УДК 004.056.2

О. Я. Матов[1], В. С. Василенко[2]
[1]Інститут проблем реєстрації інформації НАН України
вул. М. Шпака, 2, 03113 Київ, Україна
[2]Національний авіаційний університет
вул. Космонавта Комарова, 1, 03056 Київ, Україна

Алгоритми кодування інформаційних об'єктів у коді умовних лишків

Розглянуто можливості застосування алгоритмів для кодування інформаційних об'єктів у коді умовних лишків.

Ключові слова: *кодування, інформаційні об'єкти, цілісність.*

Вступ

Захист інформаційних об'єктів від спотворень в обчислювальних чи телекомунікаційних мережах (ТКС), або забезпечення основних функціональних характеристик захищеності інформації можна здійснювати під час її обробки, зберіганні та передачі. Вважається, що система захисту забезпечує цілісність інформації, якщо вона зберігається або передається своєчасно (з мінімальним часом затримки повідомлень), достовірної та повної, тобто захищеної від ненавмисних і зловмисних спотворень. Серед основних способів (механізмів) забезпечення цілісності (і в раніше певному сенсі — доступності) інформації в автоматизованих системах слід виділяти застосування різного роду завадостійких корегуючих кодів (ЗКК), які дозволяють реалізувати програмні, апаратурні або програмно-апаратурні засоби виявлення та усунення спотворень.

У таких кодах для забезпечення контролю цілісності інформаційних об'єктів і відновлення зруйнованої інформації до складу цієї інформації включають надлишкову інформацію — хеш-функцію, ознаку цілісності або контрольну ознаку (залежно від термінології, прийнятої у задачах контролю цілісності або завадостійкого кодування). Ця ознака цілісності — своєрідний образ, відображення цієї інформації, процедура формування якого відома, і який з дуже високою ймовірністю відповідає інформації, що захищається. Процедура обчислення цієї ознаки є *кодуванням інформаційного об'єкта*.

При кодуванні за правилами відповідного коду між інформацією, що захищається, і ознаками цілісності встановлюється регулярний (функціональний) односторонній зв'язок (процедури розрахунку контрольної ознаки за початковою інформацією відомі, а процедури зворотного розрахунку вихідної інформації за

контрольними ознаками часто не існує). Контроль цілісності (перевірка наявності або відсутності спотворень) — *декодування інформаційного об'єкта*, яке зводиться при цьому до тих чи інших процедур перевірки наявності зазначеного регулярного (функціонального) однобічного зв'язку між ознаками цілісності і прийнятої з каналу зв'язку інформацією. При виявленні порушення зазначеного зв'язку встановлюється факт наявності таких спотворень, а за певних умов, і їхнього місця та величини (характеру). За відсутності порушення цього зв'язку встановлюється факт відсутності спотворень.

Постановка задачі

Процедури кодування та декодування інформаційних об'єктів ґрунтуються на застосуванні тих чи інших завадостійких кодів Характеристиками цих процедур є як характеристики застосованого коду (значність, надлишковість, відносна швидкість, імовірності виявлення або невиявлення спотворень), так і характеристики відповідних алгоритмів при їхній програмній чи апаратурній реалізації. Однією з таких характеристик, яка безпосередньо впливає на вже згадану доступність інформаційних об'єктів, є потенційна швидкість виконання процедур кодування та декодування. Можливим шляхом підвищення потенційної швидкості виконання процедур кодування та декодування є розпаралелювання обчислювальних процесів і зменшення потрібної ємності запам'ятовуючих пристроїв чи потрібних інформаційних масивів за рахунок зменшення розрядності відповідних операндів.

Тому в статті ставиться задача вибору, розроблення та аналізу алгоритмів кодування-декодування коду умовних лишків з погляду швидкодії та ємності запам'ятовуючих пристроїв чи потрібних інформаційних масивів.

Характеристика коду умовних лишків

Нагадаємо, що в коді умовних лишків початковий інформаційний об'єкт, представлений довільним, наприклад двійковим кодом, розглядається як деяке умовне число A і розбивається на групи розрядів. Кожна з таких n груп α_i (i = 1, 2, …, n) умовно вважається лишком від розподілу цього умовного числа A на сукупність із так званих основ p_i, які визначають систему числення в лишкових класах $\alpha_i = A \bmod p_i$. Вимогою до таких основ p_i є те, що вони повинні бути взаємно простими, їхня величина повинна перевищувати величину максимально можливого значення «лишку» α_i відповідної групи. Окрім того, для забезпечення однозначної відповідності між визначеними системами числення, добуток згаданих основ, що визначає величину робочого діапазону представлення чисел у такій умовній системі числення в лишкових класах $P = \prod_{i=1}^{n} p_i$, повинен бути не меншим ніж максимальне значення інформаційного об'єкта, що розглядається як число $A_{\max}$: $P \geq A_{\max}$. Ознака цілісності в такому коді обчислюється на етапі кодування. З цією метою вводиться ще одна надлишкова (контрольна) основа, величина якої в задачах контролю цілісності повинна перевищувати максимальну з набору основ, що визначають робочий діапазон, $p_{n+1} \geq \max p_i$ (i = 1, 2, …, n). У задачах контролю та

поновлення цілісності величина контрольної основи повинна перевищувати значення подвійного добутку $p_{n+1} \geq 2p_n \cdot p_{n-1}$, де p_n та p_{n-1} — найбільші з основ p_i ($i = 1, 2, \ldots, n$).

Можливості з виявлення, а можливо й корегування спотворень, визначаються тим, що будь-яке спотворення в одній із груп розрядів α_i переводить початкове число з робочого діапазону $[0, P = \prod_{i=1}^{n} p_i)$ до діапазону $[P, R = p_{n+1} \cdot P)$, тобто призводить до збільшення початкового числа $A < P$ на деяку величину $l_i \cdot R_i$. Тут l_i та $R_i = R/p_i$ — цілі числа. Дійсно, якщо вихідне число

$$A = \alpha_1, \alpha_2, \ldots, \alpha_i, \ldots, \alpha_n, \alpha_{n+1}$$

є спотвореним по основі p_i і має вигляд

$$\tilde{A} = \alpha_1, \alpha_2, \ldots, \tilde{\alpha}_i, \ldots, \alpha_n, \alpha_{n+1},$$

де $\tilde{\alpha}_i = \{\alpha_i + \Delta\alpha_i\} \pmod{p_i}$, то це є еквівалентним наступному перетворенню (при виконанні операцій у лишкових класах):

$$\tilde{A} = (\alpha_1, \alpha_2, \ldots, \alpha_i, \ldots, \alpha_n, \alpha_{n+1}) + (0, 0, \ldots, \Delta\alpha_i, \ldots, 0, 0) = \\ = (\alpha_1, \alpha_2, \ldots, \{\alpha_i + \Delta\alpha_i\} \pmod{p_i}, \ldots, \alpha_{n1}, \alpha_{n+1}).$$

При цьому величина спотворення ΔA перевищує величину робочого діапазону P:

$$\Delta A = (0, 0, \ldots, \Delta\alpha_i, \ldots, 0, 0) > P,$$

оскільки тільки число виду

$$\Delta A = l_i \cdot R_i = l_i \cdot R/p_i$$

має всі лишки, окрім лишка по основі p_i, такими, що дорівнюють нулю. Але $\Delta A = l_i \cdot R_i > P = R/p_{n+1}$, тобто, навіть при $l_i = 1$ величина $R/p_i > R/p_{n+1}$ з тієї причини, що $p_{n+1} > p_i$. Відтак, сума $\tilde{A} = A + \Delta A > P$, тобто спотворене число виходить за межі робочого діапазону P і попадає до діапазону $[P, R)$.

Отже *такі факти* як: неспотворені числа не перевищують величини робочого діапазону $A \leq P$ та, навпаки, спотворені числа перевищують величину робочого діапазону $\Delta A > P$, *можна використати в процедурах як кодування, так і декодування*.

Кодування інформаційних об'єктів з використанням процедури нулізації

Кодування з використанням процедури нулізації [2] є одним із механізмів, що використовують властивості завадостійкого кодування, які має система лишкових класів.

При формуванні ознаки цілісності, контрольної ознаки чи кодуванні виконується послідовність операцій, яка зводяться до того, що по першим n лишкам α_i ($i = 1, 2, \ldots, n$) числа

$$A^{/} = (\alpha_1, \alpha_2, \ldots, \alpha_i, \ldots, \alpha_n, \alpha_{n+1})$$

послідовно формуються так звані мінімальні числа вигляду:

$$t_1 = (\alpha_1,\ \alpha_2^{(1)},\ \alpha_3^{(1)}, \ldots,\ \alpha_n^{(1)}, \alpha_k^{(1)}),$$
$$t_2 = (0, (\alpha_2 - \alpha_2^{(1)}) \ (\mathrm{mod}\, p_2),\ \alpha_3^{(2)}, \ldots,\ \alpha_n^{(2)},\ \alpha_k^{(2)}),$$
$$t_3 = (0, 0, (\alpha_3 - \alpha_3^{1} - \alpha_3^{2}) \ (\mathrm{mod}\, p_3),\ \alpha_4^{(3)}, \ldots,\ \alpha_n^{(3)}, \alpha_k^{(3)}),$$
$$\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots$$
$$t_n = (0, 0, 0, \ldots, (\alpha_n - \sum_{j=1}^{n-1} \alpha_n^{(j)}) \ (\mathrm{mod}\, p_n), \alpha_k^{(n)}).$$

Звернемо увагу на те, що кожне з таких мінімальних чисел може бути представленим у вигляді:

$$t_i = v_i \cdot \prod_{j=1}^{i-1} p_i \ .$$

З урахуванням того, що в системі лишкових класів

$$t_i \, (\mathrm{mod}\, p_i) = \alpha_i^{i-1} = \{\alpha_i - \sum_{j=1}^{i-1} \alpha_i^{(j)}\} \ (\mathrm{mod}\, p_i) = v_i \cdot \prod_{j=1}^{i-1} p_i \ (\mathrm{mod}\, p_i),$$

величину v_i можна визначити як

$$v_i = \{\alpha_i^{i-1} / \prod_{j=1}^{i-1} p_i\} (\mathrm{mod}\, p_i) = \{(\alpha_i - \sum_{j=1}^{i-1} \alpha_i^{(j)}) / \prod_{j=1}^{i-1} p_i\} \ (\mathrm{mod}\, p_i)$$

для усіх лишків α_i з номерами $i > 1$, а для першого із лишків α_1 значення $v_1 = 1$.

Сума цих чисел $T = \sum_{i=1}^{n} t_i$ має наступні дві властивості [2]. По-перше, лишки цієї суми по всім основам, окрім p_{n+1}, завжди дорівнюють лишкам вихідного числа A. По-друге, величина цієї суми завжди є меншою ніж величина робочого діапазону $T < P$, тобто величина T лежить у межах робочого діапазону і для неспотворених чисел $T = A$.

Таким чином, процес отримання величини $T = A$ є *процесом кодування* вихідного числа кодом умовних лишків (ЛУ-кодом), причому значення A залежить лише від цього вихідного числа і не залежить від невідомої при кодуванні величини лишку по контрольній основі p_{n+1}. Цей лишок α_{n+1} (контрольна ознака, хеш-функція, ознака цілісності, що розшукується) дорівнює при цьому сумі за модулем p_{n+1} проміжних величин $\alpha_k^{(i)}$ ($i = 1, 2, ..., n$), тобто

$$\alpha_{n+1} = T \,(\mathrm{mod}\, p_{n+1}) = (\sum_{i=1}^{n} \alpha_k^{(i)}) \,(\mathrm{mod}\, p_{n+1}).$$

Кодування інформаційних об'єктів з використанням *z*-алгоритму

Алгоритм [3] зручно розглядати після попереднього аналізу однойменного алгоритму декодування, тобто алгоритму виявлення наявності та корегування можливих спотворень. При цьому для виявлення спотворень в *z*-алгоритмі використовується уже відмічений факт, що спотворене число виходить за межі робочого діапазону:

$$\widetilde{A} \geq P. \tag{1}$$

Як і в попередньому алгоритмі переведення в позиційну систему числення чисел з можливими спотвореннями, величина яких виходить за межі робочого діапазону [0, *P*) і попадає до діапазону [*P*, *R*), слід ввести та використовувати надлишковість у вигляді лишку α_{n+1} за контрольною основою p_{n+1}. Але на відміну від розглянутого вище, усі розрахунки слід виконувати в діапазоні [0, *R*) із перерахованими до цих умов константами, а отже використовувати співвідношення:

$$A_{ПСЧ} = A_{СОК} \times [B] = [\alpha_1 \alpha_2 \alpha_3 \ldots \alpha_i \ldots \alpha_{n+1}] \times \begin{bmatrix} B_1 \\ B_2 \\ B_3 \\ \ldots \\ B_{n+1} \end{bmatrix},$$

звідки

$$A_{ПСЧ} = \sum_{i=1}^{i=n+1} \alpha_i B_i - [(1/R) \sum_{i=1}^{i=n+1} \alpha_i B_i] \times R, \tag{2}$$

де n +1 — кількість умовних основ, які забезпечують в позиційній системі числення діапазон представлення чисел, потрібний для забезпечення операцій кодування-декодування; B_i — константа системи числення, її ортогональний базис, такий, що:

$$B_i = P \cdot m_i / p_i, \; (i = 1, 2, \ldots, n + 1);$$

m_i — ціле позитивне число («вага» ортогонального базису B_i), таке що:

$$B_i \,(\mathrm{mod}\, p_i) = m_i b_i \,(\mathrm{mod}\, p_i) = 1;$$

$[B]$ — матриця-стовпчик ортогональних базисів

$$B_i = R \cdot m_i / p_i, \ (i = 1, 2, \ldots, n + 1). \tag{3}$$

Підставивши вираз (3) у (2) з урахуванням (1), отримаємо:

$$\sum_{i=1}^{i=n+1} \alpha_i R \cdot m_i / p_i - [(1/R) \sum_{i=1}^{i=n+1} \alpha_i R \cdot m_i / p_i] \times R > R / p_{n+1} \geq P. \tag{4}$$

Скоротивши обидві частини (4) на R, отримаємо, що в разі наявності спотворень

$$z > 1/p_{n+1}, \tag{5}$$

де

$$Z = \sum_{i=1}^{n+1} \alpha_i m_i / p_i - [\sum_{i=1}^{n+1} \alpha_i m_i / p_i]. \tag{6}$$

Вирази (5), (6) визначають z-алгоритм декодування для ЛУ-коду, який лише визначає наявність спотворень. Цей алгоритм включає $(n + 1)$ незалежних (за необхідності одночасних) операцій множення коду i-ї групи ($i = 1, \ldots, n + 1$) на відповідну константу і потім додавання $(n + 1)$ отриманих добутків.

Для побудови алгоритму, здатного не лише визначати наявність, але й виправляти спотворення скористаємось наступними міркуваннями.

Оскільки спотворення по i-й основі, як показано вище, має величину $\Delta A = l_i \cdot R_i = l_i \cdot R/p_i$, то очевидним є нерівність

$$\tilde{A} - l_i \cdot R_i < P, \tag{7}$$

причому величина l_i визначається з виразу:

$$[\tilde{A} / R_i] = [(A + l_i \cdot R_i)/R_i] = l_i. \tag{8}$$

Тоді з урахуванням (3)–(6), (7) вираз (8) прийме вигляд:

$$z \cdot p_i - [z \cdot p_i] < p_i / p_{n+1}, \tag{9}$$

Ясно, що вираз (5) й еквівалентний йому вираз (9) справедливі лише для тієї основи p_i, в лишку якої мається спотворення. Відтак, вираз (9) дозволяє визначити місце (номер групи), де виникло спотворення. Неважко упевнитися, що величина цього спотворення:

$$\Delta \alpha_i = \{[\tilde{A} / R_i] \cdot R_i\}_{p_i} = \{[zp_i] \cdot R_i\}_{p_i}.$$

Власне виправлення зводиться до операції

$$\alpha_i = \{\tilde{\alpha}_i - \Delta \alpha_i\}_{p_i}. \tag{10}$$

Таким чином, вирази (5), (9), (10) визначають *z*-алгоритм декодування для коригуючого ЛУ-коду.

Причому, оскільки лишки по будь-яким основам є рівноправними, то усе сказане вище відноситься і до контрольної основи. Прийнявши на етапі кодування $\widetilde{\alpha}_{n+1} = 0$, отримаємо

$$\alpha_{n+1} = (p_{n+1} - P\cdot[z\cdot p_{n+1}]) \pmod{p_{n+1}}, \tag{11}$$

і тоді вирази (5), (11) визначають *z*-алгоритм кодування.

Кодування інформаційних об'єктів із використанням процедури переведення із системи лишкових класів у позиційну систему числення є спробою удосконалення попереднього варіанту [1, 3–5]. З цією метою звернемо увагу на те, що на етапі кодування, тобто при переведенні в позиційну систему числення завідомо неспотворених чисел, можна використовувати відповідне співвідношення, яке відрізняється від попереднього лише величинами констант:

$$A_{ПСЧ} = A_{СЛК} \times [b] = [\alpha_1\alpha_2\alpha_3 \ldots \alpha_i \ldots \alpha_n] \times \begin{bmatrix} b_1 \\ b_2 \\ b_3 \\ \ldots \\ b_n \end{bmatrix},$$

де n — кількість умовних основ, які забезпечують потрібний діапазон представлення чисел у позиційній системі числення; b_i *— константа системи числення, її ортогональний базис, значення якого, на відміну від аналогічних констант попереднього алгоритму є меншим у* p_{n+1} *раз, тобто* ($[B] \to [b]$, $B_i \to b_i$):

$$b_i = P\cdot m_i / p_i, \ (i = 1, 2, \ldots, n);$$

m_i — ціле позитивне число («вага» ортогонального базису b_i), таке що: $b_i \pmod{p_i} = m_i b_i \pmod{p_i} = 1$; $[b]$ — матриця-стовпчик ортогональних базисів.

Розрахунки за цим виразом є еквівалентними розрахункам за формулою $A_{ПСЧ} = \sum_{i=1}^{i=n} \alpha_i b_i$. Оскільки при виконанні цієї операції її результат, як і раніше, може виходити за межі робочого діапазону [0, P), тому, як і в попередньому алгоритмі, для відповідних розрахунків слід використовувати вираз:

$$A_{ПСЧ} = (\sum_{i=1}^{i=n} \alpha_i b_i) \bmod P = \sum_{i=1}^{i=n} \alpha_i b_i - [(1/P)\sum_{i=1}^{i=n} \alpha_i b_i] \times P, \tag{12}$$

де позначка [X] означає обчислення цілої частки від X.

Звернемо увагу на те, що обчислена таким чином величина $A_{ПСЧ}$, *по-перше*, не перевищує величини робочого діапазону. *По-друге*, вираз (12) надає змогу обчислення лишку числа $A_{ПСЧ}$ і по будь-якій іншій основі, наприклад, по надлишковій, контрольній основі p_{n+1}:

$$\alpha_{n+1} = (A_{ПСЧ}) \bmod p_{n+1} =$$
$$= \{(\sum_{i=1}^{i=n} \alpha_i b_i) \bmod P\} \bmod p_{n+1} = \{\sum_{i=1}^{i=n} \alpha_i b_i - [(1/P)\sum_{i=1}^{i=n} \alpha_i b_i] \times P,\} \bmod p_{n+1},$$

тобто дозволяє *одержати контрольну ознаку, що і є метою процедури кодування*. Це другий з можливих алгоритмів кодування.

Аналіз алгоритмів кодування-декодування коду умовних лишків

Для кодування інформаційних об'єктів із застосуванням коду умовних лишків розглянуто два відомих алгоритми (алгоритм нулізації та *z*-алгоритм) і запропоновано модифікацію *z*-алгоритму. Оцінимо ці алгоритми з погляду швидкодії, та ємності запам'ятовуючих пристроїв чи потрібних інформаційних масивів.

Перевагою алгоритму нулізації є: усі операції алгоритму здійснюються над числами з кінцевою, наперед відомою розрядністю, наслідком чого є те, що всі обчислення можуть здійснюватися з абсолютною точністю, наслідком чого є те, що виявлення факту наявності чи відсутності порушень цілісності здійснюється з імовірністю, що дорівнює одиниці.

Недоліками цього алгоритму є:

1) обчислення величини T здійснюється послідовно, оскільки значення наступної величини t_i може бути визначеним лише за відомих попередніх t_i (i = 1, 2, …, j – 1), що виключає можливість розпаралелювання процесу. За рахунок цього слід очікувати не завжди високу швидкодію цього;

2) досить велика розрядність мінімальних чисел t_i. Якщо вважати, що кодуванню підлягають числа (блоки), які складаються із n лишків по b розрядів кожен, то перше мінімальне число потребує $N = n \cdot b$ розрядів, друге — $(N - b) = b \cdot (n - 1)$,..., i-те — $(N - (i - 1) \cdot b) = b \cdot (n - i + 1)$, … розрядів, що потребує наявності запам'ятовуючих пристроїв з відповідною ємністю.

Перевагою z-алгоритму є можливість розпаралелювання операцій при визначенні величини z.

До основного із недоліків z-алгоритму слід віднести наявність серед операндів результатів розподілу на основи системи числення p_i, вимогою до яких, як відомо, є їхня взаємна простота. Це призводить до того, що серед цих основ переважну кількість складають прості числа, внаслідок чого відповідні результати ділення (типу m_i / p_i та інші) є нескінченними дробовими числами. Нескінченні дробові числа, у свою чергу, для свого представлення потребують нескінченої розрядності, що, зрозуміло, є неможливим. Отже неминучим є обмеження такої розрядності, що веде до зниження точності відповідних обчислень. Операції з такими числами

найчастіше не дадуть точного значення, тому виявлення факту наявності чи відсутності порушень цілісності здійснюється з імовірністю, що не дорівнює одиниці. Окрім того, слід відмітити наявність серед операцій *z*-алгоритму операції множення, що може привести, як наслідок, до високої вартості апаратурної реалізації кодерів-декодерів. З цієї ж причини утруднена і мікропроцесорна реалізація *z*-алгоритму.

Перевагами алгоритму кодування інформаційних об'єктів із використанням процедури переведення із системи лишкових класів у позиційну систему числення є:

1) як і в алгоритмі нулізації, усі операції здійснюються над числами з кінцевою, наперед відомою розрядністю, наслідком чого є те, що всі обчислення можуть здійснюватися з абсолютною точністю, наслідком чого є те, що виявлення факту наявності чи відсутності порушень цілісності здійснюється з імовірністю, що дорівнює одиниці;

2) можливість розпаралелювання операцій при визначенні величини $A_{ПСЧ}$, а отже й шуканого значення ознаки цілісності α_{n+1};

3) найменшу серед розглянутих алгоритмів розрядність операндів, які використовуються при обчисленні значення ознаки цілісності α_{n+1}. При кодуванні чисел (блоків), які складаються із n лишків по b розрядів кожен, розрядність кожного з операндів складе $N = n \cdot b$ розрядів, що потребує наявності запам'ятовуючих пристроїв з відповідною ємністю.

Таким чином, у статті розглянуто завадостійкий корегуючий коду умовних лишків і відповідні алгоритми кодування для застосування в задачах контролю, чи контролю та поновлення цілісності інформаційних об'єктів. Здійснено їхній аналіз, показані переваги та вади. Найбільш досконалим, на погляд авторів, є алгоритм з використанням процедури переведення із системи лишкових класів у позиційну систему числення.

1. *Василенко В.С.* Код условных вычетов (монография) / В.С. Василенко. — Saarbrucken, Deutschland: LAMBERT Academic Publishing, 2013. — 129 с. — ISBN 978-3-659-48203-8.

2. *Матов О.Я.* Узагальнені завадостійкі коди в задачах забезпечення цілісності інформаційних об'єктів. Код умовних лишків / Матов О.Я., В.С. Василенко // Реєстрація, зберігання і оброб. даних. — 2006. — Т. 6, № 3. — С. 46–66.

3. *Василенко В.С.* Узагальнені завадостійкі коди в задачах забезпечення цілісності інформаційних об'єктів в умовах природних впливів / В.С. Василенко // Правове, нормативне та метрологічне забезпечення системи захисту інформації в Україні. — 2006. — Вип. 2 (13). — С. 144−159.

4. *Василенко В.С.* Прикладна теорія інформації та кодування (монографія) / В.С. Василенко, О.В. Дубчак, О.О. Мелешко. — К.: ІКІТ НАУ, Україна; Харків: Експрес-книга, 2014. — 512 с. — ISBN 978-966-8768-61-3.

5. *Акушский И.Я.* Машинная арифметика в остаточных классах / И.Я. Акушский, Д.И. Юдицкий. — М.: Сов. радио, 1966. — 421 с.

Надійшла до редакції 10.02.2015