*Скиба О.В., Доманов І.О., Кравченко В.С. Державний науково-дослідний інститут
випробувань і сертифікації озброєння та військової техніки*

ДОСЛІДЖЕННЯ РИЗИКІВ ВПЛИВУ РЕЗУЛЬТАТІВ ОПЕРАЦІЇ ДІЛЕННЯ НА ЯКІСТЬ ГЕНЕРАТОРІВ ПСЕВДОВИПАДКОВИХ ВЕЛИЧИН, ВБУДОВАНИХ У ПРОГРАМНЕ ЗАБЕЗПЕЧЕННЯ

У статті наводяться результати практичного дослідження чисел, які отримуються при виконанні операції ділення. Дослідження спрямоване на з'ясування співвідношення парних та непарних чисел у частці та залишку від ділення. Аналіз проводився з огляду на ймовірне використання цих даних в алгоритмі генерування випадкових (псевдовипадкових) величин.

Результати дослідження, наведені у даній статті, підтверджують наявність передумов для видачі генератором псевдовипадкових величин таких значень, які не повною мірою задовольнятимуть критерії, зокрема щодо парності/непарності.

Авторами статті пропонується досліджувати парність/непарність за допомогою методу, який базується на основі критерію узгодженості Пірсона (критерію χ^2 -квадрат), а також шляхом дослідження довжини довгих серій парних або непарних величин.

Запропоновані підходи рекомендується використовувати для планування та проведення тестування програмного забезпечення, яке можуть бути використані у Міністерстві оборони України.

Ключові слова: випадкові величини, випадкові числа, псевдовипадкові величини, генератори випадкових величин, генератори псевдовипадкових величин, перевірка на парність та непарність, критерій χ^2 -квадрат, критерій узгодженості Пірсона, тест на довгі послідовності, тест на довгі серії.

Постановка проблеми. Генератори випадкових (або псевдовипадкових) чисел, які впроваджені у мови програмування, середовища розроблення програмних продуктів, інші популярні програмні засоби (наприклад, Microsoft Excel) та онлайн сервіси, здійснюють вибірку числових значень за певним алгоритмом. При цьому, із запуском генератора в будь-якому разі визначається початкове значення, від якого йде обчислення наступних чисел за певною втіленою функцією або шляхом вибору із таблиці заздалегідь сформульованих значень.

Отже, ті числа, які отримуються під час застосування таких генераторів, не можуть бути гарантовано випадковими і є залежними від початкових значень та певного (як правило, єдиного реалізованого у програмному засобі чи онлайн сервісі) алгоритму. Відтак, випадковість таких чисел та їх послідовностей потребує перевірки щодо незалежності від будь-якої закономірності.

Актуальність дослідження.

Програмні засоби, в які впроваджені алгоритми генерування випадкових величин, також використовуються і у Міністерстві оборони України. Зокрема до Збройних Сил все частіше надходять зразки озброєння та військової техніки, що мають вбудовані програмні засоби, через які реалізовано заходи з питань безпеки. Це стосується створення облікових записів, надання відповідних прав доступу тощо. Також випадкові величини (як числові, так і текстові або змішані), що генеруються програмними засобами, використовуються при імітаційному моделюванні та навчанні [1, С. 98].

Тому, якість таких генераторів повинна задовольняти відповідні вимоги щодо точності, що у свою чергу гарантуватиме ефективність, надійність та безпеку їх застосування, а також формувати довіру до отриманих результатів.

Аналіз останніх досліджень і публікацій.

Підходам щодо перевірки згенерованих чисел на випадковість присвячено достатньо публікацій: як українських фахівців, так і науковців з держав пострадянського простору та

інших країн. Ряд стандартів було розроблено американським Національним Інститутом Стандартів і Технологій (НІСТ).

Більшість запропонованих підходів в опрацьованих публікаціях передбачають наступні перевірки якості таких генераторів:

- рівномірність розподілу отриманих значень;
- допустимість відхилень (за критерієм узгодженості Пірсона) [2, С. 162];
- монотонність (спадання або зростання);
- повторюваність поспіль (безперервність) однакових значень (тест на довгі послідовності, тест на довгі серії);
- циклічність послідовностей (тест дискретним перетворенням Фур'є).

Існують й інші підходи, які можливо оглянути, зокрема, у [3] та [4]. Водночас, серед запропонованих методів перевірки генераторів випадкових (псевдовипадкових) величин авторами статті було виявлено відсутність перевірок на співвідношення та безперервність отриманих парних і непарних чисел.

Зв'язок авторського доробку з практичними завданнями.

Як зазначено вище, в оборонному відомстві України використовуються програмні продукти, в які впроваджуються генератори випадкових (псевдовипадкових) чисел. Вони можуть бути використані для:

- створення навчальної бойової обстановки з розіграшом декількох варіантів її розвитку. Крім того програма випадковим чином обирає вихідну оперативну обстановку із декількох створених варіантів або моделює її “з ходу” без будь-яких завчасно підготовлених даних;
- тестування курсантів або слухачів. Програмою в довільному порядку обирається комплект питань із великого набору, сформованого завчасно викладачем. Зокрема, це може забезпечити кожного, хто здає іспит, окремим неповторюваним набором питань (білетом);
- створення облікових записів персоналу, зокрема паролів для запуску програмних продуктів або входження до інформаційних систем.

В окремих військових навчальних закладах України абітурієнтам на час вступної кампанії, із застосуванням таких генераторів, присвоюються цифрові ідентифікатори, які вступники використовують під час здачі іспитів замість прізвища та імені.

Звісно, що перед прийняттям таких програм, які генерують випадкові величини, їх потрібно піддавати випробуванню та оцінюванню. Наприклад, якщо програмний продукт має вбудований генератор випадкових (псевдовипадкових) величин, то результати перевірки ступеня випадковості чисел, які ним генеруються, впливають на таку підхарактеристику якості програмного засобу як “функціональна коректність” [5].

Зазначена перевірка перебуває у площині завдань, покладених на Державний науково-дослідний інститут випробувань і сертифікації озброєння та військової техніки.

Виклад основного матеріалу. Відповідно до підходів нормативно-технічних документів [6] та інших публікацій [7], [8] генератори випадкових величин поділяються на два основні типи:

- недетерміновані – генератори, у яких механізм генерування випадкових величин використовує джерело ентропії, що ґрунтується на фізично випадкових явищах чи випадковому явищі [8]. Такими джерелами, у залежності від можливостей програмно-апаратної складової, можуть виступати: проміжок часу між останніми двома натисканнями користувача на кнопку миші; температура процесора на момент застосування генератора, рівень шуму у приміщенні тощо. Тобто, для цього здійснюється аналогово-цифрове перетворення, яке з використанням певної функції отримує вихідне значення для формування випадкового числа. Власне, це і є генератор випадкових величин;
- детерміновані – генератори, що використовують заздалегідь визначені початкові дані, оброблення яких за певним алгоритмом дозволяє отримати псевдовипадкові числа. Це генератори псевдовипадкових величин.

Якщо розробником програмного продукту не зазначено, якою мовою чи середовищем програмування він користувався, то достеменно неможливо встановити, який підхід та алгоритм використовується у програмі та які вихідні дані він обирає.

Варіантами вихідних даних можуть бути абсолютно різні величини: поточний час, положення курсору миші на екрані, завантаженість оперативної пам'яті комп'ютера, швидкість Інтернет-з'єднання, вміст буферу обміну [4, С. 29] та багато інших.

Способів оброблення вихідних даних та в подальшому отриманих чисел також досталь: множення, обчислення кореня квадратного, відкидання цілої частини, вибір символу за його порядковим номером у послідовності тощо.

Також у якості проміжного розрахунку широко застосовується і операція ділення. Зокрема, один із розповсюджених підходів до генерування псевдовипадкових величин є метод серединних квадратів, запропонований Джоном фон Нейманом. Згідно з [4, С. 31] цей метод реалізований за допомогою операцій ділення націло та подальшого обрання залишку від ділення. Такий підхід до генерації псевдовипадкових величин закладено у достатньо популярну мову програмування "C++".

Проте якість перелічених, як і будь-яких інших, підходів потребує перевірки та підтвердження. Як зазначено у [2, С. 158], "єдиним способом оцінки того, що послідовності чисел виходять дійсно випадковими, є проведення аналізу на відсутність будь-яких "переваг" у випаданні чисел у послідовності ...".

Враховуючи відсутність відомих підходів з перевірки отриманих випадкових чисел на парність/непарність, а також те, що окремими генераторами, для формування псевдовипадкових чисел, використовується операція ділення, дослідження проводилося з акцентуванням уваги на ці два аспекти.

Для цього на початковому етапі дослідження було виконано 20 000 операції ділення (за 20 ітерацій у табличному редакторі Microsoft Excel), в ході яких отримано таку ж кількість результатів, які підлягали обробленню та аналізу.

У ході кожної ітерації була виконана така послідовність дій:

а) обирання множини значень у вигляді 100 послідовних цілих чисел, що гарантувало однакову кількість (по 50) парних і непарних чисел (з метою незаангажованості перше число визначалося випадково із діапазону 1 – 1 000, а кожне наступне – шляхом збільшення попереднього числа на 1). Зазначені 100 значень були використані у якості діленого;

б) ділення кожного із цих 100 чисел на десять інших цілих чисел, які йдуть поспіль, що теж гарантує однакову кількість парних і непарних чисел (дільники обиралися за таким самим принципом, що і ділене). У результаті проведення операцій ділення у кожній ітерації було отримано 1 000 значень часток;

в) оброблення кожної з часток таким чином:

- фіксація залишку від ділення (без її видалення з частки);
- відкидання мільйонних (6-ї, 7-ї цифри і т.д. після коми);
- відкидання сотисячних;
- відкидання десятитисячних;
- відкидання тисячних;

г) перевірка кожного з чисел на парність/непарність.

Результати такого дослідження виявили, що значення, отримані у результаті ділення та подальшого обрання залишку або зменшення розрядності частки, мали явно більше парних значень. Так, парних чисел виявлено у 86 % значень, отриманих після виконання дій, передбачених у підпункті в). Рівновага відмічена у 9 %, і лише у 5 % непарних чисел виявилось більше.

До речі, у жодній ітерації не відмічено домінування непарних чисел у залишку від ділення. А саме ці значення враховуються при генеруванні псевдовипадкових величин за методом серединних квадратів [4].

Кількість парних значень, отриманих у результаті ділення та операцій з часткою

№ ітерації	Інтервал діленого	Інтервал дільника	Кількість та відсоткове значення отриманих парних чисел:				
			у залишку від ділення	у частках, з точністю до:			
				стотисячних	десяти-тисячних	тисячних	сотих
1	444-543	68-77	505 (51 %)	543 (54 %)	537 (54 %)	532 (53 %)	531 (53 %)
2	278-379	88-98	503 (50 %)	514 (51 %)	517 (52 %)	517 (52 %)	514 (51 %)
3	641-740	309-318	500 (50 %)	509 (51 %)	506 (51 %)	505 (51 %)	499 (50 %)
4	190-289	122-131	500 (50 %)	550 (55 %)	556 (56 %)	560 (56 %)	517 (52 %)
5	491-590	187-196	500 (50 %)	513 (51 %)	509 (51 %)	505 (51 %)	508 (51 %)
6	20-199	91-100	505 (51 %)	561 (56 %)	556 (56 %)	561 (56 %)	505 (51 %)
7	66-165	419-428	500 (50 %)	511 (51 %)	503 (50 %)	510 (51 %)	505 (51 %)
8	5-104	555-564	500 (50 %)	502 (50 %)	483 (48 %)	506 (51 %)	493 (49 %)
9	982-1081	73-82	503 (50 %)	576 (58 %)	520 (52 %)	528 (53 %)	522 (52 %)
10	402-501	431-440	505 (51 %)	510 (51 %)	512 (51 %)	505 (51 %)	505 (51 %)
11	365-464	62-71	505 (51 %)	520 (52 %)	519 (52 %)	513 (51 %)	518 (52 %)
12	133-142	760-769	500 (50 %)	505 (51 %)	505 (51 %)	503 (50 %)	491 (49 %)
13	300-399	303-312	505 (51 %)	515 (52 %)	515 (52 %)	505 (51 %)	507 (51 %)
14	820-919	60-69	504 (50 %)	528 (53 %)	529 (53 %)	533 (53 %)	508 (51 %)
15	732-831	154-163	505 (51 %)	509 (51 %)	512 (51 %)	503 (50 %)	503 (50 %)
16	2-101	340-349	500 (50 %)	498 (50 %)	503 (50 %)	507 (51 %)	517 (52 %)
17	732-831	82-91	505 (51 %)	518 (52 %)	514 (51 %)	500 (50 %)	512 (51 %)
18	590-689	250-259	500 (50 %)	550 (55 %)	550 (55 %)	544 (54 %)	512 (51 %)
19	777-876	69-78	503 (50 %)	552 (55 %)	542 (54 %)	529 (53 %)	533 (53 %)
20	811-910	17-26	511 (51 %)	650 (65 %)	641 (64 %)	623 (62 %)	574 (57 %)

Для візуального сприйняття узагальнених результатів, отриманих у ході перших 20 ітерацій, їх графічно відображено на рисунку 1.

Рис.1. Відсоткове значення кількості парних чисел у частках

Виходячи із значимості залишку від ділення для генерації псевдовипадкових величин, було окремо проведено ще 300 ітерацій по 1 000 обчислень у кожному, що надало ще 300 000 значень залишків. У результаті й цих досліджень, у жодній ітерації не було виявлено перевищення у залишках від ділення загальної кількості непарних чисел позначки 50 %. Натомість кількість парних в одному експерименті сягала 59 %.

Відтак, загальна кількість проведених ітерацій та обчислень (20 000 та додатково 300 000 для дослідження залишків) виявили певну не випадкову закономірність і засвідчують, що використання операції ділення з подальшим обранням залишку або його відкидання схильне давати парні числа. Якщо ці дані будуть прийматися до подальшої обробки за певним алгоритмом для генерації псевдовипадкових величин, то це може бути лише “імітацією випадку” [9, С.41].

Для дослідження парності/непарності псевдовипадкових чисел було також застосовано й математичний апарат. З огляду на характер досліджуваної властивості псевдовипадкових величин та заздалегідь відому теоретичну ймовірність отримання парних та непарних значень ($P=0,5$) [10, С. 16], було обрано метод, що базується на критерії узгодженості Пірсона [11, С. 445] (критерію χ^2 – хі-квадрат):

$$\chi^2 = \sum_{i=1}^N \frac{(f_e - f_o)^2}{f_o}, \quad (1)$$

де N – кількість категорій (для розглядуваного випадку $N=2$: парні і непарні значення);

f_e – кількість фактично отриманих i -х значень;

f_o – очікувана (теоретична, прогнозована) кількість отримання i -го значення.

У ході обчислень за формулою (1) отримано такі значення:

– для залишку від ділення $\chi^2 \approx 0,7$;

- для частки з точністю до сотисятих $\chi^2 \approx 80,72$;
- для частки з точністю до десятитисячних $\chi^2 \approx 82,00$;
- для частки з точністю до тисячних $\chi^2 \approx 47,94$;
- для частки з точністю до сотих $\chi^2 \approx 15,03$.

У подальшому із таблиці критичних точок розподілу χ^2 було обрано значення для ступеня свободи ($k=1$), що обчислено за формулою:

$$k = C - 1,$$

де C – кількість категорій (для даного дослідження $C=2$).

Таблиця 2.

Фрагмент таблиці критичних точок розподілу χ^2 для ступеня свободи $k=1$

Число ступенів свободи k	Рівень значимості α					
	0,01	0,025	0,05	0,95	0,975	0,99
1	6,6	5	3,8	0,0039	0,00098	0,00016

Враховуючи зазначення критичних точок, зрозуміло, що проведені розрахунки χ^2 лише для залишку від ділення (0,7) не перевищують порогових рівнів і можуть претендувати на їх прийняття. Натомість усі обчислення χ^2 для часток не проходять зазначену перевірку.

Однак, стійке домінування парних чисел над непарними у залишку в цілому свідчить про певну закономірність. Це зрештою обумовить недостовірний результат генератора псевдовипадкових величин, який базуватиметься на врахуванні залишку від ділення. Це є неприпустимим і, навіть небезпечним, якщо від результатів таких генераторів залежить функціонування критичних (чутливих) систем, зокрема, у сфері безпеки та оборони. “Алгоритм є статистично безпечним, якщо послідовність, яку він генерує, за своїми властивостями не поступається випадковій послідовності” [12, С. 80].

Крім того, авторами статті вбачається за доцільне піддавати перевірці не лише загальне співвідношення отриманих парних і непарних значень, а й досліджувати характер їх серійної появи під час послідовних запусків генератора. Це обумовлено тим, що як і послідовність виключно нулів або одиниць, так і послідовність поспіль виданих парних або непарних значень не може бути довільною.

Для цього пропонується застосувати метод, подібний до перевірки випадкових чисел на довгі серії (довгі послідовності). Зазначений підхід закладено в американський федеральний стандарт з обробки інформації FIPS-140-1. Суть полягає у тому, що отримані випадкові числа перетворюються у двійковий вигляд і записуються підряд. При цьому довжина послідовності повинна бути 20 000 бітів. Вважається, що послідовність випадкових чисел відповідає критерію на випадковість, якщо не відмічено жодної серії із 34 і більше нулів або одиниць.

Аналогічно до цього методу можливо провести перевірку із 20 000 отриманих випадкових значень на наявність серій із 34 і більше поспіль розташованих парних або непарних чисел. Такий підхід також описаний у [2, С. 165].

На думку авторів статті, не має значення, який саме механізм закладений у генератор випадкових величин, навіть розроблений власне програмістом, – перевірка на парність/непарність згенерованих псевдовипадкових чисел повинна бути присутня у тестах, які досліджують випадковість значень, які надаються такими генераторами.

Головні висновки та перспективи використання результатів дослідження.

Враховуючи періодичну затребуваність серед фахівців різних сфер в оперуванні випадковими незалежними величинами та доступність комп’ютерної техніки, при виборі інструментарію перевага надаватиметься програмним генераторам псевдовипадкових величин.

Проте, як засвідчує досвід та застереження фахівців в області теорії ймовірностей та дослідження генераторів псевдовипадкових величин, комп’ютерні програми не завжди спроможні надати дійсно випадкові числа та їх послідовності.

Результати дослідження, наведені у цій статті, підтверджують наявність передумов для видачі генератором псевдовипадкових величин таких значень, які не в повній мірі задовольнятимуть критерії, зокрема щодо парності/непарності.

Тому, при проведенні випробування (тестування) програмних засобів, які плануються до використання в інтересах Збройних Сил України і мають вбудовані можливості з генерації псевдовипадкових чисел, обов'язково повинна перевірятися ця функція, у т. ч. щодо співвідношення парних та непарних значень і їх послідовності. Для цього пропонується, при відпрацюванні програм та методик випробувань таких програмних засобів (у т. ч. вбудованих в зразки озброєння та військової техніки), застосовувати підходи, зазначені у цій статті.

СПИСОК ЛІТЕРАТУРИ

1. Заторський Р.А. Генератор випадкових чисел у системі дистанційного контролю знань / Р.А. Заторський, П.І. Федорук, Н.М. Дяків // Журнал Математичні машини і системи. – 2004. – Вип. № 1. – С. 98–107.
2. Акимов Г.П. Методологический подход к оценке качества случайных чисел и последовательностей / Г.П. Акимов, Е.В. Пашкина, А.В. Соловьев // Труды ИСА РАН. – 2008. – Т. 38. – С. 156–167.
3. Потий А. Статистическое тестирование генераторов случайных и псевдослучайных чисел с использованием набора статистических тестов NIST STS / Александр Потий, Светлана Орлова, Татьяна Гриненко // Правове, нормативне та метрологічне забезпечення системи захисту інформації в Україні: науково-технічний збірник. – 2001. – Вип. № 2. – С. 206–214.
4. Слеповичев И.И. Генераторы псевдослучайных чисел: учебное пособие / И.И. Слеповичев. – Саратов. – 2017. – 113 с.
5. Інженерія систем і програмних засобів. Вимоги до якості систем і програмних засобів та її оцінювання (SQuaRE). Моделі якості системи та програмних засобів: ДСТУ ISO/IEC 25010:2016 (ISO/IEC 25010:2011, IDT). – [На заміну ДСТУ ISO/IEC 25010:2015; чинний від 2018-01-01]. – Київ: ДП “УкрНДНЦ”, 2018. – 32 с. – (Національний стандарт України).
6. Інформаційні технології. Методи захисту. Генерування випадкових бітів: ДСТУ ISO/IEC 18031:2015 (ISO/IEC 18031:2011, IDT; ISO/IEC 18031:2011/Cor1:2014, IDT). – [На заміну ДСТУ ISO/IEC 18031:2014; чинний від 2017-01-01]. – Київ: ДП “УкрНДНЦ”, 2018. – (Національний стандарт України).
7. Григорьев А.Ю. Методы тестирования генераторов случайных и псевдослучайных последовательностей // Ученые записки УлГУ. Сер. Математика и информационные технологии. УлГУ. Электрон. журн. – 2017. – Вип. № 1. – С. 22–28.
8. Горбенко Ю.І. Методи та засоби генерування псевдовипадкових послідовностей / Ю.І. Горбенко, Н.В. Шапочка, Т.О. Гріненко, А.В. Нейванов, Р.І. Мордвінов // Прикладна радіоелектроніка: наук.-техн. журнал. – 2011. – Т. 10. – Вип. № 2. – С. 141–152.
9. Боррель Эмиль. Вероятность и достоверность; перевод с французского. – М.: Издательство “Наука”, 1969. – 112 с.
10. Боровков А.А. Курс теории вероятностей. – М.: Издательство “Наука”, 1972. – 287 с.
11. Пугачев В.С. Теория вероятностей и математическая статистика. – М.: Издательство “Наука”, 1979. – 495 с.
12. Кузнецов А.А. Исследование статистической безопасности генераторов псевдослучайных чисел / А.А. Кузнецов, Р.В. Королев, Ю.Н. Рябуха // Журнал “Обработка информации в сложных технических системах”. – Харків: ХУПС, 2008. – Вип. № 3(70). – С. 79-82.

Скиба Олег Васильович

науковий співробітник Державного науково-дослідного інституту випробувань і сертифікації озброєння та військової техніки, Чернігів, Україна

<https://orcid.org/0000-0002-6252-6775>

+38050-411-61-80

Доманов Ігор Олександрович

старший науковий співробітник науково-дослідної лабораторії Державного науково-дослідного інституту випробувань і сертифікації озброєння та військової техніки, Чернігів, Україна

<https://orcid.org/0000-0002-9726-0591>

+38093-052-36-04

Кравченко Віталій Сергійович

молодший науковий співробітник Державного науково-дослідного інституту випробувань і сертифікації озброєння та військової техніки, Чернігів, Україна

<https://orcid.org/0000-0002-2881-866X>

+38093-989-28-01

Oleh Skyba

Researcher of State Scientific Research Institute of Armament and Military Equipment Testing and Certification, Chernihiv, Ukraine

<https://orcid.org/0000-0002-6252-6775>

+38050-411-61-80

Igor Domanov

Senior Researcher of State Research Institute of Armament and Military Equipment Testing and Certification, Chernihiv, Ukraine, Chernihiv, Ukraine

<https://orcid.org/0000-0002-9726-0591>

+38093-052-36-04

Vitaliy Kravchenko

Junior Researcher of State Scientific Research Institute of Armament and Military Equipment Testing and Certification, Chernihiv, Ukraine

<https://orcid.org/0000-0002-2881-866X>

+38093-989-28-01

**INVESTIGATION OF RISKS OF INFLUENCE OF THE RESULTS OF THE OPERATION DIVISION
ON THE QUALITY OF PSEUDO-RANDOM VALUE GENERATORS EMBEDDED IN THE SOFTWARE
O Skyba, I Domanov and V Kravchenko**

The article is devoted to certain approaches that concerned a checking random values obtained from the generator of random (pseudorandom) values. The article provides the results of a practical research of the numbers which are obtained during the division operation.

The research was directed to find out the ratio of even and odd numbers in a quotients and remainders in the results of performing of the division operation. The analysis was carried out due to the fact that the remainder and quotient of division is widely used in various algorithms of software tools intended for generating random and pseudorandom values (numbers).

The results of the research, which are given in this scientific publication, confirm the existence of prerequisites for the outputting by the generator of pseudorandom values such values (numbers) which will not fully meet the criteria. First of all, it is a criterion which concerns parity and oddity of the numbers which can be outputted by generator of random and pseudorandom values.

The authors propose to explore parity and oddity using two methods. The first method is based on Pearson's squared test (chi-squared test). It allows you to find out the presence and admissibility of deviation of the sequence of values obtained empirically, with theoretical (predicted) values. In this case, the level of accuracy is chosen depending on the requirements for the software product as a whole and the generator of random (pseudorandom) values in particular.

The second method is based on the long series test (long sequences test), which determines the permissible number of even or odd values that follow each other. In this case, the number of allowed such sequences depends on the number of results obtained during a series of experiments.

The proposed approaches are recommended for software products planning and testing which is supposed to be used by the Ministry of Defense of Ukraine.

Keywords: random values, random numbers, pseudorandom values, generator of random values, generator pseudorandom values, parity and oddness check, chi-squared test, Pearson's squared test, long series test, long sequences test.