф. ПЗ
Сенів М. М.

Виконав:
ст. гр. ПЗ-43
Лесневич Є. Є.

Прийняв:
ст. викладач каф. ПЗ
Угриновський Б. В.

«_____»_____2024 р.

∑=_____ .....

Львів – 2024

Тема роботи: створення генератора псевдовипадкових чисел.
Мета роботи: ознайомитись з джерелами та застосуванням випадкових чисел, алгоритмами генерування псевдовипадкових чисел та навчитись створювати програмні генератори псевдовипадкових чисел для використання в системах захисту інформації.

Теоретичні відомості

Сучасна інформатика широко використовує випадкові числа в різних програмах – від методу Монте-Карло до криптографії. Ряд алгоритмів захисту мережі, заснованих на засобах криптографії, передбачає використання випадкових чисел. Ці застосування висувають дві вимоги до послідовності випадкових чисел: випадковість і непередбачуваність.

Джерелами дійсно випадкових чисел потенційно можуть бути фізичні генератори шумів, такі як імпульсні детектори іонізуючого випромінювання, газорозрядні лампи, конденсатори з втратами струму тощо. Однак такі пристрої можуть знайти доволі обмежене застосування в додатках для захисту інформації. Туту існують проблеми як з випадковістю, так і з точністю отриманих таким методом чисел, не кажучи вже про проблеми підключення такого роду пристроїв до кожної системи в мережі.

Тому криптографічні додатки зазвичай використовують алгоритмічні методи генерування випадкових чисел. Відповідні алгоритми є детермінованими і тому породжують послідовності чисел, які не є статистично випадковими. Однак, якщо алгоритм є достатньо хорошим, породжувані ним послідовності чисел витримують багато тестів на випадковість. Такі числа часто називають псевдовипадковими.

Генератор псевдовипадкових чисел – алгоритм, що генерує послідовність чисел, елементи якої незалежні один від одного і підлягають заданому розподілу.

Найбільш популярним алгоритмом для генерування псевдовипадкових чисел є алгоритм, запропонований Лемером, який називається методом лінійного порівняння. Цей алгоритм має чотири наступних параметри.

m	модуль порівняння	$m > 0$
a	множник	$0 \le a < m$
c	приріст	$0 \le c < m$
X_0	початкове число	$0 \le X_0 < m$

Рис. 1 Параметри алгоритму метолу лінійного порівняння

Послідовність псевдовипадкових чисел {X0} отримують за допомогою ітерацій наступного співвідношення:

$$X_{n+1} = (aX_n + c) \bmod m$$

Рис. 2 Співвідношення послідовності псевдовипадкових чисел

Завдання до виконання роботи

Згідно до варіанту, наведеного в таблиці, створити програмну реалізацію генератора псевдовипадкових чисел за алгоритмом лінійного порівняння. Програма повинна генерувати послідовність із заданої при вводі кількості псевдовипадкових чисел, результати повинні як виводитись на екран, так і зберігатись у файл. Перевірити період функції генерації, зробити висновок про адекватність вибору параметрів алгоритму. У звіті навести протокол роботи програми, значення періоду функції генерації та зробити висновок про придатність цього генератора для задач криптографії.

Індивідуальне завдання

Варіант: 10

№ варіанту	Модуль порівняння, m	Множник, a	Приріст, c	Початкове значення, X_0
1.	2^{10}-1	2^5	0	2
2.	2^{11}-1	3^5	1	4
3.	2^{12}-1	4^5	2	8
4.	2^{13}-1	5^5	3	16
5.	2^{14}-1	6^5	5	32
6.	2^{15}-1	2^3	8	64
7.	2^{16}-1	3^3	13	128
8.	2^{17}-1	4^3	21	256
9.	2^{18}-1	5^3	34	512
10.	2^{19}-1	6^3	55	1024
11.	2^{20}-1	7^3	89	1
12.	2^{21}-1	8^3	144	3
13.	2^{22}-1	9^3	233	5
14.	2^{23}-1	10^3	377	7
15.	2^{24}-1	11^3	610	9
16.	2^{25}-1	12^3	987	11
17.	2^{26}-1	13^3	1597	13
18.	2^{27}-1	14^3	2584	17
19.	2^{28}-1	15^3	4181	19
20.	2^{29}-1	16^3	6765	23
21.	2^{30}-1	17^3	10946	29
22.	2^{31}-1	7^5	17711	31
23.	2^{31}	2^{16}	28657	33
24.	2^{31}-3	2^{15}	46368	37
25.	2^{31}-7	2^{14}	75025	41

Код аглоритму

```
public class RandomNumberGeneratorService : IRandomNumberGeneratorService
{
	public IEnumerable<uint> GetRandomNumbers(uint x0, uint m, uint a, uint c, uint
numOfNumbers)
	{
		if(numOfNumbers >= int.MaxValue)
		{
			throw new ArgumentException("Number of numbers cannot be int.MaxValue
or more");
		}

		var xn = x0;

		for (uint i = 0; i < numOfNumbers; ++i)
		{
			xn = (a * xn + c) % m;

			yield return xn;
		}
	}
}
```

Результати роботи

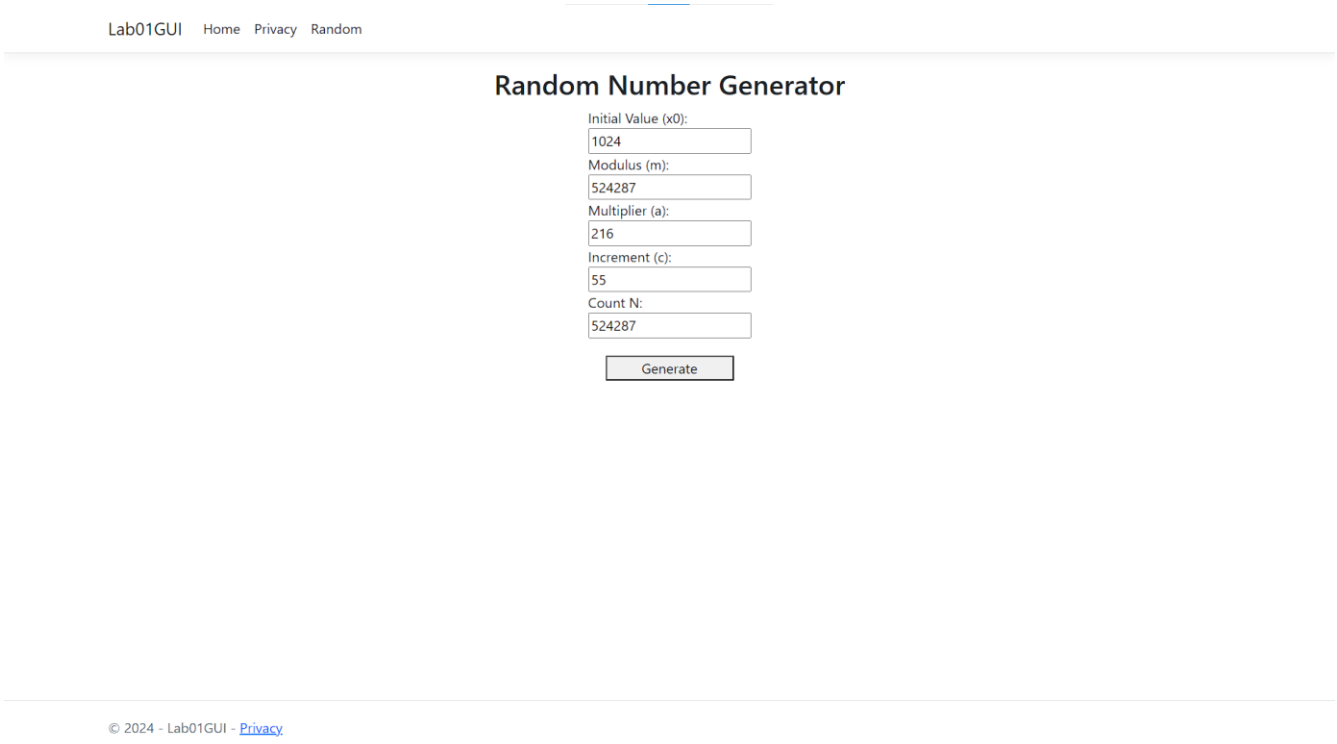

Рис. 3 Головне вікно програми

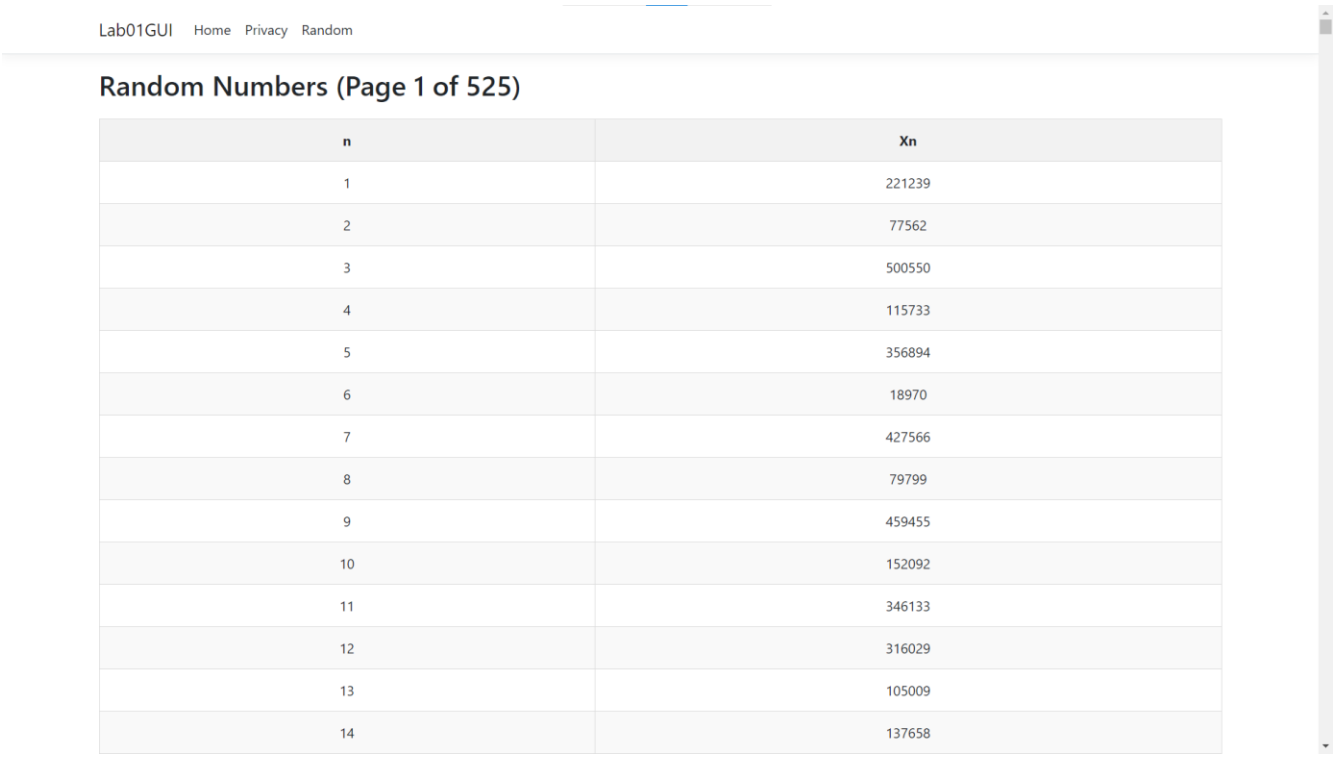

Рис. 4 Вигляд таблиці із згенерованими числами

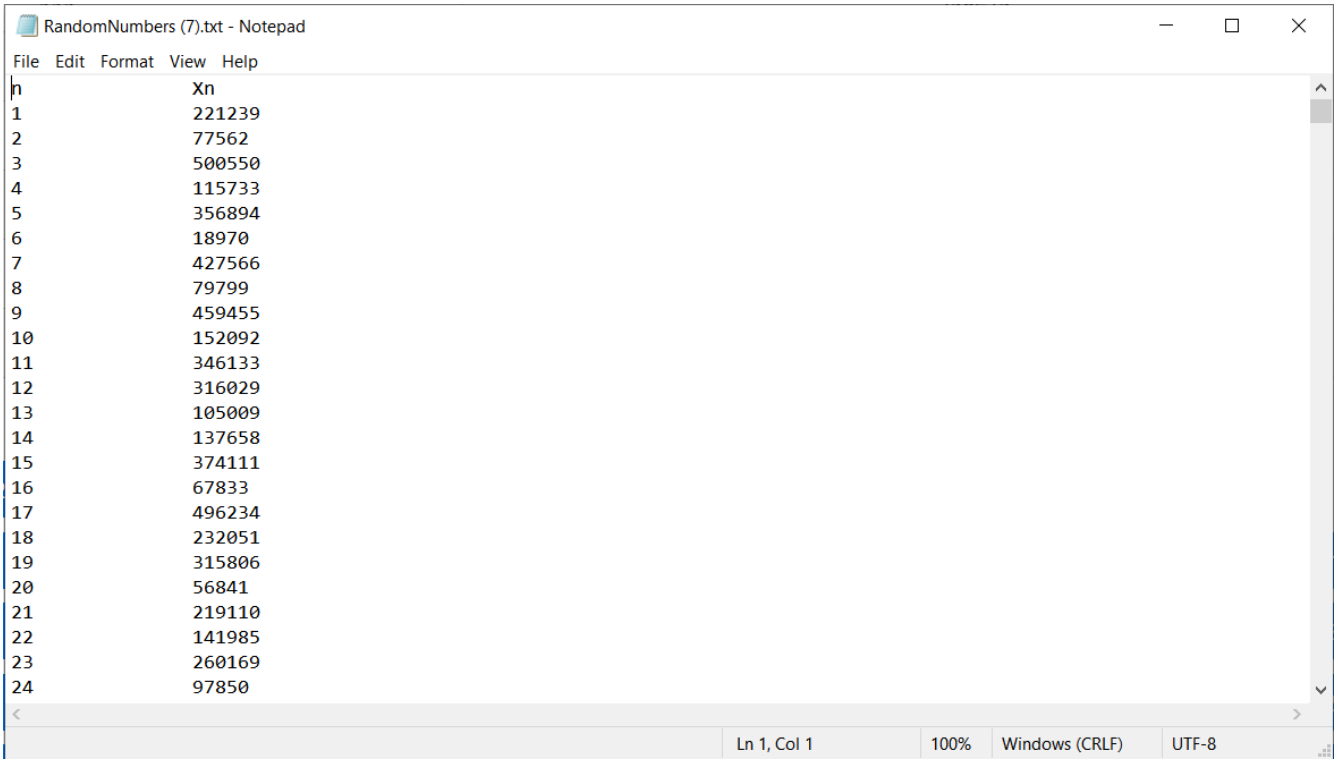

Рис. 5 Вигляд файлу із згенерованими числами

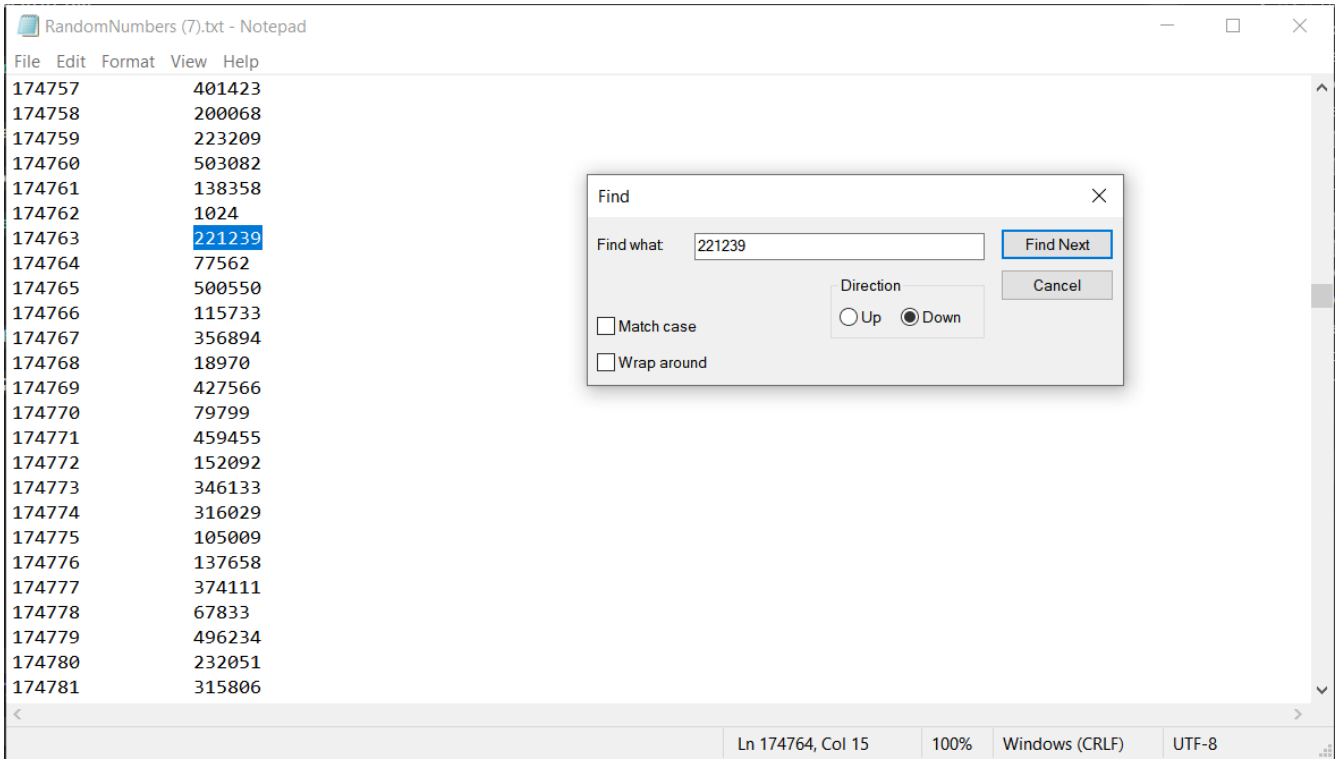

Рис. 6 Період функції генерації

Отже, оскільки згенеровані числа починаються повторюватись на 174763 числі, то період функції генерації дорівнює 174762.

Висновок про адекватність параметрів

- Модуль m = 524287 є простим числом, що є хорошим вибором для генератора, оскільки максимальний можливий період для простих чисел — це m.
- Період генератора становить лише 174762, що значно менше максимального можливого m = 524287. Це свідчить про те, що вибрані значення a та c не забезпечують повного циклу.

Придатність для криптографії

- Такий генератор не є придатним для задач криптографії. Важливим аспектом криптографічно безпечного генератора є великий період, а також непередбачуваність наступних значень.
- Генератори, що використовують алгоритм лінійного порівняння, як правило, не використовуються для криптографії через відносно короткий період і передбачуваність, особливо якщо відомі початкові параметри.

Висновки

Отже, під час виконання даної лабораторної роботи я ознайомився з джерелами та застосуванням випадкових чисел, алгоритмами генерування псевдовипадкових чисел та навчився створювати програмні генератори псевдовипадкових чисел для використання в системах захисту інформації. Створив програмну реалізацію генератора псевдовипадкових чисел за алгоритмом лінійного порівняння із виведенням результату у вигяді таблиці та збереженням у файл.