АСИМЕТРИЧНІ КРИПТОСИСТЕМИ ТА ПРОТОКОЛИ
КОМП'ЮТЕРНИЙ ПРАКТИКУМ №1

Побудова тестів для перевірки якості випадкових та псевдовипадкових послідовностей

1. Мета роботи

Вивчення критеріїв згоди і набуття навичок у побудові та застосуванні тестів для перевірки статистичних властивостей бінарних випадкових і псевдовипадкових послідовностей, ознайомлення з поняттям *M*-послідовності.

2. Основні теоретичні відомості

2.1. Критерії згоди та їх основні характеристики

Статистичною гіпотезою називається будь-яке твердження про вид чи властивості розподілу випадкових величин, що спостерігаються в експерименті. Звичайно її називають *основною* чи *нульовою гіпотезою* і позначають символом H_0. Якщо формулюється тільки одна гіпотеза, то правило, згідно з яким перевіряється, погоджуються наявні статистичні дані з цією гіпотезою, чи вони її спростовують, називаються *критерієм згоди*. Якщо гіпотеза H_0 однозначно фіксує розподіл спостережень, то її називають *простою*, у противному випадку – *складною*. Для побудови критерію шукають статистику $T(x)$ (тобто функцію від випадкових результатів спостережень), що характеризує відхилення емпіричних даних від гіпотетичних значень у випадку справедливості гіпотези H_0.

Припустимо, що така статистика і її розподіл при гіпотезі H_0 знайдені. Нехай Ω – множина, якій належать можливі значення величин, що спостерігаються в експерименті, і $\Im = \{t : t = T(x), x \in \Omega\}$ – множина усіх *можливих* значень статистики T; визначимо для фіксованого заздалегідь достатнього малого числа $\alpha > 0$ підмножину $\Im_{1\alpha} \subset \Im$ так, щоб імовірність здійснення події $\{T(X) \in \Im_{1\alpha}\}$ у випадку справедливості гіпотези H_0 задовольняла умові $P\{T(X) \in \Im_{1\alpha} | H_0\} \le \alpha$.

Правило перевірки гіпотези H_0 можна сформулювати в такий спосіб. *Нехай x – реалізація випадкової величини X, що спостерігалася,* $t = T(x)$ *– відповідне значення статистики T. Якщо виявиться, що* $t \in \Im_{1\alpha}$*, то в припущенні справедливості гіпотези* H_0 *відбулася малоймовірна подія і ця гіпотеза повинна бути відкинута як така, що суперечить статистичним даним. У противному випадку (тобто якщо* $t \notin \Im_{1\alpha}$*) немає підстав відмовлятися від прийнятої гіпотези і варто вважати, що спостереження не суперечать гіпотезі (чи погоджуються з нею).*

В описаній методиці статистику T називають *статистикою критерію,* а підмножину її значень $\Im_{1\alpha}$ – *критичною областю* для гіпотези H_0. Число α називають *рівнем значимості* критерію, і його можна вважати імовірністю помилкового відкидання

гіпотези H_0, коли вона вірна. У конкретних задачах її величину вибирають звичайно рівною 0,1; 0,05; 0,01 і т.д.

Кожен критерій будується для того, щоб визначити, чи мають місце ті чи інші відхилення від основної гіпотези. Характер таких відхилень може бути різним, тому треба мати критерії як універсального типу (такі, що „уловлюють", будь-які відхилення від основної гіпотези), так і призначені для виявлення відхилень тільки певного типу.

Будь-який розподіл $F_X = F$ спостереженої випадкової величини X, що може виявитися правдивим (тобто допустимим в даній ситуації), але такий, що відрізняється від гіпотетичного (тобто розподілу при основній гіпотезі H_0), називають *альтернативним* розподілом чи *альтернативою.* Сукупність всіх альтернативних розподілів називають *альтернативною гіпотезою* і позначають символом H_1.

Нехай, далі, для гіпотези H_0, що *перевіряється,* побудований деякий критерій з рівнем значимості α, заснований на *статистиці T(X)*, і нехай $\Im_{1\alpha}$ - відповідна критична область. Величину $W(F) = W(\Im_{1\alpha}; F)$, що являє собою імовірність влучення значення статистики критерію в критичну область, коли істинним розподілом спостережень є розподіл F, називають *функцією потужності критерію при альтернативі F.* У термінах функції $W(F)$ можна сказати, що критерій тим кращий (тим „потужніший"), чим більше його потужність при альтернативах. Бажаною властивістю критерію є властивість незміщеності, що означає, що повинна виконуватися умова $W(F) > \alpha,\ \forall F \in H_1$.

Важливим показником кожного критерію є трудомісткість практичної реалізації відповідного алгоритму. На практиці, коли потрібно швидко одержати відповідь, перевага нерідко віддається критерію, що просто реалізується, навіть якщо він не є оптимальним у теоретичному розумінні.

2.2. Критерій згоди хі-квадрат Пірсона для перевірки гіпотези про вид розподілу

Критерій можна використовувати для будь-яких розподілів, у тому числі і багатовимірних. Щоб скористатися цим критерієм, вибіркові дані попередньо групують, тобто переходять до частотного представлення вихідних даних. Нехай $\nu = (\nu_1, ..., \nu_N)$ – вектор частот влучення вибіркових точок у відповідні інтервали групування $\mathsf{E}_1, ..., \mathsf{E}_N \quad (\nu_1 + ... + \nu_N = n)$ і $p^{(0)} = (p_1^{(0)}, ..., p_N^{(0)})$, де $p_j^{(0)} = P(\xi \in \mathsf{E}_j \big| H_0), j = 1, ..., N$. У цьому випадку розподіл вектора частот ν при гіпотезі H_0 буде поліноміальним з параметрами n і $p^{(0)}$, і гіпотеза H_0 зводиться до гіпотези про те, що імовірності поліноміального розподілу побудованого вектора частот ν мають задані значення $p_j^{(0)}, j = 1, ..., N$. Як статистику, що характеризує відхилення вибіркових даних (тобто частот ν_j) від відповідних гіпотетичних значень (у даному випадку від середніх $E(\nu_j \big| H_0) = np^{(0)}$), приймають наступну статистику

$$X_n^2 = X_n^2(\nu) = \sum_{j=1}^{N} (\nu_j - np_j^{(0)})^2 \big/ (np_j^0) = \sum_{j=1}^{N} \nu_j^2 \big/ (np_j^{(0)}) - n,$$

а критичну область задають у виді $\Im_{1\alpha} = \{t \geq t_\alpha\}$. Точний умовний (при істинності гіпотези H_0) розподіл $\mathrm{L}(X_n^2 \big| H_0)$ цієї статистики незручний для обчислення (при заданому рівні значимості) критичної границі t_α, але для великих обсягів вибірок n статистика X_n^2 має

при гіпотезі H_0 простий граничний розподіл, що не залежить від гіпотези (тобто від чисел $p_j^{(0)}$). Справедливе наступне твердження.

Теорема. Якщо $0 < p_j^{(0)} < 1,\ j = 1,....,N$, то при $n \to \infty$

$$\mathrm{L}(X_n^2 \big| H_0) \to \chi^2(N-1)\ ,$$

де $\chi^2(N-1)$ – хі-квадрат розподіл зі $N-1$ ступенем волі, що затабульований і апроксимується за допомогою нормального розподілу при великих N.

На практиці граничний розподіл $\chi^2(N-1)$ можна використовувати з добрим наближенням уже при $n \geq 50$ і $\nu_j \geq 5$. При виконанні цих умов відповідно до теореми критичну границю t_α вибирають рівною $\chi^2_{1-\alpha,(N-1)}$, тобто $(1-\alpha)$-квантилі розподілу $\chi^2(N-1)$. Квантилі знаходяться за таблицями розподілу хі-квадрат. При великій кількості ступенів волі $(N > 30)$ можна скористатися апроксимуючою формулою $\chi^2_{1-\alpha,(N-1)} = \sqrt{2(N-1)}\, Z_{1-\alpha} + N - 1$, де $Z_{1-\alpha}$ – квантиль стандартного нормального розподілу.

Таким чином, критерій згоди χ^2 має наступний вид: *нехай задані рівень значимості α та обсяг вибірки n, і значення $h = (h_1,...,h_N)$ вектора частот $\nu = (\nu_1,...,\nu_N)$, що спостерігалися, задовольняють умовам $n \geq 50, h_j \geq 5,\ j = 1,...,N$; тоді якщо значення статистики $t = X_n^2(h)$, що спостерігалося, задовольняє нерівності $t \geq \chi^2_{1-\alpha,(N-1)}$, то гіпотезу H_0 відкидають, у противному випадку гіпотеза H_0 не суперечить результатам експерименту.*

2.3. Генератори псевдовипадкових чисел

В рамках даного комп'ютерного практикуму необхідно дослідити наступні дев'ять генераторів псевдовипадкових чисел та їх модифікацій. Зверніть увагу, що деякі з них є бітовими, а інші – байтовими.

1) Вбудований генератор псевдовипадкових чисел вашої мови програмування. Зазвичай (наприклад, в таких мовах, як Pascal та C/C++) для вбудованого датчика використовується лінійний конгруентний генератор (або *генератор Лемера* – див. нижче). Більш молоді мови надають можливість використання складних криптографічних генераторів (наприклад, мова Java має класи датчиків на основі геш-функції SHA-1).

2) *Лінійний конгруентний генератор (генератор Лемера)* обчислює послідовність $x_{n+1} = (a \cdot x_n + c) \bmod m$ для фіксованих значень *a*, *c* та *m* і деякого початкового значення x_0. Доведено, що максимальний період такого генератора дорівнює *m* і досягається він за виконання таких трьох умов:

– числа *m* та *c* повинні бути взаємнопрості;

– число $a-1$ повинно ділитись на кожен простий дільник числа *m*;

– якщо *m* ділиться на 4, то $a-1$ теж повинно ділитись на 4.

Значення x_n можуть повертатись безпосередньо або розглядатись як стани, з яких за допомогою деякого перетворення обчислюються біти (байти) вихідної послідовності. Пропонується дослідити дві модифікації лінійного конгруентного генератора, ***LehmerLow*** та ***LehmerHigh***, що генерують випадкові байти. Обидві вони обчислюють послідовність

невід'ємних 32-бітних чисел x_n; значення параметрів дорівнюють $m = 2^{32}$, $a = 2^{16} + 1$, $c = 119$, початкове значення x_0 обирається довільно, але не повинно дорівнювати нулю. ***LehmerLow*** в якості *n*-того вихідного значення повертає молодші 8 біт числа x_n, ***LehmerHigh*** – старші 8 біт x_n.

3) Генератор псевдовипадкових двійкових послідовностей ***L*20** задається рекурентною формулою (що описує функціонування лінійного регістра зсуву) $x_t = x_{t-3} \oplus x_{t-5} \oplus x_{t-9} \oplus x_{t-20}, \quad t = 20, 21...$, де $x_t \in \{0,1\}$, $\oplus$ – булеве додавання. Перші 20 елементів послідовності обираються довільним образом, але не всі одночасно рівні нулю. У даного генератора вихідна послідовність має максимальний можливий період – $(2^{20} - 1)$ бітів; така послідовність називається *M*-послідовністю.

4) Генератор псевдовипадкових двійкових послідовностей ***L*89** аналогічний попередньому генератору за будовою. Він відрізняється визначенням елементів послідовності за формулою $x_t = x_{t-38} \oplus x_{t-89}, \quad t = 89, 90...$, де перші вісімдесят дев'ять елементів обираються довільним чином, але не всі рівні нулю. Генератор ***L*89** має період $2^{89} - 1$ бітів і також генерує *M*-послідовність.

5) Генератор Джиффі (Geffe) – потоковий шифр, що генерує шифруючу гаму за рахунок нелінійної комбінації трьох лінійних регістрів зсуву (властивості даного шифру досліджувались у комп'ютерному практикумі з дисциплін «Симетрична криптографія» та «Прикладна криптологія 1»). Шифруюча гама генератора може розглядатись як псевдовипадкова двійкова послідовність.

Для виконання комп'ютерного практикуму пропонується дослідити генератор Джиффі, який використовує наступні три лінійні рекурентні послідовності, що генеруються відповідними лінійними регістрами зсуву:

регістр ***L*11**: $x_{11} = x_0 \oplus x_2$;

регістр ***L*9**: $y_9 = y_0 \oplus y_1 \oplus y_3 \oplus y_4$;

регістр ***L*10**: $s_{10} = s_0 \oplus s_3$.

Вихідна послідовність бітів (z_i) обчислюється за правилом $z_i = s_i x_i \oplus (1 \oplus s_i) y_i$.

Початкові заповнення кожного з регістрів генератора Джиффі обираються довільними, але не нульовими.

6) Генератор Вольфрама – запропонований Еріком Вольфрамом генератор, пристосований до швидкої апаратної реалізації.

Генератор використовує 32-бітні стани (однак може бути легко перебудований під будь-яку бітність). Початкове значення $r_0 \neq 0$ обирається довільним чином. Вихідна послідовність $x_0, x_1, x_2, ...$ обчислюється за таким правилом:

$$x_i = r_i \bmod 2,$$
$$r_{i+1} = (r_i <<< 1) \oplus (r_i \vee (r_i >>> 1)),$$

де операції логічного АБО ($\vee$) та виключного АБО ($\oplus$) виконуються окремо над кожним бітом векторів-операндів, а x_i є останнім бітом вектору r_i.

7) Генератор «Бібліотекар» перетворює у байтову послідовність змістовний текст довільною мовою (наприклад, випадкові фрагменти творів Мануїла Канта німецькою

мовою, що потрясають своєю фундаментальністю, або фанфікі за творами Джоан Кетлін Роулінг, які у більшості своїй і так є випадковим мотлохом). При реалізації цього генератору потрібно враховувати, що символи відповідного тексту повинні представлятись у вигляді байтів (тобто кодування використовує вісім біт на символ).

8) Генератор BM (Блюма-Мікалі) – перший криптографічний генератор, побудований на основі формальної теорії, яка доводить його властивості. Так, показано, що можливість вгадувати біти вихідної послідовності цього генератору еквівалентна можливості розв'язувати задачу дискретного логарифмування.

Нехай p велике просте число, a – примітивний корінь за модулем p (генератор групи Z_p^*). Початкове значення стану T_0, $0 \le T_0 \le p-1$ обирається випадковим чином. Послідовність станів обчислюється за рекурентним співвідношенням

$$T_{i+1} = a^{T_i} \bmod p .$$

Вихідна послідовність бітів $x_0, x_1, x_2, \ldots$ обчислюється за таким правилом:

$$x_i = \begin{cases} 1, & T_i < \dfrac{p-1}{2} \\ 0, & T_i \ge \dfrac{p-1}{2} \end{cases}.$$

Байтова модифікація генератору BM в якості вихідного байту x_i повертає таке значення k, що $\dfrac{k(p-1)}{256} < T_i \le \dfrac{(k+1)(p-1)}{256}$.

Для виконання практикуму пропонується використовувати такі параметри:

```
p = CEA42B987C44FA642D80AD9F51F10457690DEF10C83D0BC1BCEE12FC3B6093E3;
a = 5B88C41246790891C095E2878880342E88C79974303BD0400B090FE38A688356.
```

Зауважимо, що для зручності число p обрано у формі $p = 2q+1$, де число q також просте:

```
q = 675215CC3E227D3216C056CFA8F8822BB486F788641E85E0DE77097E1DB049F1.
```

9) Генератор BBS (Блюм-Блюма-Шуба) побудовано на ідеях Блюма та Мікалі, однак для генерування псевдовипадкових бітів він використовує апарат теорії чисел. Доведено, що можливість вгадувати біти вихідної послідовності цього генератору еквівалентна можливості розв'язувати задачу факторизації.

Нехай p та q – різні великі прості числа виду $4k+3$ і $n = pq$. Початкове значення $r_0 \ge 2$ обирається довільним чином. Вихідна послідовність $x_1, x_2, \ldots$ обчислюється за таким правилом:

$$r_i = r_{i-1}^2 \bmod n ,$$
$$x_i = r_i \bmod 2 ,$$

тобто x_i є останнім бітом числа r_i (зверніть увагу, що вихідні біти нумеруються з одиниці – стан r_0 не використовується для безпосереднього генерування вихідної послідовності).

Байтова модифікація генератору BBS обчислює вихідну послідовність як $x_i = r_i \bmod 256$, тобто повертаються вісім молодших біт числа r_i.

Для виконання практикуму пропонується використовувати такі числа:
p = D5BBB96D30086EC484EBA3D7F9CAEB07;
q = 425D2B9BFDB25B9CF6C416CC6E37B59C1F.

2.4. Критерії і тести для перевірки якості випадкових двійкових послідовностей

В цьому розділі наведено три критерії, що перевіряють статистичні властивості псевдовипадкових послідовностей: рівноімовірність знаків, незалежність сусідніх знаків, однорідність послідовності. Всі ці критерії є критеріями хі-квадрат Пірсона для перевірки відповідним чином сформульованих гіпотез.

Розглянемо послідовність $\{Y_j\}, j = 1,...,m$, де кожна Y_j є випадковою величиною, що приймає набір значень із алфавіту A. Всі величини Y_j мають однаковий розподіл та розглядаються як вихідні значення деякого генератору.

Послідовність $\{Y_j\}$ задовольняє *умові рівноімовірності знаків*, якщо кожна Y_j розподілена рівноімовірно на A. Таким чином, кожне значення із A повинно зустрічатись у довільній реалізації даної послідовності однакову кількість разів.

Тест на виконання умови рівноімовірності не відрізняється великою чутливістю, однак він доволі швидкий. В практичних задачах його рекомендується застосовувати в першу чергу, оскільки якщо послідовність не пройде цей тест, то немає рації застосовувати до неї інші. Також в якості підсилення можна розглядати *умову рівноімовірності серій знаків*, коли рівноімовірними в послідовності повинні бути пари, трійки, четвірки знаків тощо.

Послідовність $\{Y_j\}$ задовольняє *умові незалежності знаків*, якщо імовірність прийняти деяке значення для Y_j не залежить від того, які значення прийняли $Y_1, Y_2, ..., Y_{j-1}$. Однак перевірка такої умови зазвичай вкрай важка, тому часто розглядають більш послаблені вимоги – наприклад, значення Y_j не повинно залежати від значення Y_{j-1} (незалежність від попереднього знаку).

Послідовність $\{Y_j\}$ задовольняє *умові однорідності*, якщо для довільної реалізації вибірковий розподіл, одержаний на всій послідовності, буде співпадати із вибірковим розподілом, одержаним на довільній її підпослідовності достатньої довжини; іншими словами, на довільному фрагменті послідовність веде себе однаково. Зауважимо, що для виконання умови однорідності не важливо, який саме розподіл будуть мати Y_j. Зокрема, цей розподіл не обов'язково повинен бути рівноімовірним.

Перевірка умови однорідності в повному обсязі також доволі складна, тому на практиці перевіряють більш слабкі форми даної умови – наприклад, розбивають послідовність на окремі інтервали та перевіряють, чи співпадають вибіркові розподіли на цих інтервалах.

Сформулюємо критерії Пірсона для кожної з наведених умов. Надалі ми вважаємо, що послідовність, яка перевіряється, представлена у вигляді байтової послідовності, тобто область значень кожної випадкової величини Y_j лежить між 0 і 255. Якщо генератор, який перевіряється, повертає послідовність бітів, то вихідні значення групуються по вісім біт.

1) *Критерій перевірки рівноімовірності знаків*

Крок 1. Розглядається байтова послідовність $\{Y_j\}, j = 1,...,m$. Сформулюємо гіпотезу H_0, що полягає в тому, що всі байти послідовності рівноімовірні.

Крок 2. За значеннями послідовності $\{Y_j\}, j=1,...,m$ обчислюється статистика $\chi^2 = \sum_{j=0}^{255} \frac{(v_j - n_j)^2}{n_j}$, де v_j – число байтів *j*, що *спостерігається* у послідовності, n_j – очікуване число *байтів j* у послідовності за умови, що вірна гіпотеза H_0, тобто в даному випадку $n_j = \frac{m}{256}$ для всіх *j* = 0,…...255.

Крок 3. Обчислюється граничне значення $\chi^2_{1-\alpha}$, що відповідає рівню значимості α при $l = 255$, за формулою: $\chi^2_{1-\alpha} = \sqrt{2l}\, Z_{1-\alpha} + l$, де $Z_{1-\alpha}$ – $(1-\alpha)$-квантиль стандартного нормального розподілу.

Крок 4. Якщо $\chi^2 \le \chi^2_{1-\alpha}$, то гіпотеза H_0 не суперечить експериментальним даним, у противному випадку гіпотеза H_0 відкидається.

2) *Критерій перевірки незалежності знаків*

Крок 1. Байти послідовності послідовність $\{Y_j\}, j=1,...,m$ розглядаються парами (Y_{2i-1}, Y_{2i}), $i = 1,..n$, де $n = \left[\frac{m}{2}\right]$ – ціла частина $m/2$. Нехай H_0 – гіпотеза, що полягає в тому, що байти послідовності $\{Y_j\}, j=1,...,m$ незалежні від попереднього значення.

Крок 2. За значеннями послідовності за допомогою формули $\chi^2 = n\left(\sum_{i,j=0}^{255} \frac{v_{ij}^2}{v_i \alpha_j} - 1\right)$ обчислюється значення статистики χ^2 за умови, що вірна гіпотеза H_0 про незалежність байтів, де v_{ij} – кількість появи пари (i, j) і $\sum_{i,j=0}^{255} v_{ij} = n$, $v_i = \sum_{j=0}^{255} v_{ij}$ – кількість появи байта *i* на першому місці в парі, $\alpha_j = \sum_{i=0}^{255} v_{ij}$ – кількість появи байта *j* на другому місці в парі.

Крок 3. За формулою $\chi^2_{1-\alpha} = \sqrt{2l}\, Z_{1-\alpha} + l$ визначається граничне значення для розподілу $\chi^2_{1-\alpha}$, що відповідає рівню значимості α при $l = 255^2$.

Крок 4. Якщо $\chi^2 \le \chi^2_{1-\alpha}$, то гіпотеза H_0 не суперечить експериментальним даним, у противному випадку гіпотеза H_0 відкидається.

3) *Критерій перевірки однорідності двійкової послідовності*

Критерій перевірки однорідності аналогічний критерію для перевірки незалежності.

Крок 1. Послідовність $\{Y_j\}$ розбивається на *r* відрізків довжиною $m' = \left[\frac{m}{r}\right]$, де *m* – загальне число байтів, $n = m' \cdot r$ – загальне число байтів, що використовуються. Формулюється гіпотеза H_0 про вибір байтів з того самого розподілу.

Крок 2. За значеннями послідовності за допомогою формули $\chi^2 = n\left(\sum_{i=0}^{255}\sum_{j=0}^{r-1} \frac{v_{ij}^2}{v_i \alpha_j} - 1\right)$ обчислюється значення статистики χ^2 за умови, що вірна гіпотеза H_0 про вибір з одного розподілу, де v_{ij} – кількість появи байта *i* у відрізку *j*, $v_i = \sum_{j=0}^{r-1} v_{ij}$, $\alpha_j = \sum_{i=0}^{255} v_{ij} = m'$ (зауважимо, що α_j дорівнює довжині *j*-того відрізка; в нашому тесті всі відрізки мають однакову довжину, але це не обов'язково).

Крок 3. За формулою $\chi^2_{1-\alpha} = \sqrt{2l}\, Z_{1-\alpha} + l$ визначається граничне значення для

розподілу $\chi^2_{1-\alpha}$, що відповідає рівню значимості α при $l = 255(r-1)$.

Крок 4. Якщо $\chi^2 \le \chi^2_{1-\alpha}$, то гіпотеза H_0 не суперечить дослідним даним, у противному випадку гіпотеза H_0 відкидається.

Граничні значення $(1-\alpha)$-квантилей $Z_{1-\alpha}$ для стандартного нормального розподілу обчислюються за таблицями, наведеними у більшості посібників з теорії імовірностей та математичної статистики.

3. Порядок і рекомендації щодо виконання роботи

1. Написати програми, які реалізують генератори псевдовипадкових бітів, наведені у розділі 2.3 теоретичних відомостей, а саме:
   1) вбудований генератор вашої мови програмування;
   2) генератор LehmerLow;
   3) генератор LehmerHigh;
   4) генератор L20;
   5) генератор L89;
   6) генератор Джиффі (Geffe);
   7) генератор «Бібліотекар»;
   8) генератор Вольфрама;
   9) генератор Блюма-Мікалі BM;
   10) генератор BM_bytes (байтова модифікація генератору Блюма-Мікалі);
   11) генератор BBS;
   12) генератор BBS_bytes (байтова модифікація генератору BBS).
2. Розробити програми для реалізації трьох тестів перевірки якості двійкових послідовностей: про рівноімовірність розподілу, про незалежність і про однорідність (див. розділ 2.4 теоретичних відомостей). Програми повинні враховувати можливості введення випробовуваної послідовності різної довжини із зазначених датчиків, а також із зовні заданого файлу, можливості задавати різні значення рівня значимості α (обов'язково: $\alpha = 0.01; 0.05; 0.1$, інші значення за бажанням).

У програмі повинне бути передбачене відображення на екрані комп'ютера результатів обробки послідовності, що перевіряється тестами, із зазначенням вхідних даних послідовності (її довжини, датчиків, що її генерують чи зовнішнього файлу), обраних значень α, обчислених та теоретичних значень χ^2 і висновку (у яких випадках послідовності відкидаються чи приймаються зазначеними критеріями).

3. Провести обробку трьома побудованими тестами прикладів послідовностей, згенерованих зазначеними датчиками, для значень $\alpha = 0{,}01; 0{,}05; 0{,}1$. Пам'ятайте, що для одержання статистично достовірних даних ваша послідовність повинна містити щонайменше мільйон бітів (краще декілька мільонів).

При багатократному проведенні експериментів ви повинні пам'ятати, що для одержання нових послідовностей з того чи іншого датчика треба використовувати нові стартові значення. В промислових реалізаціях для ініціалізації програмних генераторів псевдовипадкових чисел використовують апаратні датчики, що реалізують фізичні генератори. В межах даної роботи для ініціалізації можна використовувати інші програмні датчики або доступну ентропію (системні таймери, траєкторію миші, клавіатурний набір, ідентифікатори системних подій тощо).

4. Оформлення звіту

Звіт до комп'ютерного практикуму оформлюється згідно зі стандартними правилами оформлення наукових робіт, за такими винятками:

- дозволяється використовувати шрифт Times New Roman 12pt та одинарний інтервал між рядками;
- для оформлення фрагментів текстів програм дозволяється використовувати шрифт Courier New 10pt та друкувати тексти в дві колонки;
- дозволяється не починати нові розділи з окремої сторінки.

До звіту можна не включати анотацію, перелік термінів та позначень та перелік використаних джерел. Також не обов'язково оформлювати зміст.

Звіт має містити:

– мету лабораторної роботи;

– постановку задачі;

– хід роботи, опис труднощів, що виникали, та шляхів їх розв'язання;

– результати дослідження, зокрема, експериментальні результати з перевірки статистичних гіпотез, подані у вигляді таблиць або діаграм; зокрема, необхідно навести підсумкову таблицю, в якій навести для кожного досліджуваного генератору, кожного тесту та кожного рівня значущості теоретичні значення χ^2, обчислені за послідовністю значення χ^2 та результат проходження відповідного тесту;

– порівняння генераторів, що досліджувались, з точки зору статистичного аналізу та криптографічного застосування;

– висновки до роботи.

Тексти всіх програм здаються викладачеві в електронному вигляді для перевірки на плагіат. До захисту комп'ютерного практикуму допускаються студенти, які оформили звіт та пройшли перевірку програмного коду.

5. Контрольні запитання

1. Що таке статистична гіпотеза? Яка гіпотеза називається простою? Яка складною?
2. Дайте визначення критерію згоди.
3. Що таке статистика критерію, критична область і рівень значимості?
4. Який розподіл називається альтернативним?
5. Який сенс функції потужності критерію?
6. Які властивості повинна мати криптографічно якісна випадкова послідовність?
7. Що таке рівноімовірність знаків послідовності? Наведіть приклад генератора, що забезпечує таку властивість.
8. Що таке незалежність знаків послідовності? Наведіть приклад генератора, що забезпечує таку властивість.
9. Що таке однорідність послідовності? Наведіть приклад генератора, що забезпечує таку властивість.
10. За яких умов лінійний конгруентний генератор має найбільший період і чому дорівнює це значення?
11. Який період може бути в лінійної рекурентної послідовності? За яких умов забезпечується максимально можливий період?
12. Як обчислюється період генератору Джиффі?
13. Покажіть, що якщо аналітик знає числа p та q, що використовуються у

генераторі BBS, то за значенням r_n та послідовністю бітів $x_1, x_2, ..., x_n$ він може знайти значення r_1.

14. Поясніть аномальні результати генераторів **LehmerLow** та **Librarian**. Починаючи з якої довжини послідовності будуть одержуватись такі значення статистик?

6. Оцінювання комп'ютерного практикуму

За виконання комп'ютерного практикуму студент може одержати до 16 рейтингових балів; зокрема, оцінюються такі позиції:

– реалізація програм – до шести балів (в залежності від правильності та швидкодії);
– теоретичний захист роботи – до восьми балів;
– своєчасне виконання практичної частини – 1 бал;
– своєчасний теоретичний захист – 1 бал;
– несвоєчасне виконання роботи – (-1) бал за кожен тиждень пропуску.

Програмний код, створений під час виконання комп'ютерного практикуму, перевіряється на наявність неправомірних запозичень (плагіату) за допомогою сервісу *Stanford MOSS Antiplagiarism*. У разі виявлення в програмному коді неправомірних запозичень реалізація програм оцінюється у 0 балів, а за виконання практикуму студент одержує штраф (–10) балів.

Студенти допускаються до теоретичного захисту тільки за умови оформленого звіту з виконання практикуму та проходження перевірки програмного коду.