***Дмитро Ганжело****, аспірант*

*Чернівецький національний університет ім. Юрія Федьковича*

ORCID: 0000-0002-0836-4568

*e-mail: hanzhelo.dmytro@chnu.edu.ua*

***Денис Трембач****, аспірант*

*Chernivtsi National University name Yuriy Fedkovych, Chernivtsi*

*Чернівецький національний університет ім. Юрія Федьковича*

ORCID: 0000-0001-8095-4186

*e-mail:* [*trembach.denys@chnu.edu.ua*](mailto:trembach.denys@chnu.edu.ua)

***Олег Макарук****, аспірант*

*Чернівецький національний університет ім. Юрія Федьковича*

*e-mail: makaruk.oleh@chnu.edu.ua*

***Олег Павліченко****, аспірант*

*Чернівецький національний університет ім. Юрія Федьковича*

*e-mail:* [*oleh.pavl@gmail.com*](mailto:oleh.pavl@gmail.com)

**ДОСЛІДЖЕННЯ ХАРАКТЕРУ РОЗПОДІЛУ ЕЛЕМЕНТІВ ПОСЛІДОВНОСТІ ВИПАДКОВИХ ЧИСЕЛ, ЩО ГЕНЕРУЄТЬСЯ ВЕБКАМЕРОЮ**

Методика, алгоритми та програмний код генерації псевдовипадкової послідовності наявний у відкритому доступі і знаходиться наприклад, для мови Java [1,2], що робить його теоретично не стійким для кібератаки. На перше місце висувається задача створення саме апаратного криптостійкого генератора послідовності випадкових чисел (ПВЧ).

Об’єктом дослідження даної роботи є послідовності випадкових чисел (ПВЧ), що одержані з кадру веб камери. Це може бути використано для створення нескладного криптостійкого апаратного генератора ПВЧ.

Проблема, що розглядається, полягала у тому, щоб оперативно оцінити відповідність згенерованих послідовностей вимогам криптостійкості.

Метою даної роботи є попередні дослідження (попри сертифіковані тести NIST[3]) рівномірності розподілу чисел випадкових послідовностей по значенню величину, де джерелом чисел, виступають значення яскравостей пікселів кадра зображення, сгенерованого фотоматрицею веб камери, яка спроектована на основі пристрою із зарядовим зв’язком чи кремній-метал-окисел-напівпровідник матриці веб камери, наприклад, смартфону.

Обчислювальні методи, швидкодія та гнучкість класу **BufferedImage** та Webcam мови програмування Java дозволили оптимально вилучити з об’єкта кадра вебкамери одразу ж саму послідовність чисел **bytes** , що відповідають величинам яскравості пікселів матриці, без складних матричних перетворень. Сама послідовність була розміщена у просту структуру даних типу **Array** і готова для подальшого дослідження[4].

Одна із вимог до послідовності випадкових чисел — це рівномірність розподілу по значенню. Значення всіх чисел типу byte знаходиться у проміжку [-128... 127]. Тож вимога полягає у тому, що всі значення однаково мать бути присутні у згенерованій послідовності. Всього чисел 256, тоді частка кожного — 0.00390625 або приблизно 0.4%.

Завдання зводиться до того, що треба підрахувати кількість кожного числа у послідовності і вирахувати його частку. Вивести ці величини зовсім не складно, це просте групування величина-кількість. Мовою Java це робиться наступним чином:

*val map = bytes.stream()*

*.collect(Collectors.groupingBy(Function.identity(), Collectors.counting()));*

В результаті маємо стандартну структуру даних Map<key, value>, де key це значення кожного числа (елемента) у проміжку [-128 … 127], а от value буде кількість цього числа (елемента) у послідовності. Ця кількість ділиться на кількість всіх чисел у послідовності і одержуємо частку цього значення або ж перерахунок у процентах.

Щоб вирахувати статистичні данні розподілу трансформуємо цю структуру даних у простий список часток значень у процентах без різниці , якому числу ця частка належить. Одержані значення завантажуються у об’єкт класа DescriptiveStatistics мови програмування Java, який надалі вираховує всю необхідно статистику. Приклад на Java:

*DescriptiveStatistics stats = new DescriptiveStatistics();*

*map.values().stream().forEach(el -> stats.addValue(el));*

Емпіричні дані, що надає об’єкт статистики, можна пояснити на наступному прикладі:

**min**: 0.05, **max**: 1.4, **mean**: 0.39, **std dev**: 0.22, **median**: 0.37, **absent**: 0.0

Мінімальна частка у розподілі по величині значень — min: 0.05, це значає, що якийсь елемент з числового проміжка [-128..+127] у вилученій послідовності зустрічається у 0.05% випадків, а саме 38 раз із 76032 можливих. І це мінімальне значення.

Максимальне значення **max**: 1.41 — якесь число зустрічається у 1.41% випадків у послідовності (не важливо , яке саме число), а саме 1072 рази.

Середнє арифметичне (**mean**: 0.39) воно завжди стандартне для любих умов, бо 1/256 = 0.00390625, або 0.39%. Це відповідає значенню 297 зустрічей для кожного числа — ідеал.

Медіана **median**: 0.37 разом із середньо квадратичним відхиленням (std dev: 0.22) визначає характерний проміжок 0.37 +/- 0.22 (або [0.15 .. 0.59]), що покриває 67% значень розподілу.

Величина **absent**: 0.0 означає, що всі числа ряда [-128..+127] присутні у згенерованій послідовності.

Дослідження цих характеристик дає змогу встановити характер розподілу елементів елементів послідовності. Наприклад, вищенаведений випадок можна віднести до рівномірно-хаотичного розподілу.

**Висновок:** універсальність та багатофункціональність мови Java дозволяє швидко і зручно обчислити статистичні характеристики ПВЧ, що згенеровані з кадру веб камери, і перевірити на відповідність умовам криптозахисту інформації.

**References**

1. Class SecureRandom. All Implemented Interfaces. URL: <https://docs.oracle.com/javase/8/docs/api/java/security/SecureRandom.html>
2. M. Cornejo, S. Ruhault, “(In)Security of Java SecureRandom Implementations”, Journées Codage et Cryptographie, 2014. https://www-fourier.ujf-grenoble.fr/JC2/exposes/ruhault.pdf
3. National Institute of Standards and Technology (NIST), “Recommendation for the Entropy Sources Used for Random Bit Generation”, SP 800-90C, August 2022, https://doi.org/10.6028/NIST.SP.800-90C.
4. Дмитро Ганжело, Георгій Прохоров, ДОСЛІДЖЕННЯ ЧИСЛОВОЇ ВИПАДКОВОЇ ПОСЛІДОВНОСТІ, ЩО ОДЕРЖАНА З ВЕБ КАМЕРИ. (2024). Herald of Khmelnytskyi National University. Technical Sciences, 333(2), 120-124. [https://doi.org/10.31891/2307-5732-2024-333-2-18](file:///C:\Users\fahre\Downloads\ДОСЛІДЖЕННЯ%20ЧИСЛОВОЇ%20ВИПАДКОВОЇ%20ПОСЛІДОВНОСТІ,%20ЩО%20ОДЕРЖАНА%20З%20ВЕБ%20КАМЕРИ.%20(2024).%20Herald%20of%20Khmelnytskyi%20National%20University.%20Technical%20Sciences,%20333(2),%20120-124.%20https:\doi.org\10.31891\2307-5732-2024-333-2-18)