ISSN 2786-5460 (Print)
ISSN 2786-5479 (Online)

МІЖРЕГІОНАЛЬНА АКАДЕМІЯ УПРАВЛІННЯ ПЕРСОНАЛОМ
INTERREGIONAL ACADEMY OF PERSONNEL MANAGEMENT

МАУП

ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ ТА СУСПІЛЬСТВО

INFORMATION TECHNOLOGY AND SOCIETY

Випуск 5 (11), 2023
Issue 5 (11), 2023

Видавничий дім
«Гельветика»
2023

Рекомендовано до друку Вченою радою
Міжрегіональної Академії управління персоналом
(протокол № 1 від 9 січня 2024 року)

Інформаційні технології та суспільство / [головний редактор О. Попов]. – Київ : Міжрегіональна Академія управління персоналом, 2023. – Випуск 5 (11). – 68 с.

Журнал «Інформаційні технології та суспільство» є науковим рецензованим виданням, в якому здійснюється публікація матеріалів науковців різних рівнів у вигляді наукових статей з метою їх поширення як серед вітчизняних дослідників, так і за кордоном.

Редакційна колегія не обов'язково поділяє позицію, висловлену авторами у статтях, та не несе відповідальності за достовірність наведених даних і посилань.

Свідоцтво про державну реєстрацію друкованого засобу масової інформації
«Інформаційні технології та суспільство» Серія КВ № 24815-14755Р від 27.04.2021 р.

Відповідно до Наказу МОН України № 1290 від 30 листопада 2021 року (додаток 3) журнал включено до Переліку наукових фахових видань України (категорія Б) зі спеціальностей 121 – Інженерія програмного забезпечення, 122 – Комп'ютерні науки, 123 – Комп'ютерна інженерія, 124 – Системний аналіз, 125 – Кібербезпека, 126 – Інформаційні системи та технології.

Усі електронні версії статей журналу оприлюднюються на офіційній сторінці видання
http://journals.maup.com.ua/index.php/it

Статті у виданні перевірені на наявність плагіату за допомогою програмного забезпечення StrikePlagiarism.com від польської компанії Plagiat.pl.

Recommended for publication
by Interregional Academy of Personnel Management
(Minutes No. 1 dated 9 January 2024)

Information Technology and Society / [chief editor Oleksandr Popov]. – Kyiv : Interregional Academy of Personnel Management, 2023. – Issue 5 (11). – 68 p.

Journal «Information Technology and Society» is a peer-reviewed scientific edition, which publishes materials of scientists of various levels in the form of scientific articles for the purpose of their dissemination both among domestic researchers and abroad.

Editorial board do not necessarily reflect the position expressed by the authors of articles, and are not responsible for the accuracy of the data and references.

Print media registration certifi cate «Information Technology and Society»
series KV No. 24815-14755P dated 27.04.2021

According to the Decree of MES No. 1290 (Annex 3) dated November 30, 2021, the journal was included in the List of scientific professional publications of Ukraine (category B) in specialties 121 – Software engineering, 122 – Computer sciences, 123 – Computer engineering, 124 – Systems analysis, 125 – Cybersecurity, 126 – Information systems and technologies.

All electronic versions of articles in the collection are available on the official website edition
http://journals.maup.com.ua/index.php/it

The articles were checked for plagiarism using the software
StrikePlagiarism.com developed by the Polish company Plagiat.pl.

ЗМІСТ

CONTENTS

УДК 004.8
DOI https://doi.org/10.32689/maup.it.2023.5.1

Володимир ДОНЕЦЬ
аспірант кафедри теоретичної та прикладної системотехніки, Харківський національний університет імені В. Н. Каразіна, майдан Свободи, 6, Харків, Україна, індекс 61022 (vol.donets@gmail.com)
ORCID: 0000-0002-5963-9998

Сергій ШМАТКОВ
доктор технічних наук, професор, завідувач кафедри теоретичної та прикладної системотехніки, Харківський національний університет імені В. Н. Каразіна, майдан Свободи, 6, Харків, Україна, індекс 61022 (s.shmatkov@karazin.ua)
ORCID: 0000-0002-0298-7174

Volodymyr DONETS
Postgraduate Student at Theoretical and Applied Systems Engineering Department, V. N. Karazin Kharkiv National University, 6, Svobody Sq, Kharkiv, Ukraine, postal code 61022 (vol.donets@gmail.com)

Serhiy SHMATKOV
Doctor of Engineering Sciences, Professor, Head of the Theoretical and Applied Systems Engineering Department, V. N. Karazin Kharkiv National University, 6, Svobody Sq, Kharkiv, Ukraine, postal code 61022 (s.shmatkov@karazin.ua)

Бібліографічний опис статті: Донець, В., Шматков, С. (2023). Методи аналізу інформативності в медичних системах підтримки прийняття рішень. *Інформаційні технології та суспільство*, 5 (11), 6–13. DOI: https://doi.org/10.32689/maup.it.2023.5.1

Bibliographic description of the article: Donets, V., Shmatkov, S. (2023). Metody analizu informatyvnosti v medychnykh systemakh pidtrymky pryiniattia rishen [On sensitivity analysis methods in medical decision-support systems]. *Informatsiini tekhnolohii ta suspilstvo – Information technology and society*, 5 (11), 6–13. DOI: https://doi.org/10.32689/maup.it.2023.5.1

МЕТОДИ АНАЛІЗУ ІНФОРМАТИВНОСТІ В МЕДИЧНИХ СИСТЕМАХ ПІДТРИМКИ ПРИЙНЯТТЯ РІШЕНЬ

Анотація. *Ця стаття присвячена розробці важливої частини комп'ютеризованих систем медичного моніторингу, а саме частини системи стратифікації даних пацієнтів – методів визначення інформативності параметрів. Внутрішній стохастичний характер даних, які генеруються цими системами, потребує передових методів для визначення стану пацієнтів, що часто потребує попередньо визначеної логіки або експертного втручання. Використання методів машинного навчання для аналізу даних у системах медичного моніторингу може допомогти виявити складні взаємозв'язки між даними та станом пацієнта, зрештою покращуючи якість лікування.*

Дослідження розглядає об'єднання моделі штучної нейронної мережі з методами для визначення інформативності параметрів даних, надаючи розуміння впливу параметрів на вихід моделі. У дослідженні розглянуто розроблений градієнтний метод оцінки загальної інформативності параметрів та модифікований метод інтегрованих градієнтів для оцінки параметрів інформативності конкретних вхідних даних.

У дослідженні використовуються дані серцевих захворювань UCI, репрезентативний набір даних, що відображає типові дані пацієнтів у комп'ютерних системах медичного моніторингу. Проблеми цих даних, як упередженість, відсутні значення та висока розмірність, підкреслюють складність реальних медичних даних, створюючи значну проблему для запропонованих методів.

В роботі показано ефективність запропонованих методів та проаналізовано їх шляхом порівняння з оцінкою варіативності PCA. Запропонований метод на основі градієнтів демонструє високу обізнаність щодо важливості параметрів і враховує нелінійні зв'язки в даних. Метод інтегрованих градієнтів показує зв'язок між загальними значеннями інформативності та інформативністю конкретних даних. Результати вплинуть на розробку систем підтримки прийняття рішень для комп'ютерних систем медичного моніторингу.

Ключові слова: *аналіз чутливості, аналіз даних, штучна нейронна мережа, інтегровані градієнти, медична діагностика, прийняття рішень.*

ON SENSITIVITY ANALYSIS METHODS IN MEDICAL DECISION-SUPPORT SYSTEMS

Abstract. *This article delves into developing the vital part of computerized medical monitoring systems, namely part of the stratification of patient data – methods of identification parameters informativeness. The inherent stochastic nature of data generated by these systems necessitates advanced methods for discerning patient states, often requiring predefined logic or*

expert intervention. Leveraging Machine Learning methods for data analysis in medical monitoring systems can help uncover complex relationships between data and patient states, ultimately enhancing treatment quality.

The study explores a combination of the artificial neural network model with methods for defining data parameter informativeness, providing insights into parameter impact on the model output. The study considered developed gradient-based methods for estimating overall parameters informativeness and modified integrated gradients method for estimating informativeness parameters of specific data.

The research employs the UCI Heart Disease Data, a representative dataset mirroring typical patient data in computer medical monitoring systems. Challenges of this data: such as bias, missing values, and high dimensionality underscore the complexity of real-world medical data, posing a significant challenge for the proposed methods.

The work showed the performance of the supposed methods and analyzed them by comparing them to PCA variance estimation. The supposed gradient-based method shows high awareness of the parameter importance and consideration of nonlinearities in the data. The integrated-gradients method shows a relation between overall informativeness values and informativeness for specific data. The results will impact the development of decision-supporting systems for computer medical monitoring systems.

Key words: *sensitivity analysis, data mining, artificial neural network, integrated gradients, medical diagnosis, decision-making.*

Вступ. Важливою частиною визначення стану пацієнта в комп'ютерній системі медичного моніторингу є визначення відповідної стратегії лікування з найкращим можливим результатом шляхом аналізу зібраних даних пацієнтів. Такі системи зазвичай генерують велику кількість стохастичних даних [1]. Дані моніторингу, створені цими системами, дозволяють розрізняти стани пацієнтів, але зазвичай за допомогою попередньо визначеної логіки або допомоги експерта (лікаря) [2].

Використання методів машинного навчання для аналізу зібраних даних у комп'ютерній системі медичного моніторингу дозволяє глибше аналізувати та знаходити можливі зв'язки між даними та станами пацієнтів, що покращує якість лікування [2]. Автоматизовані системи медичного моніторингу використовуються як системи підтримки прийняття рішень експертами та для упередження катастрофічних рішень [1; 2]. Існуючі інформаційні системи частково вирішують проблему стратифікації елементів у комп'ютерних системах медичного моніторингу за допомогою методів кластеризації даних для визначення станів, але не вирішують проблеми обґрунтування прийняття рішень щодо певних станів [3]. У нашому дослідженні розглядається методи визначення інформативності параметрів як одні з елементів стратифікації комп'ютерних систем медичного моніторингу для пояснення рішень генерованих такими системами.

Метою роботи є підвищення якості стратифікації елементів у комп'ютерних системах медичного моніторингу шляхом розробки методів визначення інформативності даних у моделі навченої штучної нейронної мережі (ШНМ) та поліпшення обґрунтування прийняття рішень такими системами.

Наукова новизна дослідження полягає в розробці градієнт заснованого методу аналізу інформативності для визначення загального рівня інформативності вхідних параметрів та використання модифікації методу інтегрованих градієнтів для обґрунтуванням прийняття рішень ШНМ для певних даних в комп'ютерній системі медичного моніторингу. Запропоновані методи запровадить аналіз чутливості вхідних параметрів в розробленій моделі ШНМ.

Аналіз останніх досліджень і публікацій. Розуміння інформативності вхідних даних має вирішальне значення для обґрунтуванням роботи ШНМ у процесах прийняття рішень. Це також допомагає визначити найбільш інформативний набір параметрів і зменшити набір параметрів, необхідних для роботи комп'ютерних систем медичного моніторингу. Розглянемо можливі способи визначення інформативності параметрів:

1. Analysis of Variance (ANOVA) – це набір статистичних моделей і процедур оцінки розкиду середніх значень параметрів [4]. Статистичний аналіз вхідних параметрів може бути пов'язана з їхнім впливом на вихідні змінні. Більша варіативність може свідчити про більш значний вплив, а значить, більшу інформативність [4; 5]. *Переваги*: ефективність в обчисленні. *Недоліки*: метод здатен визначити лише лінійні зв'язки, чутливий до незбалансованих наборів даних, чутливий до викидів.

2. Feature Importance Analysis. Такі методи, як AdaBoost, Random Forests або eXtreme Gradient Boosting, можуть оцінити важливість кожного вхідного параметру. Це відбувається шляхом аналізу цих навчених моделей на результатах роботи навченої ШНМ, де значення інформативності є результатом роботи цих методів [6]. *Переваги*: простота реалізації. *Недоліки*: недостатність точності оцінки інформативності через різницю у складністю між моделлю ШНМ та моделями для оцінки інформативності.

3. Permutation importance є популярною технікою для оцінки впливу окремих вхідних параметрів на прогнози ШНМ [7]. Метод працює шляхом випадкової зміни значення певного параметра в навчальних даних і спостереження за зміною в прогнозах моделі – чим суттєвіша зміна в прогнозі моделі, тим більша важливість цього параметру. *Переваги*: простий і легкий у реалізації, працює безпосередньо з моделлю ШНМ, тому дає точну оцінку інформативності, стійкий до викидів. *Недоліки*: не ефективний

в обчисленні, може охопити вплив лише одного параметра за один крок, непридатних для моделей класифікаторів через природу їх виходів.

4. SHApley Additive exPlanations (SHAP) – потужний метод для обґрунтування прогнозів моделей машинного навчання, включаючи ШНМ, створений на основі кооперативної теорії ігор. Метод детально аналізує взаємозв'язок між вхідними параметрами та вихідними значеннями моделей машинного навчання, кількісно визначаючи внесок кожного параметру [8; 9]. *Переваги*: точність оцінок, можливість інтерпретації отриманих результатів. *Недоліки*: низька обчислювальна ефективність, обмежена масштабованість, обмеженість за можливими архітектурами ШНМ.

5. Integrated Gradients (IG) – метод пояснення прогнозів ШНМ шляхом приписування коефіцієнтів впливу окремим параметрам [10; 11]. На відміну від інших методів, IG фокусується на внутрішній обробці даних нейронної мережі під час прогнозування, пропонуючи глибший аналіз впливу параметрів на обробку даних всередині нейронної мережі. Цей метод часто використовується в згорткових ШНМ для комп'ютерного зору для виділення зон уваги на зображеннях [12], а також для тлумачення мовних моделей на базі ШНМ [11]. *Переваги*: його можна використовувати для обґрунтування прийняття рішень ШНМ для конкретних даних, ефективний в обчисленні, має градієнтну інтерпретацію. *Недоліки*: чутливість до шуму, вплив вибору базового значення параметрів може впливати на значення інформативності.

6. Gradient-based Sensitivity Analysis (GBSA) – це метод, який використовує похідні першого або вищого порядку виходів ШНМ щодо вхідних параметрів для визначення інформативності вхідних даних відносно вихідних значень [13]. *Переваги*: забезпечує точні та точні оцінки інформативності та фіксує нелінійні залежності. *Недоліки*: метод може бути неефективним в обчисленнях, складним для реалізації та перевірки на закритих моделях машинного навчання.

Основна частина. Кожен метод визначення інформативності має свої сильні та слабкі сторони. Вибір залежить від конкретних цілей аналізу, характеру даних і доступних обчислювальних ресурсів. Оскільки ми розглядали елементи стратифікації в комп'ютерних системах медичного моніторингу, ми розробили градієнтний метод визначення загальних значень інформативності та модифікований метод інтегрованих градієнтів для визначення конкретної інформативності. Градієнтний метод розроблений для нашої модифікованої моделі ШНМ, наведеної в статті [14].

Перед визначенням запропонованих методів розглянемо архітектуру моделі ШНМ, що показана в роботі [14]. Ця модель є типовою повнозв'язною багаторівневою ШНМ. Модель має функцію активації Softmax для виходів, що дозволяє вирішувати задачі багатокласової класифікації. Ми використовуємо Sigmoid як функцію активації проміжних шарів, яка може запровадити гладку нелінійність в модель. Однак можна використовувати будь-які сучасні функції активації, такі як ReLU, Tanh або інші. У зв'язку з проблемою, яка вирішує модель ШНМ, ми використовуємо функцію втрат категорійної перехресної ентропії як цільову функцію для мінімізації [14]. Основною відмінністю між цією та іншими сучасними моделями є динамічна визначальна кількість проміжних шарів і нейронів усередині, досягнута за допомогою нашої процедури прискореного навчання, описаної в дослідженні [14].

Градієнтний метод визначення загальної інформативності. Основна мета розробленого підходу полягає в тому, аби з використанням розробленої навченої моделі ШНМ отримувати значення загальної інформативності параметрів. Крім того, за необхідності ці значення можна бути використаний для зменшення кількості вхідних параметрів шляхом виділення найбільш інформативних. Тоді визначимо P – множина усіх вхідних параметрів, отже $P=\{p_i\}$, $i=[1,\dots I]$, $I=|P|$. Далі множину P представимо як ряди Тейлора зі збереженням нескінченно малих членів, це допоможе нам визначити дисперсію виходів ШНМ у вигляді лінійної функції:

$$D_{F_j}=\sum_{i=1}^{I}\left(\frac{\delta F_j}{\delta p_i}\right)^2\sigma_{p_i}^2+\sum_{i=1}^{I}\sum_{k=1,i\neq k}^{I}r_{ik}\frac{\delta F_j}{\delta p_i}\frac{\delta F_j}{\delta p_k}\sigma_{p_i}^2\sigma_{p_k}^2, \quad (1)$$

де r_{ik} – це значення кореляції між i-м та k-м параметром, $F_j(P)$ – лінійна функція, що описує взаємозв'язок між вхідними параметрами навченої моделі ШНМ та j-м виходом, де $j=[1,J]$, J – кількість класів, що визначає ШНМ.

Для аналізу поточної ШНМ, позначимо функції для представлення вихідних та вхідних шарів моделі як F_j^{out} і F_i^{in} відповідно. Дисперсія виходів ШНМ з відносно її входів враховуючи (1) та матеріал дослідження [15] визначається за допомогою наступного рівняння:

$$D_{F_j^{out}|F_i^{in}}=\left(\frac{\delta F_j^{out}}{\delta F_i^{in}}\right)^2\sigma_{\delta F_i^{in}}^2+\left(\sum_{k=1,\ k\neq 1}^{I}r_{ik}\frac{\delta F_j^{out}}{\delta F_t^{in}}\sigma_{\delta F_t^{in}}\right)\frac{\delta F_j^{out}}{\delta F_i^{in}}\sigma_{\delta F_i^{in}}, \quad (2)$$

де для визначення $\sigma^2_{\delta F_i^{in}}$ ми використовували градієнти значень функції витрат відносно правильно означених даних.

Враховуючи вираз (2) та визначення енергії сигналу [15] визначимо енергію сигналу для кожного виходу моделі ШНМ за виразом:

$$E_j = \sum_{i=1}^{I} \left| D_{F_j^{out} | F_i^{in}} \right|. \tag{3}$$

Тоді значення інформативності по кожному входу ШНМ обчислюється за виразом:

$$GBI_i = \left(\sum_{j=1}^{J} \frac{\left| D_{F_j^{out} | F_i^{in}} \right|}{E_j} \right) / \left(\sum_{k=1}^{I} \sum_{j=1}^{J} \frac{\left| D_{F_j^{out} | F_k^{in}} \right|}{E_j} \right). \tag{4}$$

Значення інформативності, отримані за допомогою градієнтного методу, точніше відображають вплив кожного параметра з огляду на всі доступні розмічені дані. Однак цей метод не підходить для отримання значень поточної інформативності конкретних даних. Щоб усунути це обмеження, ми використовуємо метод інтегрованих градієнтів [11, 12, 13]. Ми адаптуємо цей метод відповідно до нашої моделі ШНМ.

Модифікація методу інтегрованих градієнтів має виявляти вплив вхідних параметрів на результати роботи моделі ШНМ для конкретного запису. Ця модифікація необхідна для інтерпретації рішень, прийнятих ШНМ, особливо в рамках комп'ютеризованих систем медичного моніторингу, де воно необхідне для обґрунтування експертних рішень. Фундаментальна концепція методу інтегрованих градієнтів передбачає інтегрування градієнтів даної функції втрат щодо вхідних даних, починаючи від деяких базових значень параметрів до конкретних значень вхідних параметрів за виразом [11; 13]:

$$IG(P) = \left(p_i - p_i^{'}\right) \int_{\alpha=0}^{1} \frac{\delta F\left(P' + \alpha\left(P - P'\right)\right)}{\delta p_i} d\alpha, \tag{5}$$

де $P = \{p_i\}$ – вектор вхідних параметрів, $P' = \{p_i^{'}\}$ – вектор базових значень параметрів, $F(P)$ – модель ШНМ.

Враховуючи вираз (5) та опис алгоритму IG в роботах [11; 12; 13], отримаємо модифікацію методу інтегрованих градієнтів для визначення інформативності:

1. Визначення базових значень параметрів та визначення кількості кроків інтеграції, зазвичай більше значення веде до точніших результатів.

2. Визначення точок інтеграції на лінії між базовими значеннями та поточними, їх кількість дорівнює кількості кроків інтеграції.

3. Обчислення градієнтів для кожної точки інтеграції, відштовхуючись від поточного значення та функції витрат. Та інтегрування отриманих градієнтів за допомогою обраного методу інтеграції.

4. Значення, отримані за допомогою методу IG, є ваговими коефіцієнтами, тому для перетворення на значення інформативності їх необхідно нормалізувати відповідно до наступного виразу:

$$IGI = \left|IG(P)\right| / sum\left(\left|IG(P)\right|\right). \tag{6}$$

Методологія дослідження полягає в використанні деякого набору даних, для перевірки спроможності розроблених методів оцінювати інформативність. Від оцінки варіативності PCA можна визначити кількість найбільш впливових параметрів (але через трансформацію PCA неможливо визначити які саме параметри), а порівнюючи розроблені методи один з одним можна перевірити їх точність. Також варто зазначити, що запропоновані методи мають бути чутливими до незбалансованості в даних.

У дослідженні ми використовували дані серцевих захворювань UCI [16]. Ці дані представляють типові дані пацієнтів у комп'ютерних системах медичного моніторингу. Оригінальний набір даних містить 920 записів із 76 параметрами, але ми зосередилися на 13 як найпопулярніших в інших дослідженнях [17]. Крім того, варто відмітити, що дані не були отримані з одного джерела; містить 33% записів з Клівлендської бази даних, 32% з Угорщини та 35% з інших джерел [16; 17]. Записи пацієнта в наборі даних позначені як здоровий або один із чотирьох типів захворювань серця. В наборі даних наявні наступні параметри [16]: *age* (вік), *origin* (джерело походження), *gender* (стать пацієнта), *cp* (тип грудного болю),

trestbps (тиск крові), *chol* (рівень холестеролу), *fbs* (індикатор норми рівня цукру), *restecg* (результати електрокардіографії в спокої), *thalach* (максимальний пульс), *exang* (наявність стенокардії), *oldpeak* (наявність пригнічення серцевого м'язу після навантаження), *slope* (оцінка швидкості відновлення після навантаження), *ca* (кількість набряклих судин), *thal* (тип дефекту серцевого м'язу), *class* (очікуваний діагноз: здоровий чи один з чотирьох типів захворювання).

Аналіз розподілу параметрів показує, що дані зміщені щодо найстарших пацієнтів (*age*) та пацієнтів чоловічої статі (*gender*). Далі ми провели аналіз принципових компонентів (PCA) з урахуванням цільового класу. Ми з'ясували (рис. 1), що необхідно 10 параметрів, для збереження 90% інформації. Це означає високу мінливість вхідних параметрів і ймовірно високу взаємозалежність між цільовим класом та параметрами. Також подальший аналіз даних показав, що 70% даних містять записи з відсутніми значеннями, які ми заповнили середніми значеннями. Усі ці факти вказують на те, що ми маємо справу зі складними даними, які стануть справжнім викликом для наших методів.

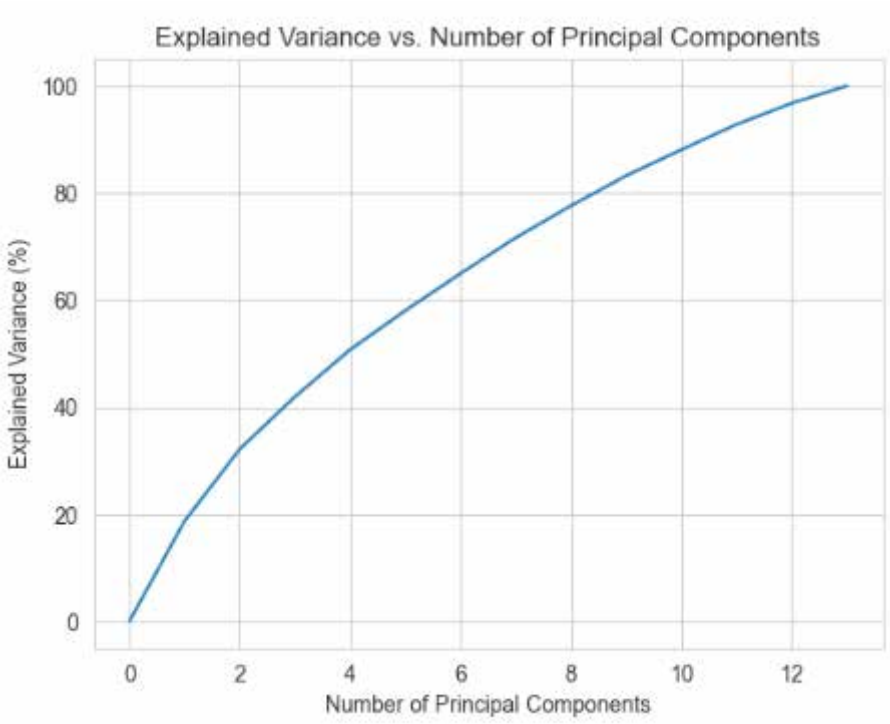

Рис. 1. Залежність варіативності від кількості принципових компонент, отримана за допомогою аналізу PCA

Для аналізу запропонованих методів нам необхідно навчити модель ШНМ на запропонованих даних, для цього випадковим чином розподіляємо набір даних UCI на набори для навчання та тестування зі співвідношенням 80% до 20%. Матриці плутанини показані на рис. 2.a для навчального набору та на рис. 2.b для тестового набору. З матриць плутанини ми можемо розрахувати точність на наборах для навчання та тестування: 93.61% та 82.6% відповідно.

Результати загальної інформативності параметрів наведено в таблиці 1, де показано 10 найбільш інформативних параметрів, що зберігають 83,13% загальної інформативності. Цей метод для набору даних UCI вирізним найбільш інформативні параметри, але такі параметри як *gender* і *age* є зміщеними, тому вони мають більш значний вплив на результати інформативності. Це свідчить, що метод спроможний визначати загальну інформативність та її значення співпадають з оцінкою PCA, проте надають ширшу інформацію по змінним.

Ми також розглянули використання модифікації методу інтегрованих градієнтів для визначення поточних значень інформативності. Типові результати використання цього методу наведені в таблиці 2. Можна вказати, які параметри мають найбільш значний вплив на рішення, що приймаються моделлю ШНМ. У цьому випадку параметри *oldpeak* і *ca*, можна визначити як найбільш впливові, незміщені прийняття рішення ШНМ. Також слід зазначити найбільш інформативні параметри загальної

інформативності виявилися найбільш інформативними в локальній інформативності, розрахованій модифікацією методу інтегрованих градієнтів, свідчить про те, що передбачувані методи спроможні точно обґрунтовано оцінити інформативність параметрів для навченої моделі ШНМ.

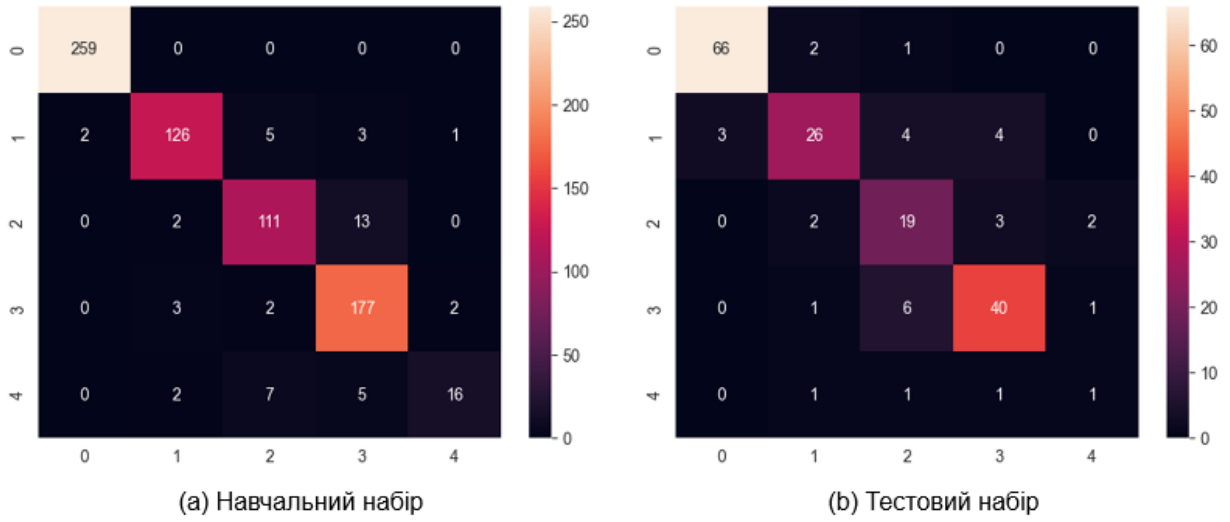

Рис. 2. Матриці плутанини для відображення точності навчання моделі ШНМ на даних UCI для: (a) навчального набору, (b) тестового набору

Таблиця 1

Результат роботи градієнтного методу для обчислення загальної інформативності, параметри відсортовані за інформативністю

Параметр	Навчальний	Тестовий	Загальний	Кумулятивний
gender	0.0947	0.1183	0.1036	0.1036
age	0.1115	0.0950	0.1053	0.2090
chol	0.0970	0.0972	0.0971	0.3061
ca	0.0807	0.0872	0.0831	0.3892
oldpeak	0.0860	0.0801	0.0838	0.4731
exang	0.0807	0.0772	0.0793	0.5525
fbs	0.0699	0.0778	0.0729	0.6254
thalch	0.0704	0.0726	0.0712	0.6967
restecg	0.0693	0.0651	0.0677	0.7644
thal	0.0645	0.0706	0.0668	0.8313

Таблиця 2

Значення інформативності вхідних параметрів конкретного запису, розрахованих модифікацією методу інтегрованих градієнтів

Параметр	**gender**	**oldpeak**	**age**	**ca**	**thal**
Інформативність	0.4959	0.1028	0.0735	0.0733	0.0726
	fbs	**trestbps**	**chol**	**thalch**	**cp**
	0.0453	0.0386	0.0314	0.0284	0.016

Висновки. Підсумовуючи, це дослідження стосується важливого аспекту комп'ютерних систем медичного моніторингу визначення інформативності даних пацієнтів для покращення та обґрунтування стратегій лікування. Реальним медичним даним, пов'язаним із проблемами медичного моніторингу, притаманна упередженість, відсутність значень та велика розмірність. Запропоновані методи виявили залежність з результатами аналізу PCA та також виявили взаємозв'язок результатів обох методів, не дивлячись на вище зазначені проблеми. Також виявлена в даних незбалансованість проявилась в результатах інформативності параметрів, що свідчить про спроможність запропонованих методів визначати точно інформативність. Вимірюючи інформативність параметрів, ми отримуємо знання про те,

як окремі параметри впливають на прогнози моделі ШНМ. Ці знання дають медикам змогу приймати більш обґрунтовані та детальніші рішення.

Хоча ця робота є значним кроком до покращення медичного моніторингу, необхідні подальші дослідження щодо тестування підсистеми загальної стратифікації в системі медичного моніторингу. Ми розглянемо перевірку запропонованих методів інформативності з більшими наборами даних і різноманітними клінічними сценаріями, що необхідно для оцінки ефективності запропонованих підходів. Крім того, ми протестуємо представлену систему в різних сценаріях використання, щоб оцінити загальну надійність і точність системи та порівняти її з продуктивністю експертів на даних тестування.

Список використаних джерел:

1. Logeshwaran, J., Malik, J. A., Adhikari, N., Joshi, S. S., Bishnoi, P. IoT-TPMS: An innovation development of triangular patient monitoring system using medical internet of things. *International Journal of Health Sciences*. 2022. 6(S5), 9070–9084. DOI: 10.53730/ijhs.v6nS5.10765.

2. Yu, M., Li, G., Jiang, D., Jiang, G., Tao, B. Chen, D. Hand medical monitoring system based on machine learning and optimal EMG feature set. *Personal and Ubiquitous Computing*. 2019. 1 – 17.

3. Humayun, M., Jhanjhi, N.Z., Almotilag, A., Almufareh, M.F. Agent-Based Medical Health Monitoring System. *Sensors*. Basel, Switzerland. 2022. *22*.

4. Miller, R., Acton, C., Fullerton, D.A., Maltby, J., Campling, J. Analysis of Variance (Anova). *The SAGE Encyclopedia of Research Design*. 2022.

5. Cozzi, M., Romano, S., Viccaro, M., Prete, C., Persiani, G. Wildlife Agriculture Interactions, Spatial Analysis and Trade-Off Between Environmental Sustainability and Risk of Economic Damage. 2015.

6. Chung, H., Ko, H., Kang, W.S., Kim, K.W., Lee, H., Park, C., Song, H., Choi, T., Seo, J.H., Lee, J. Prediction and Feature Importance Analysis for Severity of COVID-19 in South Korea Using Artificial Intelligence: Model Development and Validation. *Journal of Medical Internet Research*. 2021. 23.

7. Pereira, J.P., Stroes, E.S., Zwinderman, A.H., Levin, E. Covered Information Disentanglement: Model Transparency via Unbiased Permutation Importance. *ArXiv, abs/2111.09744*. 2021.

8. Ekanayake, I.U., Meddage, D.P., Rathnayake, U.S. A novel approach to explain the black-box nature of machine learning in compressive strength predictions of concrete using Shapley additive explanations (SHAP). *Case Studies in Construction Materials*. 2022.

9. Wu, Y., Zhou, Y. Hybrid machine learning model and Shapley additive explanations for compressive strength of sustainable concrete. *Construction and Building Materials*. 2022.

10. Lundstrom, D., Huang, T., Razaviyayn, M. A Rigorous Study of Integrated Gradients Method and Extensions to Internal Neuron Attributions. *ArXiv, abs/2202.11912*. 2022.

11. Enguehard, J. Sequential Integrated Gradients: a simple but effective method for explaining language models. *ArXiv, abs/2305.15853*. 2023.

12. Qi, Z., Khorram, S., Li, F. Visualizing Deep Networks by Optimizing with Integrated Gradients. *CVPR Workshops*. 2019.

13. Kovacs, I., Iosub, A., Topa, M.D., Buzo, A., Pelz, G. A Gradient-based Sensitivity Analysis Method for Complex Systems. *2019 IEEE 25th International Symposium for Design and Technology in Electronic Packaging (SIITME)*, 333-338. 2019.

14. Strilets, V., Bakumenko, N., Chernysh, S., Ugryumov, M., Donets, V. Application of Artificial Neural Networks in the Problems of the Patient's Condition Diagnosis in Medical Monitoring Systems. *Integrated Computer Technologies in Mechanical Engineering*. 2020.

15. В. Є. Стрілець, С. І. Шматков, М. Л. Угрюмов. *Методи машинного навчання у задачах системного аналізу і прийняття рішень* : монографія / Харків : Харківський національний університет імені В. Н. Каразіна. 2020.

16. Janosi Andras, Steinbrunn William, Pfisterer Matthias, Detrano Robert. Heart Disease. UCI Machine Learning Repository. 1988.

17. Pereira, N. Using Machine Learning Classification Methods to Detect the Presence of Heart Disease. 2019.

References:

1. Logeshwaran, J., Malik, J. A., Adhikari, N., Joshi, S. S., & Bishnoi, P. (2022). IoT-TPMS: An innovation development of triangular patient monitoring system using medical internet of things. *International Journal of Health Sciences, 6(S5)*, 9070–9084. DOI: 10.53730/ijhs.v6nS5.10765.

2. Yu, M., Li, G., Jiang, D., Jiang, G., Tao, B., & Chen, D. (2019). Hand medical monitoring system based on machine learning and optimal EMG feature set. *Personal and Ubiquitous Computing*, 1 – 17.

3. Humayun, M., Jhanjhi, N.Z., Almotilag, A., & Almufareh, M.F. (2022). Agent-Based Medical Health Monitoring System. *Sensors (Basel, Switzerland)*, *22*.

4. Miller, R., Acton, C., Fullerton, D.A., Maltby, J., & Campling, J. (2022). Analysis of Variance (Anova). *The SAGE Encyclopedia of Research Design*.

5. Cozzi, M., Romano, S., Viccaro, M., Prete, C., & Persiani, G. (2015). Wildlife Agriculture Interactions, Spatial Analysis and Trade-Off Between Environmental Sustainability and Risk of Economic Damage.

6. Chung, H., Ko, H., Kang, W.S., Kim, K.W., Lee, H., Park, C., Song, H., Choi, T., Seo, J.H., & Lee, J. (2021). Prediction and Feature Importance Analysis for Severity of COVID-19 in South Korea Using Artificial Intelligence: Model Development and Validation. *Journal of Medical Internet Research, 23*.

7. Pereira, J.P., Stroes, E.S., Zwinderman, A.H., & Levin, E. (2021). Covered Information Disentanglement: Model Transparency via Unbiased Permutation Importance. *ArXiv, abs/2111.09744*.

8. Ekanayake, I.U., Meddage, D.P., & Rathnayake, U.S. (2022). A novel approach to explain the black-box nature of machine learning in compressive strength predictions of concrete using Shapley additive explanations (SHAP). *Case Studies in Construction Materials*.

9. Wu, Y., & Zhou, Y. (2022). Hybrid machine learning model and Shapley additive explanations for compressive strength of sustainable concrete. *Construction and Building Materials*.

10. Lundstrom, D., Huang, T., & Razaviyayn, M. (2022). A Rigorous Study of Integrated Gradients Method and Extensions to Internal Neuron Attributions. *ArXiv, abs/2202.11912*.

11. Enguehard, J. (2023). Sequential Integrated Gradients: a simple but effective method for explaining language models. *ArXiv, abs/2305.15853*.

12. Qi, Z., Khorram, S., & Li, F. (2019). Visualizing Deep Networks by Optimizing with Integrated Gradients. *CVPR Workshops*.

13. Kovacs, I., Iosub, A., Topa, M.D., Buzo, A., & Pelz, G. (2019). A Gradient-based Sensitivity Analysis Method for Complex Systems. *2019 IEEE 25th International Symposium for Design and Technology in Electronic Packaging (SIITME)*, 333-338.

14. Strilets, V., Bakumenko, N., Chernysh, S., Ugryumov, M., & Donets, V. (2020). Application of Artificial Neural Networks in the Problems of the Patient's Condition Diagnosis in Medical Monitoring Systems. *Integrated Computer Technologies in Mechanical Engineering*.

15. Strilets V.E., Shmatkov S.I. & Ugryumov M.L. (2020). *Metody mashynnoho navchannia u zadachakh systemnoho analizu i pryiniattia rishen [Methods of machine learning in the problems of system analysis and decision making: monograph]*. Karazin Kharkiv National University.

16. Janosi Andras, Steinbrunn William, Pfisterer Matthias, & Detrano Robert. (1988). Heart Disease. UCI Machine Learning Repository.

17. Pereira, N. (2019). Using Machine Learning Classification Methods to Detect the Presence of Heart Disease.

УДК 004.056.5
DOI https://doi.org/10.32689/maup.it.2023.5.2

Вадим КАЛЬЧЕНКО
старший викладач кафедри кібербезпеки, Сумський державний університет, вул. Харківська, 116, Суми, Україна, індекс 40007; підполковник, Управління Державної служби спеціального зв'язку та захисту інформації України в Сумській області, вул. Герасима Кондратьєва, 32/1, Суми, Україна, індекс 40000 (v.kalchenko@ctois.sumdu.edu.ua)

ORCID: 0000-0001-6492-3806

Віктор ОБОДЯК
кандидат технічних наук, доцент, доцент кафедри кібербезпеки, Сумський державний університет, вул. Харківська, 116, Суми, Україна, індекс 40007; студент магістратури, Харківський національний університет радіоелектроніки, просп. Науки, 14, Харків, Україна, індекс 61166 (v.obodyak@cs.sumdu.edu.ua)

ORCID: 0000-0002-8539-1252

Vadym KALCHENKO
Senior Lecturer at the Cybersecurity Department, Sumy State University, 116, Kharkivska St, Sumy, Ukraine, postal code 40007; Lieutenant Colonel, Office of the State Service of Special Communications and Information Protection of Ukraine in Sumy Oblast, 32/1, Gerasyma Kondratieva St, Sumy, Ukraine, postal code 40000 (v.kalchenko@ctois.sumdu.edu.ua)

Viktor OBODIAK
Candidate of Technical Science, Associate Professor, Associate Professor at the Cybersecurity Department, Sumy State University, 116, Kharkivska St, Sumy, Ukraine, postal code 40007; Master's Student, Kharkiv National University of Radio Electronics, 14, Nauky Ave, Kharkiv, Ukraine, postal code 61166 (v.obodyak@cs.sumdu.edu.ua)

Бібліографічний опис статті: Кальченко, В., Ободяк, В. (2023). Порівняльна характеристика нормативних вимог України та ЄС у сфері кіберзахисту персональних даних в інформаційно-комунікаційних системах. *Інформаційні технології та суспільство*, 5 (11), 14–20. DOI: https://doi.org/10.32689/maup.it.2023.5.2

Bibliographic description of the article: Kalchenko, V., Obodiak, V. (2023). Porivnialna kharakterystyka normatyvnykh vymoh Ukrainy ta YeS u sferi kiberzakhystu personalnykh danykh v informatsiino-komunikatsiinykh systemakh [Comparative characteristics of regulatory requirements of Ukraine and the EU in the field of personal data cyber protection in information and communication systems]. *Informatsiini tekhnolohii ta suspilstvo – Information technology and society,* 5 (11), 14–20. DOI: https://doi.org/10.32689/maup.it.2023.5.2

ПОРІВНЯЛЬНА ХАРАКТЕРИСТИКА НОРМАТИВНИХ ВИМОГ УКРАЇНИ ТА ЄС У СФЕРІ КІБЕРЗАХИСТУ ПЕРСОНАЛЬНИХ ДАНИХ В ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ СИСТЕМАХ[1]

Анотація. *В статті розглянуто актуальне питання щодо застосування нормативних вимог в сфері кіберзахисту для збереження персональних даних, які обробляються в інформаційно-комунікаційних системах, адже роль глобальної мережі Інтернет стає все важливішою в житті людей незалежно від того в якій країні вони знаходяться. Можна зазначити, що проблема захисту персональних даних в жодній країні не вирішена повністю, але для досягнення більшої гарантії збереження персональних даних в своїй державі, потрібно спиратися на кращі наробки в цьому напрямку. Значних успіхів в захисті персональних даних досягли в Європейському Союзі. Відповідно, необхідно проведення досліджень, спрямованих на вивчення нормативних документів Європейського Союзу, завдяки яким досягнуті ці успіхи. Для цього, в першу чергу, потрібно порівняти нормативні вимоги України і Європейського Союзу в сфері захисту персональних даних. Особливу увагу потрібно звернути саме на кіберзахист, а не просто захист персональних даних. Адже саме в інформаційно-комунікаційних системах і відбувається основна втрата персональних даних в наш час. Результати досліджень, наведених в даній статті, показують, що нормативні документи України і основний нормативний документ Європейського Союзу, а саме «Загальний регламент про захист даних» (General Data Protection Regulation, GDPR), що регламентують захист персональних даних, не містять чітких і явних вимог щодо кіберзахисту персональних даних. Але вимоги GDPR, щодо захисту персональних даних більш ширші порівняно*

[1] Публікація підготовлена у рамках проєкту Модуль Жана Моне «Досвід ЄС щодо захисту персональних даних у кіберпросторі» (2023-2026 – EUEPPDC – 101125350 – ERASMUS-JMO-2023-MODULE)

з Законом України «Про захист персональних даних». Тому можливість впровадження норм GDPR в нормативні документи України по захисту персональних даних є актуальною. Таким чином, результатом подальших досліджень має стати вироблення конкретних рекомендацій по вдосконаленню нормативних документів в Україні, спрямованих на регулювання захисту персональних даних, які базуються на нормах GDPR. Також актуальним є дослідження з розробки концепції вимог до інформаційно-комунікаційних систем з точки зору кіберзахисту для збереження персональних даних.

Ключові слова: *кіберзахист, персональні дані, захист інформації, нормативні вимоги, GDPR.*

COMPARATIVE CHARACTERISTICS OF THE REGULATORY REQUIREMENTS OF UKRAINE AND THE EU IN THE FIELD OF PERSONAL DATA CYBER PROTECTION IN INFORMATION AND COMMUNICATION SYSTEMS[2]

Abstract. *The article discusses the timely issue of applying regulatory requirements in the field of cybersecurity for the preservation of personal data processed in information and communication systems. This is due to the increasing significance of the global Internet network in people's lives, regardless of their location. It is noteworthy that the challenge of personal data protection remains unresolved in any country. However, to ensure a higher guarantee of personal data preservation within one's country, it is crucial to rely on the best practices in this domain. Significant strides have been made in personal data protection within the European Union, prompting the need for research aimed at understanding the regulatory documents that led to these successes. To achieve this, it is imperative to compare the regulatory requirements of Ukraine and the European Union in the realm of personal data protection, giving particular attention to cybersecurity, not just personal data protection. Information and communication systems are the focal point for the primary loss of personal data in contemporary times. The research presented in this article reveals that the regulatory documents of Ukraine and the principal regulatory document of the European Union, the General Data Protection Regulation (GDPR), which governs the protection of personal data, lack clear and explicit requirements for the cybersecurity of personal data. However, GDPR requirements for personal data protection are more extensive compared to Ukraine's Law "On Personal Data Protection". Consequently, the relevance of incorporating GDPR norms into Ukraine's regulatory documents on personal data protection is evident. Thus, the result of further research should be the development of specific recommendations for improving regulatory documents in Ukraine aimed at regulating the protection of personal data, based on GDPR regulations. Research on developing the concept of requirements for information and communication systems from the perspective of cybersecurity for preserving personal data is also pertinent.*

Key words: *cybersecurity, personal data, information protection, regulatory requirements, GDPR.*

Актуальність теми дослідження і постановка проблеми. Останні десятиліття характеризуються значним зростання ролі Інтернету в суспільному житті. Глобальна мережа дозволяє вести бізнес, спілкуватись в режимі онлайн між собою незважаючи на відстані, проводити фінансові операції, надавати консультації, вчитися тощо. Внаслідок цього виникають питання кіберзахисту як цифрових активів підприємств так і персональних даних громадян, які обробляються в інформаційно-комунікаційних системах. Варто констатувати, що розвиток суспільних взаємовідносин в мережі Інтернет та розвиток комп'ютерних технологій в цілому відбувається набагато швидше ніж розвиток законодавства. Внаслідок цього виникають питання правового регулювання таких взаємовідносин. Особливо це стосується кіберзахисту персональних даних в інформаційно-комунікаційних системах.

Аналіз останніх досліджень і публікацій. Захисту персональних даних присвячено надзвичайно багато досліджень. Вони можуть бути спрямовані, наприклад, на аналіз викликів для конфіденційності користувачів і захисту персональних даних інтернету речей [1]. В іншій статті [2] розглядається необхідність регулювання конфіденційності даних і штучного інтелекту через Загальний регламент захисту даних (GDPR) [3]. Є, наприклад, стаття [4] в якій розглядаються проблеми особистої інформаційної безпеки та вдосконалення кримінальної відповідальності з точки зору інформаційної безпеки громадян.

Виділення недосліджених частин загальної проблем. Основна увага дослідників звертається на проблему захисту персональних даних як таких, а не на те, що потрібно зробити в інформаційно-комунікаційних системах для захисту цих даних.

Формулювання мети статті. Метою даної статті є аналіз та порівняння наявних нормативних вимог у сфері кіберзахисту персональних даних в Україні та Європейському союзі. Також стаття має на меті виробити рекомендації щодо покращення нормативного регулювання кіберзахисту персональних даних, що обробляються в інформаційно-комунікаційних системах українських органах державної влади, органах місцевого самоврядування, підприємствах, установах та організаціях незалежно від їх форми власності.

Виклад основного матеріалу. Вперше основні принципи обробки персональних даних, права осіб персональні дані яких обробляються, та норми щодо транскордонної передачі даних були викладені в Конвенції Ради Європи № 108 «Про захист осіб у зв'язку з автоматизованою обробкою персональних

[2]The publication is prepared within the Jean Monnet Module project "EU Experience in Personal Data Protection in Cyberspace" (2023-2026 – EUEPPDC – 101125350 – ERASMUS-JMO-2023-MODULE)

даних», ухваленій ще 28 січня 1981 р. [5]. З метою впровадження положень європейського нормативного акту до законодавства України, Верховна Рада України 1 червня 2010 р. ухвалила Закон «Про захист персональних даних» [6]. В ньому було закріплено принципи обробки персональних даних, права суб'єктів персональних даних, основні підстави для обробки персональних даних, підстави для обробки чутливих категорій персональних даних, положення щодо обмеження дії Закону, повноваження наглядового органу, тощо. Проте даний Закон не визначає технічних вимог до комп'ютерних систем, в яких ці персональні дані обробляються. Цей Закон фактично повністю базувався на положеннях Директиви 95/46/ЄС Європейського парламенту і Ради «Про захист фізичних осіб у зв'язку з обробкою персональних даних і вільне переміщення таких даних» від 24 жовтня 1995 року [7], яка наразі скасована і замінена Загальним регламентом із захисту персональних даних (General Data Protection Regulation – GDPR) [3].

Зазначений документ обов'язковий до виконання всіма суб'єктами господарювання, які знаходяться на території ЄС. Проте його особливістю є те, що даному регламенту повинен відповідати і будь-який інший суб'єкт господарювання, який обробляє персональні дані громадян ЄС, навіть якщо він знаходиться поза його межами (в іншій країні). Наразі варто констатувати, що існуюче законодавство України в сфері захисту персональних даних застаріло і не відповідає сучасним вимогам та рівню технологічного розвитку.

Відповідно до пункту 11 Плану заходів щодо імплементації Угоди про асоціацію між Україною та ЄС [8] Україна взяла зобов'язання щодо вдосконалення законодавства про захист персональних даних та приведення його у відповідність до GDPR. До Верховної Ради України 7 червня 2021 р. було подано проєкт Закону України «Про захист персональних даних», проте даний законопроєкт не було прийнято [9].

У відповідності до статті 8 Закону України «Про захист інформації в інформаційно-комунікаційних системах» [10] інформація, вимога щодо захисту якої встановлена законом, повинна оброблятися в системі із застосуванням комплексної системи захисту інформації (КСЗІ) з підтвердженою відповідністю. Підтвердження відповідності здійснюється за результатами державної експертизи в порядку, встановленому законодавством. Також варто зазначити, що в нашій країні існує низка нормативних документів системи технічного захисту інформації (НД ТЗІ), в яких висуваються вимоги як до самої комплексної системи захисту інформації, так і до порядку складання, наповнення та оформлення документів на таку систему. Аналізуючи увесь перелік українських нормативних документів та вимог, які в них містяться – комплексну систему захисту інформації можна представити у вигляді двох великих складових. Перша складова – це безпосередньо програмні, апаратні та програмно-апаратні компоненти, які дозволяють технічними засобами досягти захищеності інформації (забезпечення конфіденційності, цілісності та доступності інформації). Друга складова – це перелік нормативних, розпорядчих, експлуатаційних документів, що регламентують порядок обробки інформації, що захищається та порядок роботи на комп'ютерній системі зі створеною КСЗІ. Варто констатувати, що на даний момент не існує НД ТЗІ або інших нормативних документів, в яких чітко висуваються технічні вимоги щодо захисту персональних даних. Наразі в Україні прийнято Загальні вимоги до кіберзахисту об'єктів критичної інфраструктури [11], в яких містяться вимоги щодо кіберзахисту. Проте даний нормативний документ регламентує тільки порядок кіберзахисту об'єктів критичної інформаційної інфраструктури України, що внесені до відповідного переліку.

Як показує аналіз законодавства Європейського Союзу в сфері захисту інформації [12], можна виділити такі нормативні документи, дія яких розповсюджується на всі країни-учасники ЕС: General Data Protection Regulation (GDPR) [3], Payment Services Directive (PSD2) [13], The eIDAS Regulation [14], NIS2 Directive [15].

General Data Protection Regulation (GDPR) – це основний регуляторний документ ЄС, який встановлює правила обробки персональних даних. Він надає всім зацікавленим сторонам більше контролю над їхніми даними та вимагає від організацій дотримуватися строгих процедур захисту даних.

Payment Services Directive (PSD2) – це директива ЄС, яка регулює платіжні послуги та платіжних посередників з метою збільшення конкуренції та підвищення безпеки платежів у ЄС.

The eIDAS Regulation – це регламент встановлює правила для електронних ідентифікацій та довірчих послуг для електронних транзакцій в ЄС.

NIS2 Directive: NIS2 – це директива ЄС, яка спрямована на забезпечення вищого рівня кібербезпеки в усіх країнах-членах, розширюючи обов'язки та вимоги до різних секторів та послуг.

Як зазначалось вище, основним нормативним документом ЕС в сфері захисту персональних даних є General Data Protection Regulation (GDPR). Даний документ регламентує порядок обробки персональних даних та встановлює вимоги до систем захисту в тих інформаційно-комунікаційних

системах, в яких будуть оброблятись персональні дані. Проте даний документ більше фокусується на порядку та процедурах обробки персональних даних, питання кіберзахисту в чистому вигляді не розглядаються. Але можна виділити деякі його положення, які так чи інакше стосуються даного питання, а саме:

1. Необхідність забезпечення конфіденційності, цілісності та доступності інформації, що обробляється в системі (стаття 32).
2. Необхідність проведення тестування і оцінки тих заходів, які були вжиті для захисту персональних даних (стаття 32).
3. Використання шифрування (стаття 32).
4. Контроль доступу до персональних даних (стаття 32).
5. Здатність відновлювати доступність персональних даних у випадку виникнення фізичного або технічного інциденту (стаття 32).
6. Впровадження «псевдонімізації» (стаття 32).
7. Необхідність впровадження концепції “privacy by design” та “privacy by default” (стаття 25).
8. Необхідність мінімізації терміну зберігання даних (стаття 5).

Зазначені положення зазвичай є стандартними для будь-яких систем кіберзахисту, окрім «псевдонімізації» та концепції “privacy by design” та “privacy by default”. В GDPR «псевдонімізація», або «використання псевдонімів» – це опрацювання персональних даних у спосіб, який не дозволяє віднести персональні дані до конкретного суб'єкта без використання додаткової інформації.

Концепція “privacy by design” означає, що такі поняття, як «конфіденційність» та «безпека», повинні впроваджуватись в товари і послуги на самих ранніх етапах їх розробки та протягом всього життєвого циклу продукту. В свою чергу “privacy by default” означає, що в усіх продуктах і послугах повинні бути встановлені найвищі налаштування конфіденційності. Наприклад, якщо якась послуга вимагає реєстрації чи збирання даних, за замовчуванням повинно збиратися мінімально можлива кількість даних, необхідних для надання послуги [16]. Той факт, що необхідність застосування даних концепцій надається в нормативному документі, вимагає від установ та організацій бути більш відповідальними в виборі підходів до обробки персональних даних.

Нормативні документи України [6, 17] не містять чітких і явних вимог щодо кіберзахисту персональних даних. Аналізуючи ці нормативні документи, можна виділити наступні вимоги для побудови системи кіберзахисту на підприємстві:

1. Необхідність забезпечення конфіденційності, цілісності та доступності персональних даних (стаття 24 [6], пункти 3.3, 3.13, 3.14 [17]).
2. Необхідність забезпечення резервного копіювання персональних даних (стаття 24 [6], пункт 3.3 [17]).

При цьому вищезазначені вимоги в нормативних документах явно не прописані, а виходять з тексту статей.

Пунктом 3.2 [17] вимагається у розпорядника персональних даних самостійно визначити перелік і склад заходів, спрямованих на безпеку обробки персональних даних з урахуванням вимог законодавства в сфері інформаційної безпеки. Таким чином, в Україні кіберзахист персональних даних вимагається іншими нормативними актами.

Так, згідно зі ст. 11 Закону України «Про інформацію» [18], до конфіденційної інформації про фізичну особу належать, зокрема, дані про її національність, освіту, сімейний стан, релігійні переконання, стан здоров'я, а також адреса, дата і місце народження. Враховуючи даний факт, чітка вимога щодо захисту такого роду інформації міститься в Законі України «Про захист інформації в інформаційно-комунікаційних системах» [10] та в «Правилах забезпечення захисту інформації в інформаційних, електронних комунікаційних та інформаційно-комунікаційних системах [19].

В цих документах можна зазначити наявність вимог з кіберзахисту:

1. Забезпечення конфіденційності, цілісності та доступності інформації (п.п. 5, 6 [19]).
2. Забезпечення криптографічного захисту інформації (шифрування) конфіденційної інформації в органах державної влади, органах місцевого самоврядування, на підприємствах, в установах та організаціях, які належать до сфери їх управління, військових формуваннях, які створені відповідно до закону (пункти 9, 13 [19]).
3. Забезпечення аудиту подій (пункт 11 [19]).
4. Розмежування повноважень привілейованого та звичайного користувача (пункт 11 [19]).
5. Забезпечення ідентифікації та автентифікації користувачів (пункт 13 [19]).
6. Забезпечення цілісності засобів захисту інформації в системі (пункт 15 [19]).
7. Забезпечення антивірусного захисту(пункт 16 [19]).

Можна зробити висновок, що в Україні не існує нормативного документу, який би чітко формулював вимоги до кіберзахисту інформаційно-комунікаційних систем, в яких обробляються персональні дані. Внаслідок цього виникає проблема реалізації дієвого захисту такого роду чуттєвої інформації.

Висновки та перспективи подальших досліджень. Нормативні документи України і основний нормативний документ Європейського Союзу, а саме «Загальний регламент про захист даних» (General Data Protection Regulation, GDPR), що регламентують захист персональних даних, не містять чітких і явних вимог щодо кіберзахисту персональних даних. Але вимоги GDPR щодо захисту персональних даних більш ширші порівняно з Законом України «Про захист персональних даних». Тому можливість впровадження норм GDPR в нормативні документи України по захисту персональних даних є актуальною. Таким чином, результатом подальших досліджень має стати вироблення конкретних рекомендацій по вдосконаленню нормативних документів в Україні, спрямованих на регулювання захисту персональних даних, які базуються на нормах GDPR. Також актуальним є дослідження з розробки концепції вимог до інформаційно-комунікаційних систем з точки зору кіберзахисту для збереження персональних даних.

Список використаних джерел:

1. Romansky, Radi. 2023. Internet of Things and User Privacy Protection. 37th International Conference on Information Technologies, InfoTech 2023 – Proceedings. URL: http://infotech-bg.com/proceedings.

2. Brown, R., Truby J., Imad Antoine Ibrahim. Mending Lacunas in the EU's GDPR and Proposed Artificial Intelligence Regulation. European Studies. Volume 9 (2022): Issue 1. (August 2022). URL: https://sciendo.com/issue/EUSTU/9/1/.

3. Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation). URL: https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32016R0679&from=EN.

4. Yu Zhang, Haoyun Dong. Criminal law regulation of cyber fraud crimes–from the perspective of citizens' personal information protection in the era of edge computing. Journal of Cloud Computing volume 12, Article number: 64 (2023). URL: https://journalofcloudcomputing.springeropen.com/articles/10.1186/s13677-023-00437-3#citeas.

5. Convention for the Protection of Individuals with regard to Automatic Processing of Personal Data. Amendment to Convention ETS. No. 108 allowing the European Communities to accede. URL: https://rm.coe.int/1680078b37.

6. Закон України «Про захист персональних даних». URL: https://zakon.rada.gov.ua/laws/show/2297-17#Text.

7. Директива 95/46/ЄС Європейського парламенту і Ради «Про захист фізичних осіб у зв'язку з обробкою персональних даних і вільне переміщення таких даних» від 24 жовтня 1995 року. URL: https://zakon.rada.gov.ua/laws/show/994_242#Text.

8. План заходів з виконання Угоди про асоціацію між Україною, з однієї сторони, та Європейським Союзом, Європейським та Європейським Союзом, Європейським співтовариством з атомної енергії і їхніми державами-членами, з іншої сторони. Затверджено постановою Кабінету Міністрів України від 25 жовтня 2017 р. № 1106. URL: https://zakon.rada.gov.ua/laws/show/1106-2017-%D0%BF#Text.

9. Проект Закону України «Про захист персональних даних». URL: https://w1.c1.rada.gov.ua/pls/zweb2/webproc4_1?pf3511=72160.

10. Закон України «Про захист інформації в інформаційно-комунікаційних системах». URL: https://zakon.rada.gov.ua/laws/show/80/94-%D0%B2%D1%80#Text.

11. Загальні вимоги до кіберзахисту об'єктів критичної інфраструктур. Затверджено постановою Кабінету Міністрів України від 29 червня 2019 р. № 518. URL: https://zakon.rada.gov.ua/laws/show/373-2006-%D0%BF#Text.

12. [Лит. Статья по законодательству ЕС] Mantelero, A., Vaciago, G., Esposito, M. S., Monte N. The common EU approach to personal data and cybersecurity regulation. *International Journal of Law and Information Technology, 2020, 28, 297–328 doi: 10.1093/ijlit/eaaa021*. URL: https://academic.oup.com/ijlit/article/28/4/297/6120059?login=false.

13. Directive (EU) 2015/2366 of the European Parliament and of the Council of 25 November 2015 on payment services in the internal market, amending Directives 2002/65/EC, 2009/110/EC and 2013/36/EU and Regulation (EU) No 1093/2010, and repealing Directive 2007/64/EC (Text with EEA relevance). URL: https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=celex%3A32015L2366.

14. Regulation (EU) No 910/2014 of the European Parliament and of the Council of 23 July 2014 on electronic identification and trust services for electronic transactions in the internal market and repealing Directive 1999/93/EC. URL: https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=uriserv%3AOJ.L_.2014.257.01.0073.01.ENG.

15. Directive (EU) 2022/2555 of the European Parliament and of the Council of 14 December 2022 on measures for a high common level of cybersecurity across the Union, amending Regulation (EU) No 910/2014 and Directive (EU) 2018/1972, and repealing Directive (EU) 2016/1148 (NIS 2 Directive) (Text with EEA relevance). URL: https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX%3A32022L2555&qid=1704742816799.

16. Guidelines 4/2019 on Article 25 Data Protection by Design and by Default. Version 2.0. Adopted on 20 October 2020. URL: https://edpb.europa.eu/our-work-tools/our-documents/guidelines/guidelines-42019-article-25-data-protection-design-and_en.

17. Типовий порядок обробки персональних даних. Затверджено наказом Уповноваженого Верховної Ради України з прав людини 08.01.2014 № 1/02-14. URL: https://zakon.rada.gov.ua/laws/show/v1_02715-14#Text.

18. Закон України «Про інформацію». URL: https://zakon.rada.gov.ua/laws/show/2657-12#Text.

19. Правила забезпечення захисту інформації в інформаційних, електронних комунікаційних та інформаційно-комунікаційних системах. Затверджено постановою Кабінету Міністрів України від 29 березня 2006 р. **№** 373. URL: https://zakon.rada.gov.ua/laws/show/373-2006-%D0%BF#Text.

References:

1. Romansky, R. (2023). Internet of Things and User Privacy Protection. *37th International Conference on Information Technologies, InfoTech 2023 – Proceedings. http://infotech-bg.com.* Retrieved from http://infotech-bg.com/proceedings.

2. Brown, R., Truby J., Imad Antoine Ibrahim. Mending Lacunas in the EU's GDPR and Proposed Artificial Intelligence Regulation. *European Studies. Volume 9 (2022): Issue 1. (August 2022). https://sciendo.com.* Retrieved from https://sciendo.com/issue/EUSTU/9/1/.

3. Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation). (2016). *https://eur-lex.europa.eu.* Retrieved from https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32016R0679&from=EN.

4. Yu Zhang, Haoyun Dong. Criminal law regulation of cyber fraud crimes–from the perspective of citizens' personal information protection in the era of edge computing. *Journal of Cloud Computing volume 12, Article number: 64 (2023). https://journalofcloudcomputing.springeropen.com.* Retrieved from https://journalofcloudcomputing.springeropen.com/articles/10.1186/s13677-023-00437-3#citeas.

5. Convention for the Protection of Individuals with regard to Automatic Processing of Personal Data. Amendment to Convention ETS. No. 108 allowing the European Communities to accede. (1981). *https://rm.coe.int.* Retrieved from https://rm.coe.int/1680078b37.

6. Zakon Ukrainy "Pro zakhyst personalnykh danykh". [Law of Ukraine "On Personal Data Protection"]. (n.d.). *https://zakon.rada.gov.ua*. Retrieved from https://zakon.rada.gov.ua/laws/show/2297-17#Text [in Ukrainian].

7. Dyrektyva 95/46/IeS Yevropeiskoho parlamentu i Rady "Pro zakhyst fizychnykh osib u zv'iazku z obrobkoiu personalnykh danykh i vilne peremishchennia takykh danykh". [Directive 95/46/EU of the European Parliament and the Council "On the protection of natural persons in connection with the processing of personal data and the free movement of such data"]. (1995). *https://zakon.rada.gov.ua*. Retrieved from https://zakon.rada.gov.ua/laws/show/994_242#Text [in Ukrainian].

8. Plan zakhodiv z vykonannia Uhody pro asotsiatsiiu mizh Ukrainoiu, z odniiei storony, ta Yevropeiskym Soiuzom, Yevropeiskym ta Yevropeiskym Soiuzom, Yevropeiskym spivtovarystvom z atomnoi enerhii i yikhnimy derzhavamy-chlenamy, z inshoi storony. Zatverdzheno postanovoiu Kabinetu Ministriv Ukrainy vid 25 zhovtnia 2017 r. № 1106. [Action plan for the implementation of the Association Agreement between Ukraine, on the one hand, and the European Union, the European and European Union, the European Atomic Energy Community and their member states, on the other hand. Approved by the Resolution of the Cabinet of Ministers of Ukraine dated October 25, 2017 No. 1106]. (2017) *https://zakon.rada.gov.ua*. Retrieved from https://zakon.rada.gov.ua/laws/show/1106-2017-%D0%BF#Text [in Ukrainian].

9. Proekt Zakonu Ukrainy "Pro zakhyst personalnykh danykh". [The Draft Law of Ukraine "On Personal Data Protection"]. (n.d.). *https://w1.c1.rada.gov.ua.* Retrieved from https://w1.c1.rada.gov.ua/pls/zweb2/webproc4_1?pf3511=72160 [in Ukrainian].

10. Zakon Ukrainy "Pro zakhyst informatsii v informatsiino-komunikatsiinykh systemakh". [Law of Ukraine "On Protection of Information in Information and Communication Systems"]. *https://zakon.rada.gov.ua*. Retrieved from https://zakon.rada.gov.ua/laws/show/80/94-%D0%B2%D1%80#Text [in Ukrainian].

11. Zahalni vymohy do kiberzakhystu ob'iektiv krytychnoi infrastruktur. Zatverdzheno postanovoiu Kabinetu Ministriv Ukrainy vid 29 chervnia 2019 r. № 518. [General requirements for cyber protection of critical infrastructure facilities. Approved by the Resolution of the Cabinet of Ministers of Ukraine of June 29, 2019 No. 518]. (2019). *https://zakon.rada.gov.ua*. Retrieved from https://zakon.rada.gov.ua/laws/show/373-2006-%D0%BF#Text [in Ukrainian].

12. Mantelero, A., Vaciago, G., Esposito, M. S., Monte N. (2020) The common EU approach to personal data and cybersecurity regulation. *International Journal of Law and Information Technology, 2020, 28, 297–328 doi: 10.1093/ijlit/eaaa021*. Retrieved from https://academic.oup.com/ijlit/article/28/4/297/6120059?login=false.

13. Directive (EU) 2015/2366 of the European Parliament and of the Council of 25 November 2015 on payment services in the internal market, amending Directives 2002/65/EC, 2009/110/EC and 2013/36/EU and Regulation (EU) No 1093/2010, and repealing Directive 2007/64/EC (Text with EEA relevance). (2015). *https://eur-lex.europa.eu*. Retrieved from https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=celex%3A32015L2366.

14. Regulation (EU) No 910/2014 of the European Parliament and of the Council of 23 July 2014 on electronic identification and trust services for electronic transactions in the internal market and repealing Directive 1999/93/EC. (2014). *https://eur-lex.europa.eu.* Retrieved from https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=uriserv%3AOJ.L_.2014.257.01.0073.01.ENG.

15. Directive (EU) 2022/2555 of the European Parliament and of the Council of 14 December 2022 on measures for a high common level of cybersecurity across the Union, amending Regulation (EU) No 910/2014 and Directive (EU) 2018/1972, and repealing Directive (EU) 2016/1148 (NIS 2 Directive) (Text with EEA relevance). (2022). *https://eur-lex.europa.eu.* Retrieved from https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX%3A32022L2555&qid=1704742816799.

16. Guidelines 4/2019 on Article 25 Data Protection by Design and by Default. (2019). *https://edpb.europa.eu*. Retrieved from https://edpb.europa.eu/our-work-tools/our-documents/guidelines/guidelines-42019-article-25-data-protection-design-and_en.

17. Typovyi poriadok obrobky personalnykh danykh. Zatverdzheno nakazom Upovnovazhenoho Verkhovnoi Rady Ukrainy z prav liudyny 08.01.2014 № 1/02-14. [Typical procedure for processing personal data. Approved by the order of the Commissioner for Human Rights of the Verkhovna Rada of Ukraine on January 8, 2014 No. 1/02-14]. (2014). *https://zakon.rada.gov.ua*. Retrieved from https://zakon.rada.gov.ua/laws/show/v1_02715-14#Text [in Ukrainian].

18. Zakon Ukrainy "Pro informatsiiu". [Law of Ukraine "On Information"]. (n.d.). *https://zakon.rada.gov.ua*. Retrieved from https://zakon.rada.gov.ua/laws/show/2657-12#Text [in Ukrainian].

19. Pravyla zabezpechennia zakhystu informatsii v informatsiinykh, elektronnykh komunikatsiinykh ta informatsiino-komunikatsiinykh systemakh. Zatverdzheno postanovoiu Kabinetu Ministriv Ukrainy vid 29 bereznia 2006 r. № 373. [Rules for information protection in information, electronic communication and information and communication systems. Approved by the Resolution of the Cabinet of Ministers of Ukraine dated March 29, 2006 No. 373]. (2006). *https://zakon.rada.gov.ua*. Retrieved from https://zakon.rada.gov.ua/laws/show/373-2006-%D0%BF#Text [in Ukrainian].

УДК 004.72
DOI https://doi.org/10.32689/maup.it.2023.5.3

Олексій КЛИМЕНКО
аспірант кафедри комп'ютерних систем, мереж та кібербезпеки, факультету інформаційних технологій, Національного Університету Біоресурсів та Природокористування України, вул. Героїв Оборони, 16А, Київ, Україна, індекс 03041 (o.klymenko@nubip.edu.ua)

ORCID: 0009-0005-2590-1803

Oleksii KLYMENKO
Postgraduate Student at the Department of Computer Systems, Networks and Cybersecurity, Faculty of Information Technology, National University of Life and Environmental Sciences of Ukraine, 16A, Heroiv Oborony St, Kyiv, Ukraine, postal code 03041 (o.klymenko@nubip.edu.ua)

Бібліографічний опис статті: Клименко, О. (2023). Тенденції розвитку самовідновлювальних мереж. *Інформаційні технології та суспільство*, 5 (11), 21–27. DOI: https://doi.org/10.32689/maup.it.2023.5.3

Bibliographic description of the article: Klymenko, O. (2023). Tendentsii rozvytku samovidnovliuvalnykh merezh [Development tendencies of self-healing networks]. *Informatsiini tekhnolohii ta suspilstvo – Information technology and society*, 5 (11), 21–27. DOI: https://doi.org/10.32689/maup.it.2023.5.3

ТЕНДЕНЦІЇ РОЗВИТКУ САМОВІДНОВЛЮВАЛЬНИХ МЕРЕЖ

Анотація. *У статті розглянуто приклади самовідновлення працездатності мережі під час збоїв, описано етапи розвитку Self-Healing мереж, визначено основні принципи та технології, що відповідають концепції самовідновлення. Комп'ютерні мережі активно змінюються у масштабах та складності технологій. Проте, попри динамічний розвиток технологій концептуальний підхід у вирішенні проблем залишається сталим. Основна тенденція – разом з розвитком та автоматизацією мережі повинні з'являтися інструменти для її автоматичного відновлення, тобто самовідновлення (self-healing). В корні автоматизації лежить програмована складова – скрипти, мови програмування, технології, що дозволяють програмувати ті речі, які раніше не були програмованими. Для цього створюються протоколи, нові програми-прошарки, що здатні впливати на сталі процеси, нові технології. Можливе також використання пропрієтарних механізмів від розробників обладнання. Комп'ютерні мережі стають програмованими з єдиними центром керування. Щоб підтримувати систему моніторингу в актуальному стані та мати практичний зиск, потрібно розуміти тенденції розвитку мереж. Self-Healing мережа здатна здійснювати розширений моніторинг і виконувати коригувальні задачі, які зазвичай потребують втручання людини. Це дає можливість зменшити витрати на IT-персонал і більш стратегічно розподіляти людські ресурси, скорочуючи кількість годин, витрачених на реактивну роботу, та зосереджуватися більше на проактивній діяльності. Хоча концепція самовідновлення почалася понад 10 років тому, її актуальність набирає обертів на фоні активного переходу до SDN-мереж. Автоматизація моніторингу та підтримки працездатності мережі є невід'ємною складовою подальшого розвитку комп'ютерних мереж. Self-Healing мережа, що базується на різних інструментах програмування, позбавлена залежності від конкретного розробника обладнання та є більш універсальною.*

Ключові слова: *self-healing, моніторинг, автоматизація, комп'ютерна мережа, програмування, скрипт, резервування.*

DEVELOPMENT TENDENCIES OF SELF-HEALING NETWORKS

Abstract. *The article considers examples of self-healing network performance during failures, describes the stages of development of self-healing networks, defines the main principles and technologies corresponding to the concept of self-healing. Computer networks are actively changing in scale and complexity of technologies. However, despite the dynamic development of technologies, the conceptual approach to solving problems remains stable. The main trend is that along with the development and automation of the network, tools for its automatic recovery, i.e. self-healing, should appear. The root of automation is the programmable component – scripts, programming languages, technologies that allow programming those things that were not programmable before. For this, protocols, new layer programs capable of influencing stable processes, and new technologies are being created. It is also possible to use proprietary mechanisms from equipment developers. Computer networks are becoming programmable with a single control center. To keep the monitoring system up-to-date and to have a practical benefit, you need to understand the development trends of the networks. A self-healing network is capable of advanced monitoring and corrective tasks that would normally require human intervention. This enables to reduce IT staff costs and allocate human resources more strategically, reducing the number of hours spent on reactive work and focusing more on proactive activities. Although the concept of self-healing began more than 10 years ago, its relevance is gaining momentum against the background of the active transition to SDN networks. Automation of monitoring and maintenance of network performance is an integral component of the further development of computer networks. A self-healing network, which is based on various programming tools, is free from dependence on a specific hardware developer and is universal.*

Key words: *self-healing, monitoring, automation, computer network, programming, script, reservation.*

Вступ. У сучасних мережах завдання самовідновлення сервісів виходить на новий рівень. У великих сервіс-провайдерів практикується підхід, що сервіс, який перестав працювати, треба швидко вивести з експлуатації та підняти новий сервіс, замість того, щоб витрачати час на пошук причини. Це означає, що потрібно налаштувати системи моніторингу, які протягом секунд виявлять найменші відхилення від норми. Проте, звичних метрик замало, таких як завантаження інтерфейсу або доступності вузла. Недостатньо і ручного стеження чергового інженеру за ними. Для багатьох речей має бути Self-Healing – самовідновлення роботи у разі виникнення проблеми. Відслідковувати потрібно не лише окремі пристрої, а й здоров'я мережі. Це невід'ємна частина автоматизації.

Постановка проблеми. В сучасній розгорнутій динамічній мережі стає все важче, або майже неможливо, вчасно помічати проблеми та реагувати на них у ручному режимі. Процес автоматизації повинен стосуватися не лише Control Plane чи Data Plane, що присутнє у SDN-мережах, але й моніторингу.

Self-Healing (самовідновлювальна) мережа необхідна для мінімізації людських зусиль і витрат, пов'язаних із визначенням причин збою в складних системах. Потрібен час, щоб побачити мережі, які стануть достатньо інтелектуальними для контролю, ідентифікації та виправлення збоїв під час їх виявлення, але дослідження Self-Healing мереж тривають [1].

Аналіз досліджень і публікацій. Визначення Self-Healing мереж добре розкрите в дослідженні [15]. Цей термін означає здатність мережі відновлювати свою роботу незалежно та без зовнішнього втручання у разі будь-якого збою. Кожна система з властивостями самовідновлення має здатність виявляти, діагностувати та реагувати на збої. Основною метою інтеграції функцій самовідновлення в будь-яку роботу мережі є підвищення її надійності та зручності обслуговування. Ці атрибути якості традиційно підвищуються в системах самовідновлення [10; 11].

SDN-мережі останні роки досить активно розвиваються. Технологія постійно досліджується. Self-Healing мережі мають значно менше досліджень, хоча використовуються як у SDN, так і в класичних мережах.

Одне з таких досліджень розкрито у статті [14]. Інженери запропонували модель, в якій мережі Байєса використовуються для діагностування причини аварії, що сталася. Діагностичний блок використовує алгоритм байєсівських мереж, який включає спостереження, що надходять з повідомлень про помилки від NMS (Network Management System) і SM (Service Manage), і додає їх як докази для діагностики першопричини. Цей блок використовує інформацію про топологію мережі, надану контролером SDN, для побудови графа мережі Байєса. Також система отримує статус послуги від SM. Мережі Байєса – це модельний алгоритм, який моделює складні залежності мережі в умовах невизначеності. Алгоритм на основі байєсівської мережі вводиться у SDN, щоб дозволити цій архітектурі виявляти збої в площині додатків, площині керування та площині даних [14].

В задачах моніторингу та самовідновлення мереж також широко використовується протокол OpenFlow [3; 7; 12; 15]. Однак, OpenFlow доречно використовувати тільки в SDN-мережах. Крім того, попри проведені дослідження OpenFlow поки не став універсальним інструментом.

Метою статті є висвітлення тенденцій розвитку Self-Healing мереж, визначення основних принципів та технологій, що відповідають концепції самовідновлення.

Виклад основного матеріалу.

Моніторинг можна розділити на два основні типи:

1) Проактивний – дає можливість побудувати плани розвитку інфраструктури та оцінити результативність сценаріїв по внесенню змін до інфраструктури. Також є можливість прогнозувати та запобігти майбутнім системним недолікам та попередити несправності заздалегідь відносно зібраних історичних даних, та провести аналіз поведінки інфраструктури [13].

2) Реактивний моніторинг – спостереження за IT-інфраструктурою та іншими сервісами в режимі реального часу, можливість визначення невідповідності параметрів та вузьких місць роботи кожного компоненту відносно поточних показників [13].

При цьому поняття Self-Healing доцільне в кожному з визначних типів. Якщо проактивний моніторинг сигналізує про потенційне "слабке" місце в мережі, потрібно вжити автоматизовані превентивні заходи для запобігання аварії. Якщо реактивний моніторинг зафіксував аварію, Self-Healing мережа має автоматично побудувати альтернативні шляхи трафіку, в той час як інженер займається вирішенням проблеми, що сталася.

Управління складними мережами частіше за все дороге та вимагає великої кількості IT-персоналу. Крім того, час, потрібний для виявлення першопричин і вжиття заходів для виправлення проблем, може призвести до більшої втрати прибутку для компаній [2]. Самовідновлення призведе до скорочення або майже повного усунення процесу вирішення проблеми.

Етапи розвитку самовідновлювальних мереж. Одним з базових принципів самовідновлювальності роботи мережі є метод резервації. Якщо на border-маршрутизаторі перестає працювати основний Інтернет-канал, потрібно це виявити та переключити на резервний канал. Якщо зламався пристрій, потрібно пустити трафік альтернативними шляхами через інші пристрої. Кількість резервних одиниць (канал, пристрій тощо) залежить від потрібного рівня відмовостійкості.

Приклад 1. Один з поширених прикладів відмовостійкої мережі початку XXI сторіччя [4] складається з двох маршрутизаторів, які мають окреме підключення до ISP (Internet Service Provider) та мають між собою декілька з'єднань (рис. 1). У таблиці 1 наведено конфігурацію маршрутизаторів у класичному прикладі резервації каналів.

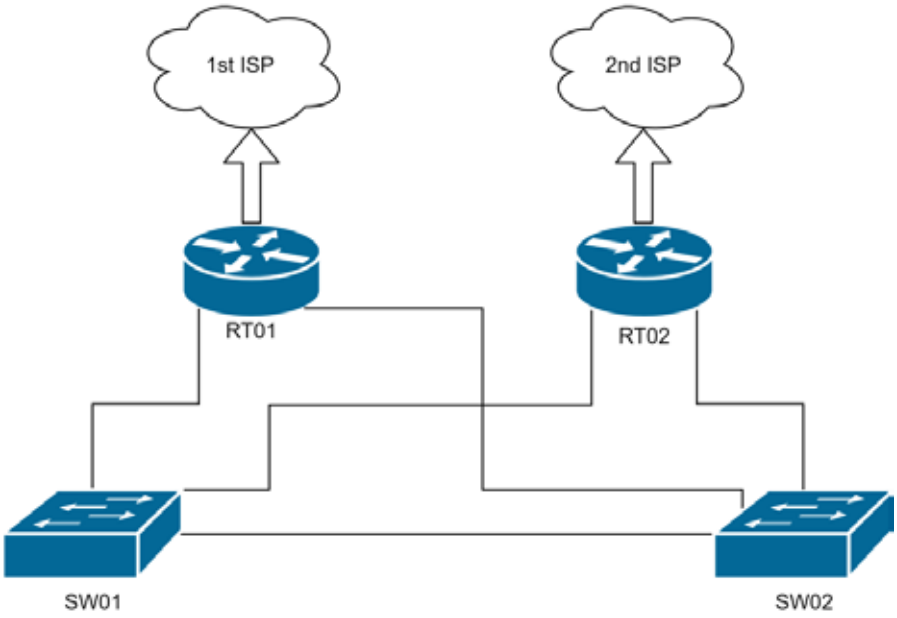

Рис. 1. Класичний приклад резервації каналів

В даній конфігурації потрібно звернути увагу на стек технологій IP SLA+TRACK, HSRP (Hot Standby Router Protocol) та EEM (Embedded Event Manager). Перша пара інструментів реалізує перевірку доступності певних ресурсів в мережі Інтернет через підключений Інтернет-канал. Якщо за дві секунди на запит не прийшла відповідь, тест IP SLA вважається не пройденим. Запускається лічильник TRACK і, якщо за цей час повторні IP SLA не повернуть позитивний результат, у дію запускається протокол HSRP. На інтерфейсі змінюється пріоритет, і роль головного маршрутизатора перехоплює резервний маршрутизатор. EEM в свою чергу потрібен для очищення NAT-трансляцій аби запобігти переповненню пам'яті, оскільки при перенаправлені трафіку через інший маршрутизатор дані записи втрачають свою актуальність.

Приклад 2. Наступний підхід стосується мережі ДЦ (дата-центр). В подібній мережі використовується велика кількість серверів та мережевого обладнання [5]. З'єднання будуються вичерпним чином, тобто є зарезервованими. У разі виникнення втрат чи перевантаження каналів на якомусь спайні (комутаторі), трафік потрібно м'яко (тобто без розриву з'єднання) перенаправити через інші маршрути. На рисунку 2 проблема виникає на верхньому спайні, трафік перенаправлено через нижній спайн [6].

Для вирішення подібної проблеми в складній мережі ДЦ використовується складний стек технологій MPTCP (MultiPath Transmission Control Protocol), Flow Label IPv6, eBPF (extended Berkeley Packet Filter). Поле заголовку IPv6 Flow Label з'являється в IPv6 (його немає у IPv4) і воно займає 20 біт. eBPF як невелику програму на C можна вставити в різних місцях виконання стека ядра та TCP-стека. За допомогою eBPF можна динамічно змінювати різноманітні налаштування TCP, в тому числі знизити таймери RTO та SYN-RTO – вплинути на час реакції на подію. Програмування ядра – низькорівнева задача, що вимагає високого рівня компетенцій та навичок.

Між конфігурацією у наведених прикладах різниця у понад 10 років, при цьому: топологія мережі – різна, масштаби – різні, складність – кардинально різна, основна задача в обох випадках – одна: виявити збій та перенаправити трафік альтернативним маршрутом. І тільки наступним кроком настає етап для з'ясування причини збою.

Таблиця 1

Конфігурація маршрутизаторів

Перший маршрутизатор	Другий маршрутизатор
track 1 ip sla 1 reachability delay down 5 up 5 ! interface vlan100 description -=LAN=- ip address 10.1.100.253 255.255.255.0 ip nat inside standby 100 ip 10.1.100.254 standby 100 priority 150 standby 100 track 101 decrement 60 ! ! interface GigabitEthernet0/0/1 description -=1st ISP=- ip address 1.1.1.2 255.255.255.0 ip nat outside ! ip route 0.0.0.0 0.0.0.0 1.1.1.1 ip route 10.0.0.0 255.0.0.0 10.1.100.252 250 ! ip sla 1 icmp-echo 8.8.8.8 source-interface GigabitEthernet0/0/1 threshold 1500 timeout 2000 frequency 3 ip sla schedule 1 life forever start-time now ! event manager applet ISP1-UP event track 1 state up maxrun 40 action 001 wait 30 action 002 cli command "enable" action 003 cli command "clear ip nat trans *" action 004 syslog msg "EEM cleared nat" action 005 cli command "end" action 006 cli command "exit" event manager applet ISP1-DOWN event track 1 state down maxrun 40 action 001 wait 30 action 002 cli command "enable" action 003 cli command "clear ip nat trans *" action 004 syslog msg "EEM cleared nat trans" action 005 cli command "end" action 006 cli command "exit"	track 2 ip sla 2 reachability delay down 5 up 5 ! interface vlan100 description -=LAN=- ip address 10.1.100.252 255.255.255.0 ip nat inside standby 100 ip 10.1.100.254 standby 100 priority 120 standby 100 preempt standby 100 track 12 decrement 60 ! interface GigabitEthernet0/0/1 description -=2nd ISP=- ip address 2.2.2.2 255.255.255.0 ip nat outside ! ip route 0.0.0.0 0.0.0.0 2.2.2.1 ip route 10.0.0.0 255.0.0.0 10.1.100.253 250 ! ip sla 2 icmp-echo 8.8.8.8 source-interface GigabitEthernet0/0/1 threshold 1500 timeout 2000 frequency 3 ip sla schedule 2 life forever start-time now ! event manager applet ISP2-UP event track 2 state up maxrun 40 action 001 wait 30 action 002 cli command "enable" action 003 cli command "clear ip nat trans *" action 004 syslog msg "EEM cleared nat trans" action 005 cli command "end" action 006 cli command "exit" event manager applet ISP2-DOWN event track 2 state down maxrun 40 action 001 wait 30 action 002 cli command "enable" action 003 cli command "clear ip nat trans *" action 004 syslog msg "EEM cleared nat trans" action 005 cli command "end" action 006 cli command "exit"

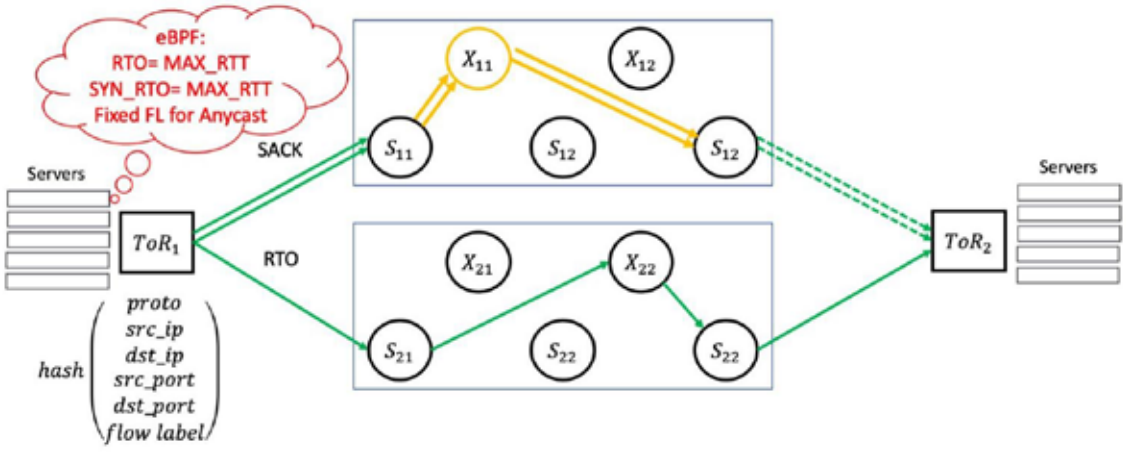

Рис. 2. Перенаправлення трафіку через інші спайни

Звідси можна отримати формулу часу реакції на збій:

$$T = (t_1 - t_0) + (t_2 - t_1) = t_2 - t_0, \quad (1)$$

де t_0 – час збою, t_1 – час виявлення збою, t_2 – час, коли було вжито дії.

На рисунку 3 зображено моніторинг зв'язності між ДЦ [7]. Коли виникає проблема, відповідна секція втрачає зелений колір. Інженер NOC (Network Operation Centre) отримує оповіщення та має відреагувати на інцидент. Для цього йому доведеться проаналізувати додаткові дані, ймовірно, зайти на обладнання, подивитися log-файли.

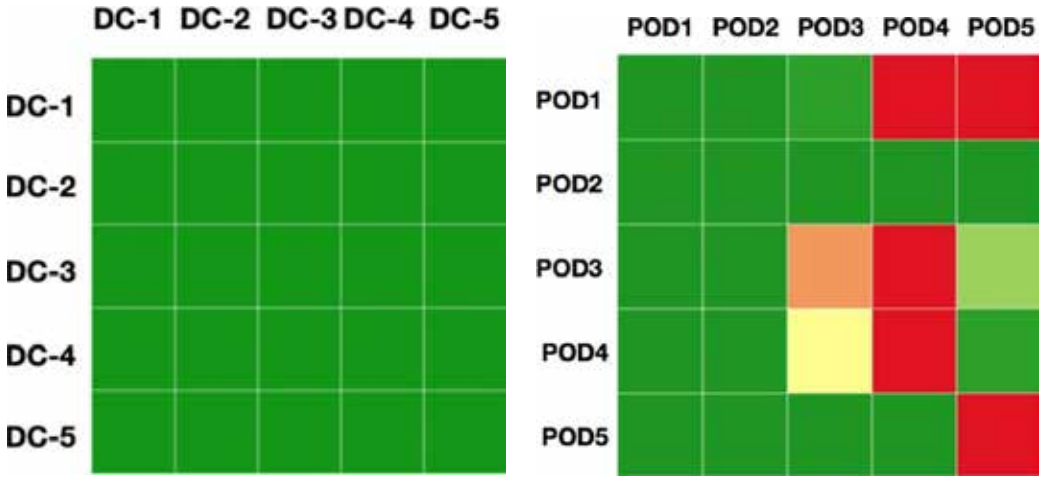

Рис. 3. Зв'язаність між різними ДЦ

До цього моменту було розглянуто приклади, де система чи пристрій автоматично виявляли збій (t_1) та автоматично вживали дії щодо його усунення (t_2). Відповідні таймери можна регулювати в протоколах та впливати таким чином на час реакції на подію. У випадку, коли інцидент обробляє NOC, t_1 залежить від налаштувань моніторингу, на нього також можна впливати; t_2 у свою чергу повністю залежить від роботи інженера та є важко передбачуваним.

Для того, щоб виявити збій, потрібен моніторинг. Мережа кожного окремого підприємства індивідуальна та може виконувати різні специфічні задачі. Доцільність моніторингу того чи іншого параметру мережі визначають інженери компанії. Більше того, деякі проблеми можуть бути подібно до «zero day» – до сьогодні невідомими. Відповідно невідомо, що потрібно відслідковувати у моніторингу. Додавати у моніторинг все не тільки недоцільно, але й неможливо.

Для збільшення прогнозованості часу реакції на подію потрібна автоматизація процесу.

Приклад 3. Нижче наведено фрагмент коду, де скрипт перевіряє коректність введеної команди на обладнанні. Якщо було задано помилкову команду, система виправляє конфігурацію на потрібну. Для реалізації використовується стек NETCONF та Python. Подібні скрипти використовуються з метою вичитки конфігурації після молодших інженерів, виправлення помилок, їх додавання у журнали з метою подальшої роботи над помилками при навчанні інженерів.

```
interface_type = "GigabitEthernet"
interface_id = 1
desired_description = "Tunnel endpoint to cloud provider"

configured+description = get_interface_description(device=DEVICE,
                                        interface_type=interface_type,
                                        interface_id=interface_id)

print(f"Desired interface description: '{desired_description}'\n"
   f"Configured interface description: '{configured_description}'")
```

```
if configured_description != desired_description:
   print("Desired does NOT match configured. Updating…")
   configure_interface_description(device=DEVICE,
                           interface_type=interface_type,
                           interface_id=interface_id,
                           interface_description=desired_description)
else:
   print("Desired description matches configured!")
```

Іншим прикладом може бути ситуація, коли мережу автоматизовано настільки ретельно, що вона не дозволяє інженеру руками вносити зміни на обладнані. Натомість попереджає, що усі зміни в мережі мають відбуватися через оркестратор, тобто через центральний стек серверів, що керують програмованою мережею.

Приклад 4. Self-Healing мережа має прагнути до автоматизації максимальної кількості етапів. У прикладах 1-2 відбувається автоматизоване переключення на резервні канали чи обладнання, але нічого не відбувається для усунення причини проблеми. Якщо комутатор вийшов з ладу, на сьогодні ще немає автоматизованих інструментів замінити його фізично. Проте, у разі падіння Інтернет-каналу процес його відновлення можна частково автоматизувати. Наприклад:

1. Система моніторингу фіксує збій у роботі каналу.
2. Запускає траблшутінг-скрипт на обладнанні.
3. Автоматично аналізує результати.
4. У разі наявності характерних «червоних прапорців» відправляє повідомлення поштою до ISP.
5. Якщо проблема була на стороні ISP, теоретично канал можуть відновити навіть віддалено. Або буде подальша діагностика, а даний етап отримує характер напівавтоматизованого.

Подібний сценарій, окрім мови програмування для написання скрипту, додатково потребує щонайменше протокол доставки команд на обладнання.

Загалом дослідження в напрямку повністю самовідновлювальної мережі тривають.

Частково програмування можуть замінити певні технології від розробників обладнання. Наприклад, Cisco NAE (Network Assurance Engine) – це комплексне рішення для мереж ДЦ, створене на основі запатентованої технології мережевої перевірки Cisco. Проте працювати воно буде лише на обладнанні від Cisco і лише на конкретних моделях з конкретною версією ПЗ (програмне забезпечення) [8; 9].

Висновки. У всіх прикладах неминуче присутні інструменти програмування. EEM – це функціонал, вбудований у Cisco IOS, який дозволяє створювати сценарії для автоматизації роботи пристроїв. eBFP – це підсистема ядра, що дає можливість писати невеликі програми, які будуть запущені ядром у відповідь на події. Python – це мова програмування високого рівня загального призначення. На місці Python можуть бути інші мови.

Отже, Self-Healing мережа обов'язково має програмовану складову, за допомогою якої і відбувається самовідновлення. Мережі майбутнього мають бути програмованими з централізованим керуванням.

Абсолютно все в мережі відслідковувати неможливо. Варто концентрувати увагу на дійсно потрібних метриках та за можливості намагатися автоматизувати в якості реакції на оповіщення процес відновлення мережі.

Сама мережа є динамічною, автоматизація – живий та постійний процес. Оскільки кожна мережа є індивідуальною, процес автоматизації також не може бути шаблонним. Необхідно володіти навичками програмування для автоматизації та гнучко поєднувати Open Source продукти з пропрієтарними продуктами від розробників обладнання.

Альтернативою програмуванню можуть бути пропрієтарні технології від розробників. Проте, при роботі з подібними інструменти інженер працює з певними абстракціями, за якими знову ж таки ховається код від виробника – програмована складова. Подібні технології вимагають повної підтримки обладнанням: тільки конкретний розробник обладнання, тільки на конкретних прошивках для обладнання. Оскільки складно побудувати мережу повністю на обладнанні лише одного розробника, найчастіше дані технології можуть автоматизувати тільки частину мережі, або не працювати взагалі.

Подальші дослідження будуть пов'язані зі створенням універсальних інструментів для самовідновлення мереж як програмно-визначених, так і традиційних.

Список використаних джерел:

1. What does the self-healing network of tomorrow look like? URL: https://irisns.com/2014/11/21/self-healing-network-tomorrow-look-like/

2. Збій Facebook: чому він стався і тривав так довго. BBC News Україна. URL: https://www.bbc.com/ukrainian/features-58795279

3. Thorat P., Jeon S., Choo H. Enhanced local detouring mechanisms for rapid and lightweight failure recovery in OpenFlow networks. *Computer Communications.* 2017. Vol. 108. P. 78-93.

4. Empson S., Roth H. CCNP ROUTE Command Guide: Implementing Path Control. *Cisco Press.* 2010. P. 199-208.

5. Clos Networks: What's Old Is New Again. Network World. URL: https://www.networkworld.com/article/2226122/clos-networks-what-s-old-is-new-again.html

6. Self healing Network The Magic of Flow Label – IETF Datatracker. URL: https://datatracker.ietf.org/meeting/111/materials/slides-111-rtgwg-sessb-3-selfhealing-network-01

7. Adrichem N.L. M.v., Asten B.J.v., Kuipers F.A. Fast recovery in software-defined networks. *Proc. Of the 3rd European Workshop on, Software-Defined Networks*. 2014. P. 61-66.

8. Cisco Network Assurance Engine – Cisco Network Assurance Engine At-a-Glance. Cisco. URL: https://www.cisco.com/c/en/us/products/collateral/data-center-analytics/network-assurance-engine/at-a-glance-c45-740230.html

9. Building self-healing networks with Cisco Network Assurance Innovations and Service Now ITSM. Cisco Blogs. URL: https://blogs.cisco.com/datacenter/building-self-healing-networks-with-cisco-network-assurance-innovations-and-servicenow-itsm

10. D. Ghosh Self-healing systems – survey and synthesis. *Decision Support Systems*. 2007. Vol. 42, no. 4. P. 2164–2185.

11. Al-Oqily I., Bani-Mohammad S., Subaih B., Alshaer J.J. A survey for self-healing architectures and algorithms. *Proc. of the International Multi-Conference on Systems, Signals Devices*. 2012. P. 1-5.

12. Cascone C., Pollini L., Sanvito D., Capone A. Traffic management applications for stateful SDN data plane. *Proc. of the Fourth European Workshop on Software-Defined Networks*. 2015. P. 85-90.

13. Системи моніторингу та керування – IT-Solutions, Україна. IT-Solutions, Україна. URL: https://it-solutions.ua/servisi/sistemi-monitoringu-ta-keruvannya/

14. Sanchez J., Ben Yahia I.G., Crespi N. Self-Healing Mechanisms for Software Defined Networks. *8th International Conference on Autonomous Infrastructure, Management and Security.* 2014.

15. Ochoa-Aday L., Cervelló-Pastor C., Fernández-Fernández A. Self-healing and SDN: bridging the gap. *Digital Communications and Networks*. 2020. Vol. 6, no. 3. P. 354-368.

References:

1. What does the self-healing network of tomorrow look like? (n.d.). Retrieved from: https://irisns.com/2014/11/21/self-healing-network-tomorrow-look-like/

2. Zbiy Facebook: chomu vin stavsya i tryvav tak dovho [The Facebook crash: why it happened and lasted so long]. (2021). BBC News Ukraine. Retrieved from: https://www.bbc.com/ukrainian/features-58795279 [in Ukrainian].

3. Thorat P., Jeon S., Choo H. (2017). Enhanced local detouring mechanisms for rapid and lightweight failure recovery in OpenFlow networks. *Computer Communications.* Vol. 108. P. 78-93.

4. Empson S., Roth H. (2010). CCNP ROUTE Command Guide: Implementing Path Control. *Cisco Press.* P. 199-208.

5. Clos Networks: What's Old Is New Again. Network World. (2014). Retrieved from: https://www.networkworld.com/article/2226122/clos-networks-what-s-old-is-new-again.html

6. Self healing Network The Magic of Flow Label – IETF Datatracker. (n.d.). Retrieved from: https://datatracker.ietf.org/meeting/111/materials/slides-111-rtgwg-sessb-3-selfhealing-network-01

7. Adrichem N.L. M.v., Asten B.J.v., Kuipers F.A. (2014). Fast recovery in software-defined networks. *Proc. Of the 3rd European Workshop on, Software-Defined Networks*. P. 61-66.

8. Cisco Network Assurance Engine – Cisco Network Assurance Engine At-a-Glance. Cisco. (2021). Retrieved from: https://www.cisco.com/c/en/us/products/collateral/data-center-analytics/network-assurance-engine/at-a-glance-c45-740230.html

9. Building self-healing networks with Cisco Network Assurance Innovations and Service Now ITSM. Cisco Blogs. (2021). Retrieved from: https://blogs.cisco.com/datacenter/building-self-healing-networks-with-cisco-network-assurance-innovations-and-servicenow-itsm

10. Ghosh D. (2007). Self-healing systems – survey and synthesis. *Decision Support Systems*. Vol. 42, no. 4. P. 2164–2185.

11. Al-Oqily I., Bani-Mohammad S., Subaih B., Alshaer J.J. (2012). A survey for self-healing architectures and algorithms. *Proc. of the International Multi-Conference on Systems, Signals Devices*. P. 1-5.

12. Cascone C., Pollini L., Sanvito D., Capone A. (2015). Traffic management applications for stateful SDN data plane. *Proc. of the Fourth European Workshop on Software-Defined Networks*. P. 85-90.

13. Systemy monitorynhu ta keruvannya [Systems of monitoring and management] – IT-Solutions, Ukraine. IT-Solutions, Ukraine. (n.d.). Retrieved from: https://it-solutions.ua/servisi/sistemi-monitoringu-ta-keruvannya/ [in Ukrainian].

14. Sanchez J., Ben Yahia I.G., Crespi N. 2014. Self-Healing Mechanisms for Software Defined Networks. *8th International Conference on Autonomous Infrastructure, Management and Security*.

15. Ochoa-Aday L., Cervelló-Pastor C., Fernández-Fernández A. (2020). Self-healing and SDN: bridging the gap. *Digital Communications and Networks*. Vol. 6, no. 3. P. 354-368.

UDC 004.8
DOI https://doi.org/10.32689/maup.it.2023.5.4

Serhii KOLOMOIETS
Postgraduate Student, Assistant at the Department of Information Systems, Faculty of Informatics and Computer Science, National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", 37, Beresteyskyi Ave, Kyiv, Ukraine, postal code 03056 (serhii.o.kolomoiets@ukr.net)

ORCID: 0000-0003-3741-4517

Сергій КОЛОМОЄЦЬ
аспірант, асистент кафедри Інформаційних систем та технологій, факультету Інформатики та обчислювальної техніки, Національний технічний університет України «Київський політехнічний Інститут імені Ігоря Сікорського», просп. Берестейський, 37, Київ, Україна, індекс 03056 (serhii.o.kolomoiets@ukr.net)

Bibliographic description of the article: Kolomoiets, S. (2023). Kontseptsii stvorennia intelektualnoi medychnoi systemy diahnostyky dlia dopomohy v roboti ta navchanni likariv na osnovi shtuchnoho intelektu [Concepts of creating an intelligent medical diagnostic system to assist in the work and training of doctors based on artificial intelligence]. *Informatsiini tekhnolohii ta suspilstvo – Information technology and society*, 5–6 (11–12), 28–33. DOI: https://doi.org/10.32689/maup.it.2023.5.4

Бібліографічний опис статті: Коломоєць, С. (2023). Концепції створення інтелектуальної медичної системи діагностики для допомоги в роботі та навчанні лікарів на основі штучного інтелекту. *Інформаційні технології та суспільство*, 5 (11), 28–33. DOI: https://doi.org/10.32689/maup.it.2023.5.4

CONCEPTS OF CREATING AN INTELLIGENT MEDICAL DIAGNOSTIC SYSTEM TO ASSIST IN THE WORK AND TRAINING OF DOCTORS BASED ON ARTIFICIAL INTELLIGENCE

Abstract. *The article is devoted to the development of intelligent medical diagnostic systems using information systems and technologies. The article provides an overview of the current state and development of information systems, technologies and artificial intelligence in the medical field, analyzes existing intelligent diagnostic systems in medicine and cardiology in particular, and proves the need to create intelligent systems to help primary care physicians and cardiologists. The purpose of the publication was to investigate the current state of intelligent medical diagnostic systems, analyze the shortcomings of such systems, determine the feasibility of creating a new diagnostic system, and formulate principles and criteria for its operation. The scientific novelty of the article is a new approach for diagnosing and treating patients, as well as training medical professionals using an intelligent medical system based on artificial intelligence. Leveraging advanced AI algorithms, the system analyzes vast datasets encompassing patient records, medical literature, and real-time clinical data to provide accurate and timely diagnostic insights. The intelligent medical diagnostic system operates as a supportive tool, aiding doctors in the diagnostic process by offering refined suggestions, identifying potential anomalies, and recommending personalized treatment plans. By harnessing machine learning and deep learning techniques, the system continuously adapts and evolves, learning from each diagnostic scenario and refining its predictive accuracy over time. Through interactive simulations and case-based learning modules, aspiring and practicing doctors can engage in immersive, realistic scenarios, honing their diagnostic skills and expanding their knowledge base. The disadvantages of this article include only a theoretical approach to the formation of concepts and tasks for an intelligent medical system, and a review of existing systems.*

Key words: *diagnostics, intelligent information system, training, forecasting, artificial intelligence.*

КОНЦЕПЦІЇ СТВОРЕННЯ ІНТЕЛЕКТУАЛЬНОЇ МЕДИЧНОЇ СИСТЕМИ ДІАГНОСТИКИ ДЛЯ ДОПОМОГИ В РОБОТІ ТА НАВЧАННІ ЛІКАРІВ НА ОСНОВІ ШТУЧНОГО ІНТЕЛЕКТУ

Анотація. *Стаття присвячена розвитку інтелектуальних медичних систем діагностики з використанням інформаційних систем та технологій. У статті представлено огляд сучасного стану і розвитку інформаційних систем, технологій та штучного інтелекту в медичній галузі, проаналізовано існуючи інтелектуальні системи діагностики в медицині та кардіології зокрема, доведено необхідність створення інтелектуальних систем для допомоги лікарям первинної ланки та кардіологам. Метою публікації було дослідити сучасний стан інтелектуальних медичних систем діагностики, проаналізувати недоліки таких систем, визначити доцільність створення нової системи діагностики та сформулювати принципи та критерії для її роботи. Науковою новизною статті є новий підхід для діагностики та лікування пацієнтів, а також навчання медичних працівників за допомогою інтелектуальної медичної системи на основі штучного інтелекту. Використовуючи передові алгоритми штучного інтелекту, система аналізує великі масиви даних, що охоплюють записи пацієнтів, медичну літературу і клінічні дані в режимі реального часу, щоб надати точну і своєчасну діагностичну інформацію. Інтелектуальна медична діагностична система працює як допоміжний інструмент, допомагаючи лікарям*

у процесі діагностики, пропонуючи уточнені пропозиції, визначаючи потенційні аномалії та рекомендуючи персоналізовані плани лікування. Використовуючи методи машинного та глибинного навчання, система постійно адаптується та розвивається, навчаючись на кожному діагностичному сценарії та вдосконалюючи свою точність прогнозування з часом. Завдяки інтерактивним симуляціям і навчальним модулям, заснованим на конкретних випадках, лікарі-початківці та практикуючі лікарі можуть брати участь у захоплюючих, реалістичних сценаріях, відточуючи свої діагностичні навички та розширюючи свою базу знань. До недоліків даної статті можна віднести лише теоретичний підхід до формування концепцій та задач до інтелектуальної медичної системи, огляд вже існуючих систем.

Ключові слова: *діагностика, інтелектуальна інформаційна система, навчання, прогнозування, штучний інтелект.*

Introduction. In 2019, Ukraine conducted the first nationwide survey to study the prevalence of noncommunicable disease (NCD) risk factors – STEPS [1].

The STEPS study reflects the World Health Organization's (WHO) STEPwise approach to noncommunicable disease risk factor surveillance and is a simple, standardized method for collecting, analyzing and disseminating data in WHO member countries.

The STEPS tool covers three different levels of risk factor assessment – three steps: questionnaires, physical measurements, and biochemical analyses. Each level is divided into basic, advanced and additional modules that can be used in the study depending on the conditions and needs of different countries.

The research results can be used not only to study trends within a country, but also for comparisons between countries. In particular, they contain socio-demographic indicators, data on alcohol and tobacco consumption, nutrition, physical activity, blood pressure, glucose, cholesterol, information on the presence of circulatory system diseases, physical examination data, and biochemical indicators.

More than 130 countries around the world have participated in STEPS at least once. In some countries, this study has been conducted several times, while in Ukraine it was organized for the first time. The study was conducted in 2019, and in 2021, WHO presented the results to the public.

In Ukraine, cardiovascular disease is the leading cause of death. By this indicator, our country remains one of the world leaders (Figure 1) [2].

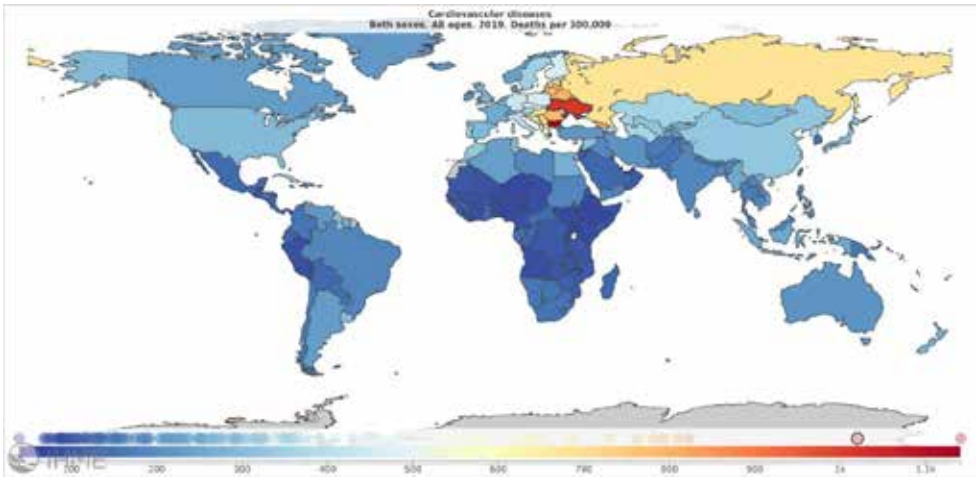

Fig. 1. Statistics from IHME (Institute for Health Metrics and Evaluation founded by Bill and Melinda Gates). Global mortality rates from cardiovascular diseases in 2019

According to a ranking based on the number of deaths in Ukraine, the most common causes are:

1. Cardiovascular diseases (64.3%).
2. Neoplasms (14.1%).
3. Diseases of the digestive system (4.3%).
4. Neurological disorders (3.1%).
5. Self-harm and interpersonal violence (2.7%).

It was also possible to identify the most common causes of death from cardiovascular diseases by gender among all age groups (Table 1).

Table 1

Causes of death of Ukrainians from cardiovascular diseases

№	Male	Female
1	Coronary heart disease	Coronary heart disease
2	Cerebrovascular diseases	Cerebrovascular diseases
3	Cardiomyopathy and myocarditis	Cardiomyopathy and myocarditis
4	Diseases of peripheral vessels	Atrial fibrillation
5	Aortic aneurysm	Diseases of peripheral vessels
6	Atrial fibrillation	Other cardiovascular diseases
7	Other cardiovascular diseases	Hypertensive heart disease
8	Hypertensive heart disease	Rheumatic heart disease
9	Rheumatic heart disease	Aortic aneurysm
10	Endocarditis	Non-neuromuscular valve disorders

From the above data, we can conclude that cardiovascular diseases require more attention, especially in Ukraine. The average time for an outpatient visit to a doctor is 15-20 minutes, during which it is impossible to fully diagnose a patient's problem and prescribe treatment. The patient has to undergo a large number of tests and repeatedly come to the doctor's office. There are also often difficult cases when the doctor is unsure of the diagnosis and needs to prescribe additional tests or an outside expert opinion. These problems can be solved by optimizing the work of healthcare professionals with an AI-based intelligent cardiovascular disease diagnostic system that would predict diagnosis and treatment based on the available medical history to verify the doctor's preliminary diagnosis, work in conditions of incomplete information, suggest additional necessary tests (if necessary), and help with the determination of the treatment regimen.

Problem statement. Resistant arterial hypertension (RAH) is defined as hypertension in which blood pressure (BP) remains above the target level despite the simultaneous use of 3 antihypertensive drugs of different classes [3]. Ideally, one of the 3 drugs should be a diuretic and all drugs should be prescribed in optimal doses. Despite the relative arbitrariness of the requirements for the number of prescribed drugs, resistant hypertension is defined in order to identify patients who are at higher risk or who have treatable causes of hypertension and/or patients who may benefit from additional special diagnostic and therapeutic measures due to persistently high blood pressure. As defined, resistant hypertension includes patients whose blood pressure is controlled with more than 3 medications. That is, patients whose blood pressure is controlled but requires 4 or more medications should be considered resistant to treatment (Figure 2).

The study focused on the problem of predicting the probability of true resistant hypertension based on the analysis of general clinical data for primary care physicians (family doctors, general practitioners). This intelligent system is designed for both primary care physicians and subspecialists – cardiologists, because patients always see a primary care physician first, and then a subspecialist. The task consists of two stages: 1) to create an intelligent system based on artificial intelligence, which, based on the available tests and data from the primary care physician, could determine whether the patient has true or false resistance to treatment of hypertension. 2) If the patient does have true resistance, which fourth-line drug would be appropriate for him or her to overcome hypertension.

The analysis is based on two groups of patients: controlled hypertension (CH) and resistant hypertension (RH). To determine the patient group, the physician must have the following patient data:

- office blood pressure (BP) level;
- heart rate;
- duration of arterial hypertension (AH);
- body mass index (BMI);
- waist circumference (WC);
- history of stroke;
- coronary heart disease (CHD), myocardial infarction;
- atrial fibrillation (AF);
- diabetes mellitus (DM);
- ECHO cardiography parameters (Tcp, Tmrp, LVMI, LV);
- blood test results: potassium, sodium, creatinine, total cholesterol, LDL cholesterol, TG, glucose, CRP, uric acid (UA);
- glomerular filtration rate (GFR);
- albuminuria (BUN).

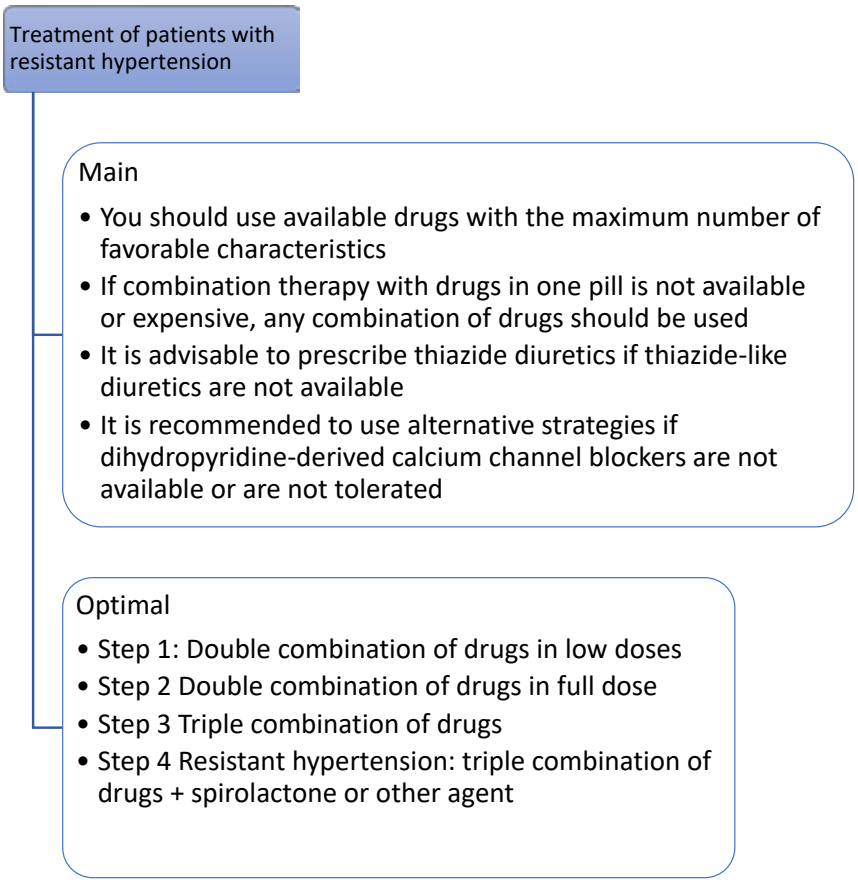

Fig. 2. Pharmacological strategy for the treatment of patients with hypertension in accordance with the standards

These parameters can be used to identify truly resistant patients. If the patient does not have true resistance, it means that he or she is pseudo-resistant, which may be due to: incorrect BP measurement technique, low adherence to treatment, and "white coat" hypertension [4]. If therapy is followed correctly, a pseudoresistant patient will overcome hypertension. After that, the intelligent system should determine the sensitivity to the fourth antihypertensive drug in patients with true RAH based on all available indicators.

Intelligent information system. An intelligent information system (IIS) is a type of automated information system; a set of software, linguistic and logical-mathematical tools for the implementation of the main task: to support human activity and search for information in the mode of extended dialogue in natural language [5].

The characteristic features of IIS are:

1) developed communication skills;

2) the ability to solve complex and poorly formalized problems;

3) ability to develop and self-learn.

Intelligent information systems can be divided into 3 classes (Table 2):

Table 2

Classification of intelligent information systems

Class I: systems with an intelligent interface (communication abilities)	Class II: expert systems (solving complex problems)	Class III: systems capable of self-learning
1. Intelligent databases	1. Systems that classify	1. Inductive systems
2. Natural language interface	2. Systems that redefine	2. Neural networks
3. Hypertext systems	3. Systems that transform	3. Precedent-based systems
4. Contexts and systems	4. Multi-agent systems	4. Information repositories
5. Cognitive graphics		

Intelligent diagnostic systems in cardiology. Intelligent systems based on artificial intelligence are constantly becoming more widespread and in demand in medicine, especially in certain areas such as ophthalmology, cardiology, oncology, orthopedics, and dentistry [6]. It can be concluded that the use of intelligent diagnostic systems in medicine is widely used and is in demand and expedient. Let's consider the feasibility of using intelligent diagnostic systems in cardiology, since cardiovascular mortality is the highest in Ukraine.

As of today, there are many works and created intelligent diagnostic systems dedicated to cardiology. Among them:

1) Intelligent decision support system for radionuclide diagnostics in cardiology [7];
2) Ischemic heart disease [8];
3) Valvular heart disease [9];
4) Atrial fibrillation [10];
5) Heart failure [11];
6) Cardiomyopathy [12];
7) Congenital heart disease [13];

An analysis of publications [7-13] confirms that the creation of intelligent systems based on artificial intelligence improves the diagnosis and treatment of both cardiac and general diseases. Since the problem of identifying a patient with hypertension treatment resistance is extremely urgent and poorly diagnosed by doctors, it would be advisable to use information systems and technologies to solve this problem.

Consider the problem of intellectual classification of a set of patient's signs in order to prescribe the optimal method (protocol) of hypertension treatment

Algorithm for improving the diagnosis and treatment of true/false resistance of hypertension:

1) Examination of a patient with controlled/resistant hypertension by a primary care physician, examination, and initial tests. After receiving the tests, the doctor uses an intelligent system to classify the patient (enters all available data).

2) Classification task. Determine whether the patient has true drug resistance (if the resistance is false, the patient needs to lose weight, modify lifestyle, strictly adhere to medication, and if these conditions are met, hypertension will be controlled and blood pressure will be within normal limits).

Diagnosed: if the patient's blood pressure is above 160 mmHg, lifestyle modifications have been made, weight is normal, and the patient is taking 3 antihypertensive drugs, the primary care physician assumes that the patient has true resistance to antihypertensive drugs and refers the patient to a specialized cardiologist.

Proposal: The primary care physician enters all available patient data (age, gender, height, weight, race), test results, and patient lifestyle questionnaire data into the app; the neural network classifies the patient into one of two groups based on the available data.

3) If resistance to three drugs is true, the cardiologist must select a fourth drug to stabilize the patient's blood pressure.

Diagnosed: the doctor tries to determine the optimal combination for the patient by selecting different drugs (time-consuming, costly, the patient will have high blood pressure and discomfort for a long time due to the search for a combination that suits him).

Proposal: the app will calculate which fourth drug will be suitable for this particular patient based on additional tests and trained on a large sample of patients.

Conclusion. The most common cause of death in Ukraine is cardiovascular disease. Stabilizing blood pressure can significantly reduce mortality and improve the quality of life of patients. To determine the true resistance of hypertension, speed up the diagnostic process, understand the feasibility of prescribing additional tests, and select the fourth line of therapy for patients with resistant hypertension, it would be advisable to use an ANN to solve the decision-making problem. It is proposed to create an artificial neural network (multilayer perceptron), where the inputs will be the basic data for determining the group of patients with true hypertension, and the output will be one, because it is a binary classification task. The following publications will discuss the results of the described ANN. The intelligent medical diagnostic system represents a pioneering advancement in the healthcare landscape, revolutionizing the diagnostic process and elevating the standard of medical education. As AI continues to evolve, its synergistic integration with human expertise holds the promise of a more efficient, accurate, and personalized healthcare experience.

Bibliography:

1. Національне дослідження STEPS в Україні. URL: https://phc.org.ua/naukova-diyalnist/doslidzhennya/doslidzhennya-z-neinfekciynikh-zakhvoryuvan/nacionalne-doslidzhennya-steps-v-ukraini

2. Центр громадського здоров'я МОЗ України. Серцево-судинні захворювання – головна причина смерті українців. 2021. URL: https://phc.org.ua/naukova-diyalnist/doslidzhennya/doslidzhennya-z-neinfekciynikh-zakhvoryuvan/nacionalne-doslidzhennya-steps-v-ukraini

3. Коробка О. Практичні рекомендації щодо ведення пацієнтів з артеріальною гіпертензією. Кардіологія, Ревматологія, Кардіохірургія. 2020. № 4 (71) С. 25-27 URL: https://health-ua.com/multimedia/userfiles/files/2020/Cardio_4_2020/Cardio_4_2020_st25_27.pdf

4. Лисенко Г.І., Ященко О.Б. Медикаментозне лікування пацієнтів із артеріальною гіпертензією. 2011. Укр. Мед. Часопис, 3 (83) – V/VI. URL: https://api.umj.com.ua/wp/wp-content/uploads/2011/06/3002.pdf

5. Інтелектуальна інформаційна система. URL: https://pidru4niki.com/74257/informatika/intelektualna_informatsiyna_sistema

6. Висоцький А.А., Суріков О.О., Василюк-Зайцева С.В. Розвиток штучного інтелекту в сучасній медицині. 2023. Укр. Мед. Часопис 2 (154) – III/IV

7. Москаленко А. С. Інтелектуальна система підтримки прийняття рішень для радіонуклідної діагностики в кардіології. 2016. Радіоелектронні і комп'ютерні системи № 3. С. 49–55 URL: http://www.irbis-nbuv.gov.ua/cgi-bin/irbis_nbuv/cgiirbis_64.exe?I21DBN=LINK&P21DBN=UJRN&Z21ID=&S21REF=10&S21CNR=20&S21STN=1&S21FMT=ASP_meta&C21COM=S&2_S21P03=FILA=&2_S21STR=recs_2016_3_8

8. Betancur J., Commandeur F., Motlagh M. Deep Learning for Prediction of Obstructive Disease From Fast Myocardial Perfusion SPECT: A Multicenter Study. 2018. URL: https://www.jacc.org/doi/abs/10.1016/j.jcmg.2018.01.020

9. Kwon J.M., Lee S.Y., Jeon K.H., Lee Y. Deep Learning–Based Algorithm for Detecting Aortic Stenosis Using Electrocardiography. 2020. URL: https://www.ahajournals.org/doi/full/10.1161/JAHA.119.014717

10. Khurshid S., Friedman S., Reeder C. ECG-Based Deep Learning and Clinical Risk Factors to Predict Atrial Fibrillation. 2021. URL: https://www.ahajournals.org/doi/full/10.1161/CIRCULATIONAHA.121.057480

11. Zachi I. Attia, Kapa S., Lopez-Jimenez F. Screening for cardiac contractile dysfunction using an artificial intelligence–enabled electrocardiogram. 2019. URL: https://www.nature.com/articles/s41591-018-0240-2

12. Khurshid S., Friedman S., Pirruccello J.P. Deep Learning to Predict Cardiac Magnetic Resonance–Derived Left Ventricular Mass and Hypertrophy From 12-Lead ECGs. 2021. URL: https://www.ahajournals.org/doi/full/10.1161/CIRCIMAGING.120.012281

13. Arnaout R., Curran L., Zhao Y. An ensemble of neural networks provides expert-level prenatal detection of complex congenital heart disease. 2021. URL: https://www.nature.com/articles/s41591-021-01342-5%C2%A0

References:

1. Nacionalne doslidzhennya STEPS v Ukraini [National STEPS study in Ukraine]. Retrieved from https://phc.org.ua/naukova-diyalnist/doslidzhennya/doslidzhennya-z-neinfekciynikh-zakhvoryuvan/nacionalne-doslidzhennya-steps-v-ukraini [in Ukrainian].

2. Sercevo sudinni zakhvoryuvannya – golovna prichina smerti Ukrainciv visnovki z doslidzhennya [Cardiovascular disease is the leading cause of death in Ukraine. Findings from the global burden of disease study in 2019]. Retrieved from https://phc.org.ua/news/sercevo-sudinni-zakhvoryuvannya-golovna-prichina-smerti-ukrainciv-visnovki-z-doslidzhennya [in Ukrainian].

3. Praktychni rekomendatsii shchodo vedennia patsiientiv z arterialnoiu hipertenziieiu [Practical recommendations for the management of patients with hypertension]. Retrieved from https://health-ua.com/multimedia/userfiles/files/2020/Cardio_4_2020/Cardio_4_2020_st25_27.pdf [in Ukrainian].

4. G.I. Lysenko, O.B. Yashchenko (2011). Medykamentozne likuvannia patsiientiv iz arterialnoiu hipertenziieiu [Drug treatment of patients with arterial hypertension]. Retrieved from https://api.umj.com.ua/wp/wp-content/uploads/2011/06/3002.pdf [in Ukrainian].

5. Intelektualna informatsiina systema [Intelligent systems and technologies in organization management]. Retrieved from https://pidru4niki.com/74257/informatika/intelektualna_informatsiyna_sistema [in Ukrainian].

6. A.A. Vysotskyi, O.O. Surikov, S.V. Vasyliuk-Zaitseva (2023). Rozvytok shtuchnoho intelektu v suchasnii medytsyni [Development of artificial intelligence in modern medicine]. Retrieved from https://api.umj.com.ua/wp/wp-content/uploads/2023/04/5262.pdf [in Ukrainian].

7. Moskalenko A. S (2016). Intelektualna systema pidtrymky pryiniattia rishen dlia radionuklidnoi diahnostyky v kardiolohii [Intelligent decision support system for radionuclide diagnostics in cardiology]. Retrieved from http://www.irbis-nbuv.gov.ua/cgi-bin/irbis_nbuv/cgiirbis_64.exe?I21DBN=LINK&P21DBN=UJRN&Z21ID=&S21REF=10&S21CNR=20&S21STN=1&S21FMT=ASP_meta&C21COM=S&2_S21P03=FILA=&2_S21STR=recs_2016_3_8 [in Ukrainian].

8. Betancur J., Commandeur F., Motlagh M. (2018). Deep Learning for Prediction of Obstructive Disease From Fast Myocardial Perfusion SPECT: A Multicenter Study. Retrieved from https://www.jacc.org/doi/abs/10.1016/j.jcmg.2018.01.020

9. Kwon J.M., Lee S.Y., Jeon K.H., Lee Y. (2020). Deep Learning–Based Algorithm for Detecting Aortic Stenosis Using Electrocardiography. Retrieved from https://www.ahajournals.org/doi/full/10.1161/JAHA.119.014717

10. Khurshid S., Friedman S., Reeder C. (2021). ECG-Based Deep Learning and Clinical Risk Factors to Predict Atrial Fibrillation. Retrieved from https://www.ahajournals.org/doi/full/10.1161/CIRCULATIONAHA.121.057480

11. Zachi I. Attia, Kapa S., Lopez-Jimenez F. (2019). Screening for cardiac contractile dysfunction using an artificial intelligence–enabled electrocardiogram. Retrieved from https://www.nature.com/articles/s41591-018-0240-2

12. Khurshid S., Friedman S., Pirruccello J.P. (2021). Deep Learning to Predict Cardiac Magnetic Resonance–Derived Left Ventricular Mass and Hypertrophy From 12-Lead ECGs. Retrieved from https://www.ahajournals.org/doi/full/10.1161/CIRCIMAGING.120.012281

13. Arnaout R., Curran L., Zhao Y. (2021). An ensemble of neural networks provides expert-level prenatal detection of complex congenital heart disease. Retrieved from https://www.nature.com/articles/s41591-021-01342-5%C2%A0

УДК 004.75
DOI https://doi.org/10.32689/maup.it.2023.5.5

Оксана КОШОВА
кандидат педагогічних наук, доцент, доцент кафедри комп'ютерних наук та інформаційних технологій, Полтавський університет економіки і торгівлі, вул. Коваля (Івана Банка), 3, Полтава, Україна, індекс 36000 (koshova.o111@gmail.com)
ORCID: 0000-0003-0794-6774

Оксана ЧЕРНЕНКО
кандидат фізико-математичних наук, доцент, доцент кафедри комп'ютерних наук та інформаційних технологій, Полтавський університет економіки і торгівлі, вул. Коваля (Івана Банка), 3, Полтава, Україна, індекс 36000 (oksanachernenko7@gmail.com)
ORCID: 0000-0002-9084-0999

Оксана ОРІХІВСЬКА
старший викладач кафедри комп'ютерних наук та інформаційних технологій, Полтавський університет економіки і торгівлі, вул. Коваля (Івана Банка), 3, Полтава, Україна, індекс 36000 (aka.jeita@gmail.com)
ORCID: 0000-0003-2775-0832

Володимир ТУР
аспірант, Полтавський університет економіки і торгівлі, вул. Коваля (Івана Банка), 3, Полтава, Україна, індекс 36014 (Tur.vladimir1983@gmail.com)
ORCID: 0009-0003-2825-1434

Олексій ЯНКО
здобувач освіти магістерського рівня напрямку «Комп'ютерні науки», Полтавський університет економіки і торгівлі, вул. Коваля (Івана Банка), 3, Полтава, Україна, індекс 36000
ORCID: 0009-0006-5469-1072

Oksana KOSHOVA
Candidate of Pedagogical Sciences, Associate Professor, Associate Professor at the Department of Computer Science and Information Technology, Poltava University of Economics and Trade, 3, Kovalia St, Poltava, Ukraine, postal code 36000 (koshova.o111@gmail.com)

Oksana CHERNENKO
Candidate of Physical and Mathematical Sciences, Associate Professor, Associate Professor of Department of Computer Science and Information Technology, Poltava University of Economics and Trade, 3, Kovalia St, Poltava, Ukraine, postal code 36000 (oksanachernenko7@gmail.com)

Oksana ORIKHIVSKA
Senior lecturer at the Department of Computer Science and Information Technology, Poltava University of Economics and Trade, 3, Kovalia St, Poltava, Ukraine, postal code 36000 (aka.jeita@gmail.com)

Volodymyr TOUR
Postgraduate Student, Poltava University of Economics and Trade, 3, Kovalia St, Poltava, Ukraine, postal code 36014 (Tur.vladimir1983@gmail.com)

Oleksiy YANKO
Master's Degree in Computer Science, Poltava University of Economics and Trade, 3, Kovalia St, Poltava, Ukraine, postal code 36000

Бібліографічний опис статті*:* Кошова, О., Черненко, О., Оріхівська, О., Тур, В., Янко, О. (2023). Розробка навчального андроїд-застосунку з теми «Сортування вставками» дистанційного навчального курсу «Алгоритми і структури даних». *Інформаційні технології та суспільство*, 5 (11), 34–42. DOI: https://doi.org/10.32689/maup.it.2023.5.5

Bibliographic description of the article: Koshova, O., Chernenko, O., Orikhovska, O., Tour, V., Yanko, O. (2023). Rozrobka navchalnoho android-zastosunku z temy «Sortuvannia vstavkamy» dystantsiinoho

navchalnoho kursu «Alhorytmy i struktury danykh» [Development of an educational Android application on the topic "Sorting by insertions" of the distance learning course "Algorithms and data structures"]. *Informatsiini tekhnolohii ta suspilstvo – Information technology and society*, 5 (11), 34–42. DOI: https://doi.org/10.32689/maup.it.2023.5.5

РОЗРОБКА НАВЧАЛЬНОГО АНДРОЇД-ЗАСТОСУНКУ З ТЕМИ «СОРТУВАННЯ ВСТАВКАМИ» ДИСТАНЦІЙНОГО НАВЧАЛЬНОГО КУРСУ «АЛГОРИТМИ І СТРУКТУРИ ДАНИХ»

Анотація. *Сучасний світ технологій вимагає все більш ефективних методів навчання, особливо в області програмування та комп'ютерних наук. Відповідаючи цим вимогам, виникає необхідність створення інтерактивих Android-застосунків, які є незамінними помічниками для дистанційного та змішаного форматів навчання.*

Мета роботи *– розробка мобільного навчального застосунку для Android на тему «Сортування вставками».*

Методологія. *Програмне забезпечення для Android-застосунку розроблене за допомогою мови програмування Kotlin та використання Android SDK для створення користувацького інтерфейсу. Перелік методів включає проєктування мобільних навчальних додатків, зокрема, методики UX/UI дизайну. Щодо технічних інструментів використано Android Studio для розробки, Git для контролю версій, а також бібліотеки та фреймворки, специфічні для Android розробки.*

Наукова новизна*. Визначено ключові вимоги до мобільного додатку для вивчення алгоритмів сортування. Здійснено аналіз застосунків для навчання, включаючи їх сильні та слабкі сторони. Опрацьовані основні проєктні рішення, вибрані інструменти та методики для розробки Android-застосунку у сфері дистанційної освіти. Встановлено конкретну методологію для розробки програмного забезпечення. Складено діаграму прецедентів для кращого розуміння взаємодії користувачів із системою. Розроблено інтуїтивний мобільний інтерфейс для взаємодії користувача з матеріалом. Розроблено та апробовано навчальний андроїд застосунок з теми «Сортування вставками» для дистанційного навчального курсу «Алгоритми і структури даних».*

Розроблений застосунок можна використовувати для будь-яких професійних дисциплін під час вивчення алгоритмів сортування.

Висновки. *Розроблене програмне забезпечення імплементоване в дистанційний курс освітньої компоненти «Алгоритми і структури даних». Програмний продукт є результатом автоматизації навчального процесу для дистанційної форми навчання. Саме тому він покриває основні потреби студентів та викладачів у навчальному процесі. Результати розробки впроваджено в навчальний процес Полтавського університету економіки і торгівлі.*

Ключові слова: *діаграма прецедентів, фреймворк, мобільний застосунок.*

DEVELOPMENT OF AN EDUCATIONAL ANDROID APPLICATION ON THE TOPIC “SORTING BY INSERTIONS” OF THE DISTANCE LEARNING COURSE “ALGORITHMS AND DATA STRUCTURES”

Abstract. *The modern world of technology requires more and more effective teaching methods, especially in the field of programming and computer science. Meeting these requirements, there is a need to create interactive Android applications that are indispensable assistants for distance and mixed learning formats.*

The purpose of the work *is to develop a mobile educational application for Android on the topic “Sorting by insertions”.*

Methodology. *Android application software is developed using the Kotlin programming language and using the Android SDK to create a user interface. The list of methods includes the design of mobile educational applications, in particular, UX/UI design methods. As for technical tools, Android Studio was used for development, Git for version control, as well as libraries and frameworks specific to Android development.*

Scientific novelty. *The key requirements for a mobile application for studying sorting algorithms have been determined. An analysis of learning applications, including their strengths and weaknesses, was carried out. The main design solutions, selected tools and methods for developing an Android application in the field of distance education were developed. A specific methodology for software development has been established. Made a case diagram for better understanding of user interaction with the system. An intuitive mobile interface for user interaction with the material has been developed. An educational Android application on the topic “Sorting by insertions” was developed and tested for the distance learning course “Algorithms and data structures”.*

The developed application can be used for any professional disciplines when studying sorting algorithms.

Conclusions. *The developed software is implemented in the distance course of the educational component “Algorithms and data structures”. The software product is the result of the automation of the educational process for distance education. That is why it covers the basic needs of students and teachers in the educational process. The results of the development are implemented in the educational process of the Poltava University of Economics and Trade.*

Key words: *case diagram, framework, mobile application.*

Постановка проблеми в загальному вигляді та її зв'язок з важливими науковими чи практичними завданнями. У контексті модернізації освітнього процесу, актуальним є завдання адаптації освітньої системи до сучасних вимог, з метою підготовки фахівців, які будуть конкурентоспроможні в сучасному світі. Однією з можливих відповідей на це завдання є розробка мобільних навчальних додатків для Android.

Основним плюсом мобільних навчальних додатків [1-6], в тому числі для платформи Android, є здатність надавати освітні послуги незалежно від географічного розташування користувача, допоки є доступ до Інтернету. Це надає студентам гнучкість у виборі оптимального для них формату навчання, дозволяючи адаптувати процес освіти до власного ритму життя та специфічних потреб. Крім того, мобільне навчання може виявитися також економічно вигідним.

Аналіз останніх досліджень і публікацій. В [1, 2, 4] розглянуті інтерактивні технології навчання та їх застосування у вищій школі. Використання тренажерів при підготовці здобувачів освіти напряму «Комп'ютерні науки» описано в [3-5].

Постановка завдання. В даній публікації пропонується проєктування та реалізація мобільного навчального застосунку для платформи Android.

Виклад основного матеріалу дослідження. Для розробки навчального андроїд-застосунку були використані наступні технології:

– Програма для розробки концептів дизайну Photoshop.
– Набір для розробки програмного забезпечення Android SDK.
– Реляційна база даних sqlite для збереження інформації локально.
– Мова програмування Kotlin.
– Середовище розробки Android Studio.
– Архітектура MVVM для мобільного додатка.

Для створення мобільного додатку було обрано методологію водоспаду. Каскадна модель є послідовною методологією розробки, де кожен наступний етап починається після завершення попереднього (рис. 1). Основною перевагою каскадної моделі є її структурованість і простота [7].

Рис. 1. Модель розробки за каскадною методологією

На першому етапі визначено основні вимоги до розробки: інтерактивність, поєднання теоретичного матеріалу з практичними вправами. Для забезпечення чіткості і наочності, основні правила та ключові моменти супроводжуються конкретними прикладами. Для мотивації та кращого розуміння своїх досягнень користувачам важливо мати можливість відстежувати свій прогрес. Статистика є важливим інструментом самоаналізу та самовдосконалення.

Враховуючи вищеназвані вимоги, була запропонована така структура додатку:

1. Головна сторінка. Доступ до основних розділів додатку та відображення статистики користувача.
2. Розділ теорії. Докладний опис алгоритму, приклади код та інші матеріали, які допоможуть користувачеві вивчити алгоритм.
3. Модуль тестів. Набір завдань, спрямованих на перевірку розуміння алгоритму користувачем.
4. Модуль статистики. Візуалізація даних про активність та успішність користувача в додатку [8].

На рис. 2 представлена діаграма прецедентів роботи з системою. Прецедент у цьому контексті є відображенням того, що додаток може виконувати, або тих дій, які користувачі можуть здійснювати за допомогою нього. Така діаграма допомагає краще зрозуміти та відобразити взаємодію між користувачами та додатком [9, 10].

Розглянемо створений програмний продукт.

При відкритті додатка користувача вітає «Splash Screen» – перший екран, що служить мостом між запуском програми та її основною функціональністю. Цей екран не лише забезпечує візуальне підтвердження запуску додатка, але і є своєрідною візитівкою програми.

За першого запуску, після «Splash Screen», користувач відразу зустрічається з привітальним екраном, при наступних запусках додатка користувач потрапляє відразу на головний екран.

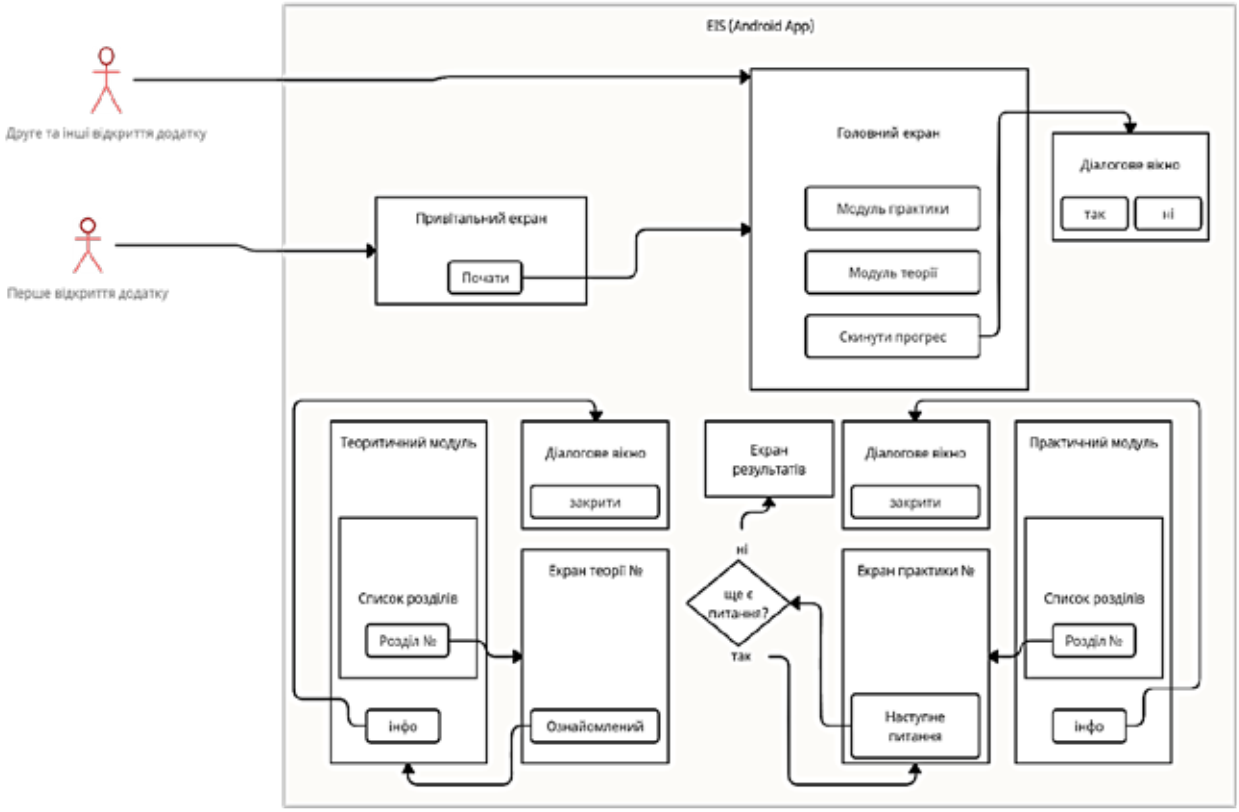

Рис. 2. Діаграма прецедентів

Головний екран (рис. 3) включає такі елементи: Порада для вивчення, Теорія, Практика.

Порада до вивчення – це корисне повідомлення, яке генерується при кожному запуску додатка, і надає поради для ефективного вивчення теми.

Розділ "Теорія та практика – центральний компонент, що дозволяє користувачеві обрати тематичний модуль для вивчення або практики.

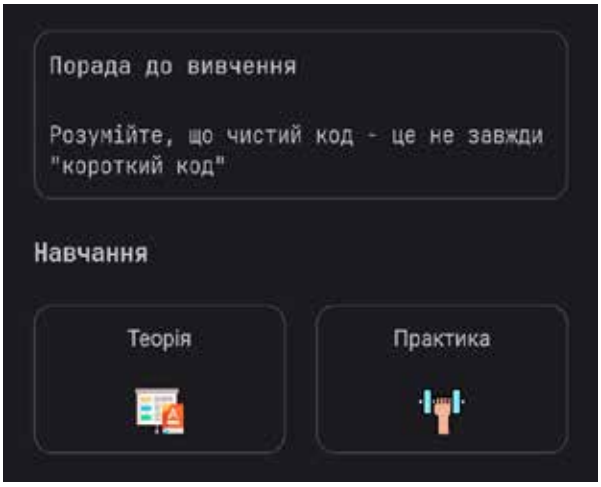

Рис. 3. Головний екран

По завершенню навчання користувач може проглянути статистику щодо прогресу навчання.

Статистика (рис. 4) та Прогрес (рис. 5) – візуалізація досягнень користувача. Тут відображено прогрес вивчення, а також графік правильних відповідей на тести, організований по днях тижня.

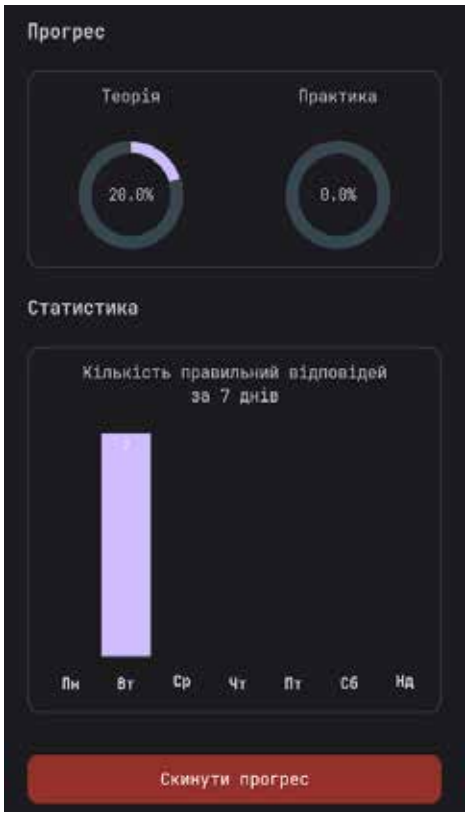

Рис. 4. Статистика

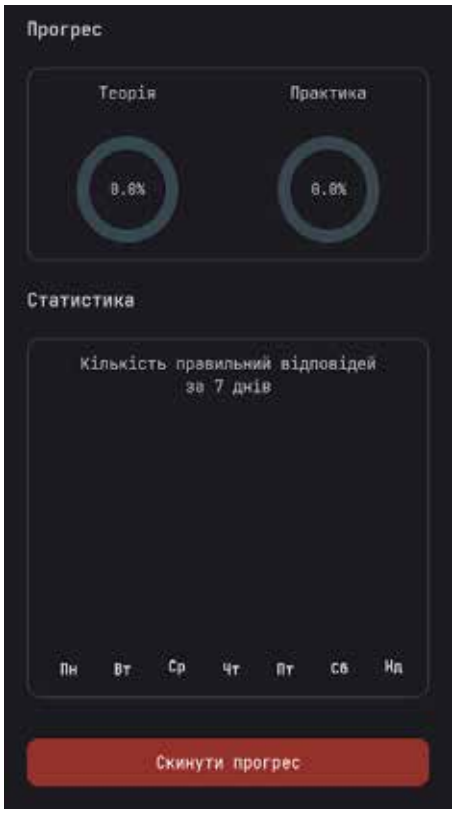

Рис. 5. Прогрес користувача

Користувач має можливість скинути всю статистику за допомогою кнопки "Стерти прогрес". Після її натиснення з'являється діалогове вікно (рис. 6), що попереджає про втрату всіх зібраних даних.

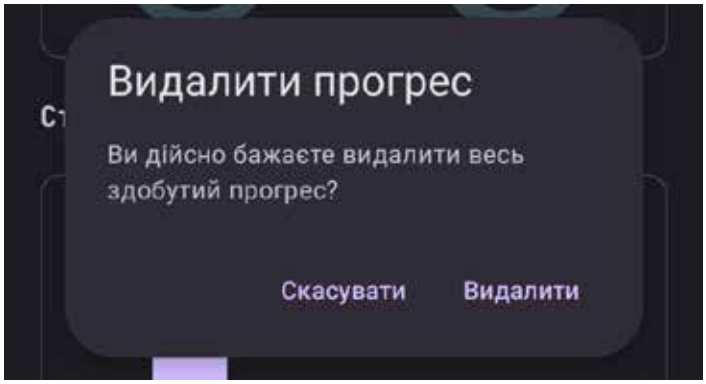

Рис. 6. Попередження про видалення даних

У розділі "Теорія" розміщена кнопка FAQ, натиснувши на яку, користувачу відкривається діалогове вікно (рис. 7) із загальною інформацією про додаток. Також тут представлений список розділів теорії (рис. 8).

Кожен конкретний розділ теорії включає в себе докладний опис теми, а також кнопку для відстеження особистого прогресу користувача в опануванні матеріалу (рис. 9).

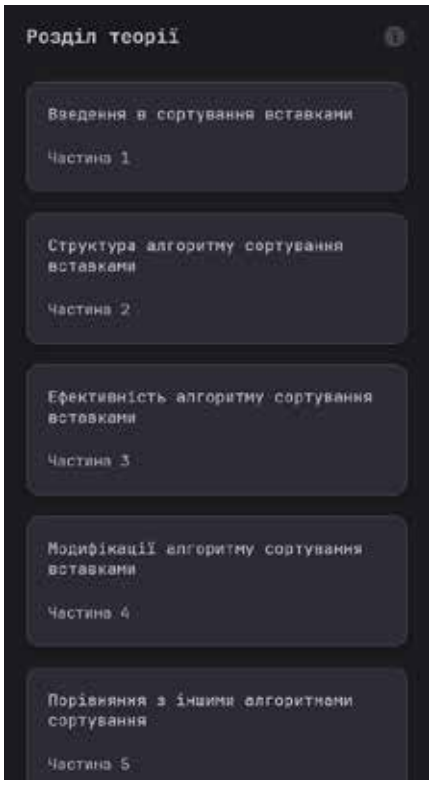

Рис. 7. Розділ Теорія

Рис. 8. Інформація про застосунок

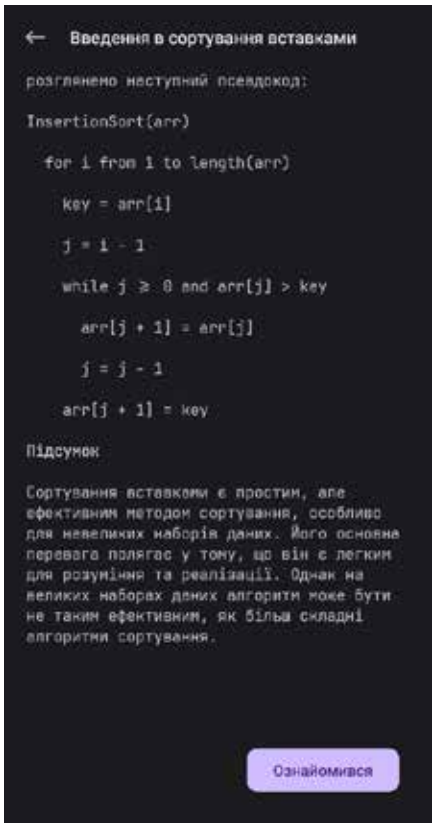

Рис. 9. Частина процесу тестування із поясненням

У розділі "Практика" також знаходиться кнопка FAQ, відкриваючи яку користувач знову зустрічається з діалоговим вікном (рис. 10) та отримує інформацію про додаток. Окрім того, тут представлено список практичних тестів, що відповідають темам теорії (рис. 11).

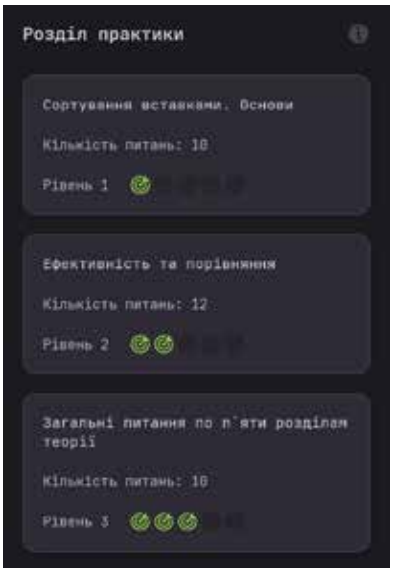

Рис. 10. Практичні тести

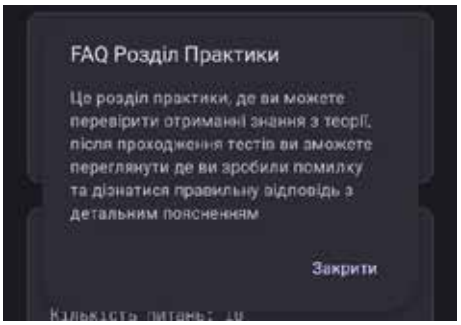

Рис. 11. Інформація про застосунок

Кожне завдання в тестах містить питання, три варіанти відповіді та кнопку "Далі" (рис. 12).

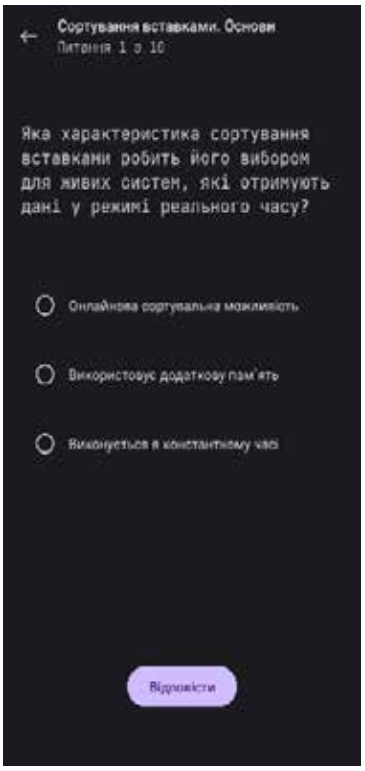

Рис. 12. Частина процесу тестування

Після завершення тестування користувач отримує звіт із вказівкою на правильні та неправильні відповіді. Додатково до кожної помилкової відповіді надається пояснення з правильним аргументуванням (рис. 13).

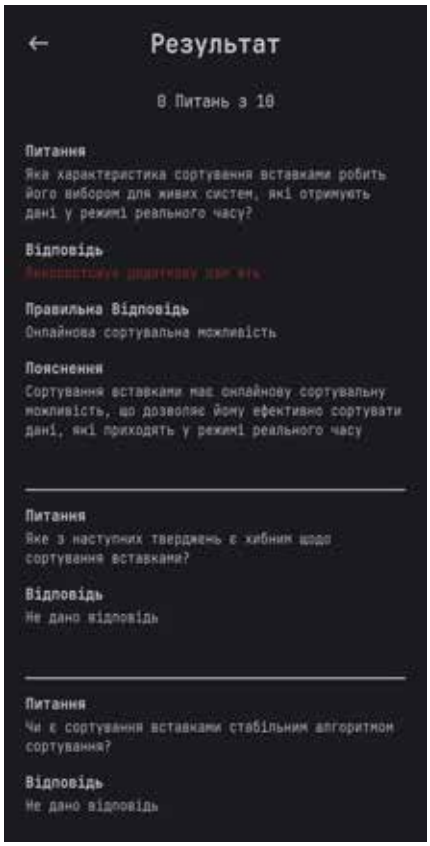

Рис. 13. Результати тестування із поясненням

Висновки з даного дослідження та перспективи подальших розвідок у даному напрямі. Ключовою особливістю застосунку є його підхід до навчання. Об'єднання теоретичного матеріалу з практичними вправами дозволяє користувачам глибше розуміти концепції алгоритму та набути практичних навичок його застосування. Статистика в застосунку не лише відображає рівень досягнень користувача, але й служить важливим інструментом самоаналізу та самовдосконалення, допомагаючи користувачу максимально ефективно використовувати свій час та ресурси.

Андроїд-застосунок протестовано та впроваджено в навчальний процес Полтавського університету економіки та торгівлі для здобувачів освіти спеціальності «Комп'ютерні науки». У подальшому планується його удосконалення шляхом розширення функціоналу для викладача. Зокрема, можливість коригувати чи додавати завдання.

Список використаних джерел:

1. Волкова, Н.П. Інтерактивні технології навчання у вищій школі: навчально-методичний посібник. Дніпро: Університет імені Альфреда Нобеля. 2018. 360 с.

2. Доценко, Н. Застосування навчальних комп'ютерних інтерактивних тренажерів здобувачами вищої освіти інженерних спеціальностей в умовах інформаційно-освітнього середовища. *Педагогічні науки: теорія, історія, інноваційні технології.* 2018. № 2(76). С. 118–128.

3. Черненко, О.О., Чілікіна, Т.В., Ольховська, О.В. Розробка та використання навчальних тренажерів при підготовці фахівців напряму «Комп`ютерні науки». *International scientific and practical conference "Mathematics, physics, mechanics, astronomy, computer science and cybernetics: issues of productive interaction": conference proceedings, Yuly 9-10. 2021. Wloclawek, Republic of Poland: "Baltija Publishing"*. 2021. С. 55-59.

4. С.В. Гаркуша, О.О. Черненко, О.П. Кошова, І.В. Субота, А.І. Литвиненко РОЗРОБКА ПРОГРАМИ-ТРЕНАЖЕРУ ДИСТАНЦІЙНОГО НАВЧАЛЬНОГО КУРСУ «ОСНОВИ НАУКОВИХ ДОСЛІДЖЕНЬ В ІНФОРМАТИЦІ». *Збірник наукових праць Національного університету кораблебудування імені адмірала Макарова*. 2023. № 1. С. 165-175.

5. O. Chernenko, N. Rudenko, D. Bondar. Development of simulator software on the topic "Normal algorithms" of the distance learning course "Theory of Algorithms" *Центральноукраїнський науковий вісник. Технічні науки: зб. наук. пр.* Кропивницький : ЦНТУ. 2023. Вип. 7(38). Ч. 1. С. 3-9.

6. Khandii, O., Derzhak, N. Digitalization of higher education and features of interactive learning. *The Second Special Humanitarian Issue of Ukrainian Scientists. European Scientific e-Journal.* Ostrava: Tuculart Edition. 2022. 3 (18), 97-104.

7. Victor L. de Oliveira. On the adoption of kotlin on android development: a triangulation study. *27th IEEE International Conference on Software Analysis, Evolution, and Reengineering* (SANER 2020).

8. Luca Ardito, Riccardo Coppola, Giovanni Malnati, Marco Torchiano. Effectiveness of Kotlin vs. Java in android app development tasks. *Information and Software Technology.* Volume 127, November 2020, P. 106374.
9. Josh Skeen, David Greenhalgh. Kotlin Programming: The Big Nerd Ranch. 2018. 1005 p.
10. Greg Lim. Beginning Android Development With Kotlin Kindle Edition. 2022. 1243 p.

References:

1. Volkova, N.P. (2018). Interaktyvni tekhnolohii navchannia u vyshchii shkoli: navchalno-metodychnyi posibnyk [Interactive learning technologies in higher education: educational and methodological guide]. Dnipro: Universytet imeni Alfreda Nobelia. 360 p. [in Ukrainian].
2. Dotsenko N. (2018). Zastosuvannia navchalnykh kompiuternykh interaktyvnykh trenazheriv zdobuvachamy vyshchoi osvity inzhenernykh spetsialnostei v umovakh informatsiino-osvitnoho seredovyshcha [The use of educational computer interactive simulators by students of higher education in engineering specialties in the conditions of an informational and educational environment]. *Pedahohichni nauky: teoriia, istoriia, innovatsiini tekhnolohii – Pedagogical sciences: theory, history, innovative technologies,* № 2(76), 118–128. [in Ukrainian].
3. Chernenko, O.O., Chilikina, T.V., Olkhovska, O.V. (2021). Rozrobka ta vykorystannia navchalnykh trenazheriv pry pidhotovtsi fakhivtsiv napriamu «Komp`iuterni nauky» [Development and use of educational simulators in the training of specialists in the field of "Computer Sciences"]. *International scientific and practical conference ``Mathematics, physics, mechanics, astronomy, computer science and cybernetics: issues of productive interaction``: conference proceedings*, Yuly 9-10. 2021. Wloclawek, Republic of Poland: ``Baltija Publishing``. 55-59. [in Ukrainian].
4. S.V. Harkusha, O.O. Chernenko, O.P. Koshova, I.V. Subota, A.I. Lytvynenko (2023). ROZROBKA PROHRAMY-TRENAZHERU DYSTANTSIINOHO NAVCHALNOHO KURSU «OSNOVY NAUKOVYKH DOSLIDZHEN V INFORMATYTSI» [DEVELOPMENT OF THE TRAINER PROGRAM FOR THE DISTANCE EDUCATIONAL COURSE "BASES OF SCIENTIFIC RESEARCH IN COMPUTER SCIENCES"]. *Zbirnyk naukovykh prats Natsionalnoho universytetu korablebuduvannia imeni admirala Makarova – Collection of scientific works of the Admiral Makarov National Shipbuilding University*. № 1. P.165-175. [in Ukrainian].
5. O. Chernenko, N. Rudenko, D. Bondar. (2023). Development of simulator software on the topic "Normal algorithms" of the distance learning course "Theory of Algorithms" *Tsentralnoukrainskyi naukovyi visnyk. Tekhnichni nauky: zb. nauk. pr. – Central Ukrainian scientific bulletin. Technical sciences: coll. of science Kropyvnytskyi Ave.: Central Technical University.* Issue 7(38). Part 1. P. 3-9. [in Ukrainian].
6. Khandii, O., Derzhak, N. (2022). Digitalization of higher education and features of interactive learning. *The Second Special Humanitarian Issue of Ukrainian Scientists. European Scientific e-Journal.* Ostrava: Tuculart Edition. 3 (18), 97-104.
7. Victor L. de Oliveira (2020). On the adoption of kotlin on android development: a triangulation study. *27th IEEE International Conference on Software Analysis, Evolution, and Reengineering* (SANER).
8. Luca Ardito, Riccardo Coppola, Giovanni Malnati, Marco Torchiano (2020). Effectiveness of Kotlin vs. Java in android app development tasks. *Information and Software Technology.* Volume 127, P. 106374.
9. Josh Skeen, David Greenhalgh (2018). Kotlin Programming: The Big Nerd Ranch. 1005 p.
10. Greg Lim (2022). Beginning Android Development With Kotlin Kindle Edition. 1243 p.

УДК 004
DOI https://doi.org/10.32689/maup.it.2023.5.6

Назарій КУЧЕР-САВІНСЬКИЙ
магістрант кафедри Інформаційних Систем та Технологій, Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського», Київ, Берестейський проспект, 37, індекс 03056 (zxcaaa12@outlook.com)

Nazarii KUCHER-SAVINSKYI
Master's Student at the Department of Information Systems and Technologies, National Technical University of Ukraine "Ihor Sikorskyi Kyiv Polytechnic Institute", Kyiv, 37, Beresteyskyi Ave, postal code 03056 (zxcaaa12@outlook.com)

Бібліографічний опис статті: Кучер-Савінський, Н. (2023). Система аналізу вигідності контрактів у сфері засобів масової інформації. *Інформаційні технології та суспільство,* 5 (11), 43–49. DOI: https://doi.org/10.32689/maup.it.2023.5.6

Bibliographic description of the article: Kucher-Savins'kyy, N. (2023). Systema analizu vyhidnosti kontraktiv u sferi zasobiv masovoyi informatsiyi [System for contract profit analysis in the media sector]. *Informatsiini tekhnolohii ta suspilstvo – Information technology and society*, 5 (11), 43–49. DOI: https://doi.org/10.32689/maup.it.2023.5.6

СИСТЕМА АНАЛІЗУ ВИГІДНОСТІ КОНТРАКТІВ У СФЕРІ ЗАСОБІВ МАСОВОЇ ІНФОРМАЦІЇ

Анотація. *У цій статті детально розглядається тема прийняття рішень, зосереджуючись на їх різновидах та методах, етапах та алгоритмах. Подано визначення поняття рішення і далі переходить до аналізу різних підходів та методів, які можуть бути використані в процесі прийняття рішень. Особлива увага приділяється етапам алгоритмів для раціонального прийняття рішень. Розкрито, як ці етапи можуть сприяти вирішенню складних задач і знаходженню оптимальних рішень у різноманітних ситуаціях, а також, як знання цих етапів може допомогти у модифікації конкретного алгоритму. В статті також наведено конкретний приклад: задачу вибору найбільш вигідного рекламного контракту у сфері засобів масової інформації. Для її розв'язання було застосовано метод лінійної згортки, що дозволило оптимізувати вибір, враховуючи різні критерії. Докладно описується актуальність вибору даного методу, кроки застосування цього методу, адаптуючи його до специфічних умов та встановлених критеріїв. Крім того, в статті обговорюються особливості використання методу лінійної згортки при збільшенні кількості критеріїв та змінних. Аналізується, як зміна цих параметрів впливає на процес прийняття рішень та на якість кінцевих результатів. Цей аналіз важливий для глибшого розуміння потенціалу та обмежень методу лінійної згортки в управлінському рішенні, виявлення недоліків та переваг цього методу, а також сферу ефективного застосування і дозволяє краще порівняти цей метод з аналогами. Загалом, стаття надає цінну інформацію для фахівців у сфері менеджменту та управління, автоматизації прийняття рішень, а також для тих, хто цікавиться теорією та практикою прийняття рішень. Вона аналізує процес прийняття рішень та демонструє, як теоретичні знання та методи прийняття рішень можуть бути застосовані на практиці для вирішення конкретних бізнес-задач.*

Ключові слова: *алгоритм, прийняття рішень, проблема, системний підхід, процес.*

SYSTEM FOR CONTRACT PROFIT ANALYSIS IN THE MEDIA SECTOR

Abstract. *This article examines the topic of decision-making in detail, focusing on their varieties and methods, stages and algorithms. The definition of the concept of decision is given and then it goes to the analysis of various approaches and methods that can be used in the decision-making process. Special attention is paid to the stages of algorithms for rational decision-making. It is revealed how these stages can contribute to solving complex problems and finding optimal solutions in various situations, as well as how knowledge of these stages can help in modifying a specific algorithm. The article also provides a specific example: the task of choosing the most profitable advertising contract in the field of mass media. To solve it, the linear convolution method was used, which made it possible to optimize the choice, taking into account various criteria. The relevance of choosing this method, the steps of applying this method, adapting it to specific conditions and established criteria are described in detail. In addition, the article discusses the features of using the linear convolution method when increasing the number of criteria and variables. It is analyzed how the change of these parameters affects the decision-making process and the quality of the final results. This analysis is important for a deeper understanding of the potential and limitations of the linear convolution method in management decision-making, identifying the disadvantages and advantages of this method, as well as the scope of effective application and allows a better comparison of this method with analogues. In general, the article provides valuable information for professionals in the field of management and control, automation of decision-making, as well as for those interested in the theory and practice of decision-making. It analyzes the decision-making process and demonstrates how theoretical knowledge and decision-making methods can be applied in practice to solve specific business problems.*

Key words: *algorithm, decision-making, problem, system approach, process.*

Постановка проблеми. Недостатня ефективність аналізу вигідності контрактів у сфері медіа та ЗМІ обумовлюється неоднозначністю критеріїв оцінки ефективності угод, відсутністю механізмів прогнозування ризиків та недооціненим впливом контрактів на стратегічний розвиток підприємства, що може обмежувати здатність компаній масової інформації до оптимального управління ресурсами та фінансами. Для того, щоб краще зрозуміти прогалини в такій системі розглянемо основні переваги та недоліки. До переваг відноситься: об'єктивність (система дозволяє проводити аналіз на основі чітких критеріїв, що сприяє об'єктивності оцінки контрактів); ефективність прийняття рішень (забезпечує надійний аналіз, який полегшує процес прийняття рішень щодо укладання чи продовження контрактів); оптимізація ресурсів (дозволяє виявляти найбільш вигідні контракти, сприяючи оптимізації використання ресурсів). До недоліків такої системи відноситься: суб'єктивність визначення критеріїв (може виникнути проблеми суб'єктивності при визначенні критеріїв оцінки ефективності контрактів); недостатнє урахування ризиків (система може недостатньо ураховувати потенційні ризики, що може призвести до неочікуваних втрат); бракування динамічності (відсутність адаптивності до змін в ринкових умовах може ускладнити аналіз в контексті швидко змінюваних технологічних та економічних трендів). Для вирішення вищезазначених проблем в роботі розглядається класифікація за критеріями та алгоритм прийняття рішень.

Аналіз останніх досліджень і публікацій. Дослідження проблем модифікації алгоритму прийняття рішень, його методів, розробки завдань та реалізації планів знайшли відповідне відображення в наукових працях учених – економістів: Бондар Н. [2], Копистинська І. [4], Пушкар О., Грабовський Є. [8], Патаракін Е., Буров В., Реморенко І. [11]. Однак питання необхідності модифікації алгоритму прийняття рішень, його методів, розробки завдань та реалізації планів в умовах формування ринкової економіки опрацьовані недостатньо і потребують подальшого дослідження.

Постановка завдання. Метою дослідження є модифікація алгоритму прийняття рішень на основі методу лінійної згортки з використанням критеріїв, що може використовуватися для ЗМІ при укладанні рекламного контракту.

Виклад основного матеріалу дослідження. Кожна людина постійно шукає відповіді на безліч питань у своєму житті. Рішення можуть бути широкими або невеликими, важливими або повсякденними. В будь-якому випадку прийняття рішень означає вибір найбільш відповідного варіанту з наявних альтернатив. Людина, приймаючи рішення, бере на себе ризики та відповідальність за можливі наслідки. Тому процес прийняття будь-якого рішення завжди є важливим. У побутових рішеннях особливість полягає в тому, що особа, яка приймає ці рішення, часто виконує їх особисто.

Робота менеджера вимагає від особи планування, відповіді на різноманітні запитання та вирішення завдань різного рівня складності. Проте управлінські рішення відрізняються від звичайних за кількома основними аспектами. Вони мають специфічні цілі, які прив'язані до функціонування організації та суттєво впливають на роботу багатьох людей. Тому ступінь відповідальності в управлінських рішеннях значно вищий, порівняно з особистими рішеннями, які стосуються лише однієї особи. Рішення можна класифікувати за кількома критеріями [1]:

1. За терміном реалізації: короткострокові, середньострокові та довгострокові рішення.
2. За ступенем впливу на діяльність організації: стратегічні, тактичні та оперативні.
3. За методом або алгоритмом прийняття: евристичні та послідовні рішення.
4. За провідною функцією: організаційні, координуючі, мобілізуючі, регулюючі та контролюючі. Також вони можуть бути одиничними чи повторюваними за частотою прийняття.
5. За кількістю учасників: індивідуальні, групові та корпоративні.
6. За ступенем формалізації процесу: алгоритмізовані, структуровані та контурні рішення.

Кожен менеджер володіє своїм набором інструментів для вирішення виробничих завдань. Зазвичай методи пошуку варіантів поділяють на імпульсивні, раціональні, інтуїтивні та евристичні. У менеджменті переважною є підтримка прийняття рішень, заснована на раціональному підході. Кожен керівник, користуючись власним досвідом, розвиває свій унікальний алгоритм для пошуку рішень. Дерево прийняття рішень – це один з найпопулярніших логічних методів для аналізу різних варіантів розвитку ситуацій. Ця методика передбачає створення візуальної схеми, де кожне питання має можливі відповіді та прогнозується можлива реакція на кожну з цих відповідей. Даний метод побудований на ланцюжках причинно-наслідкових зв'язків, при цьому він спрямований на виключення впливу емоцій або спонтанних рішень. Він застосовується в таких галузях як машинне навчання, логіка та статистика. Однак його основне обмеження полягає в тому, що не завжди можна передбачити всі можливі наслідки. Крім того, всі методи вирішення завдань можуть бути класифіковані за числом учасників у процесі прийняття рішень [6].

Кожній особі потрібно приймати безліч рішень у щоденному житті. Однак для менеджера це стає ще складнішим завданням через значну кількість професійних вирішуваних задач, що він зустрічає щодня.

Якість кожного прийнятого рішення має значення, адже вона може суттєво вплинути на подальший розвиток. Тому необхідно вміло й грамотно приймати рішення. Розгляньмо 10 кроків алгоритму для раціонального управлінського рішення та розглянемо, як вони можуть допомогти знайти оптимальний вихід з будь-якої ситуації [5].

Етап 1. Аналіз проблеми. Для ефективного розв'язання будь-якої задачі, ключовим є чітке визначення проблеми, яка потребує вирішення. Саме у формулюванні цієї проблеми фахівці бачать можливості знайти вихід. Традиційно це формулювання проводиться у негативному контексті, з фокусом на тому, що не влаштовує організацію в даній ситуації.

Етап 2. Збір і аналіз інформації. Другий етап алгоритму прийняття рішень – це дослідницький процес. Для знаходження правильного варіанту розв'язання ситуації необхідно зібрати якнайбільше інформації. Це може включати статистичні дані, експертні оцінки, результати досліджень та звіти. На цьому етапі важливо оцінити зовнішні й внутрішні фактори, що впливають на ситуацію. У випадку індивідуального прийняття рішення керівник може також користуватися своїм власним досвідом у схожих обставинах. При групових методах зазвичай кожен експерт вносить свій внесок у розуміння контексту, в якому буде прийматися рішення.

Етап 3. Розробка умов і критеріїв прийняття рішення. Будь-який варіант повинен враховувати різні фактори й наслідки. Тому алгоритм прийняття рішень обов'язково включає етап, на якому відбувається оцінка можливих ризиків при тому чи іншому розвитку подій. Умови в першу чергу можуть відрізнятися за обсягом наявної інформації: визначені та невизначені. У першому випадку рішення приймати легше, бо це задача з усіма відомими, але частіше менеджерам доводиться стикатися з другою групою умов.

Етап 4. Формулювання бажаного рішення та постановка мети. Наступний найважливіший етап алгоритму прийняття рішень – це визначення мети. Перш ніж починати діяти, потрібно зрозуміти, чого хочеться досягти. Спочатку треба уявити собі мету, яку можна досягти при максимально сприятливому розвитку подій. Це своєрідний ідеал. А далі це бажане стан необхідно скоригувати з урахуванням наявних умов. Постановка мети – найважливіший етап будь-якої діяльності. У менеджменті вважається, що мета повинна бути досяжною, актуальною для виконавців та організації, обмеженою в часі, вимірної й конкретною.

Етап 5. Оцінка альтернатив. У багатьох випадках прийняття рішень включає розгляд декількох варіантів. Різні моделі алгоритмів прийняття рішень мають свої методи порівняння альтернатив. Це може бути кількісне порівняння, компаративний аналіз або використання методу експертних оцінок. Кожен з варіантів потребує аналізу. Для порівняння може використовуватися стандарт, що допомагає виділити найважливіші аспекти та уникнути дрібниць. Цей стандарт базується на критеріях, визначених на етапі 3.

Етап 6. Прийняття рішення. Отримане рішення має бути втілене у новий виробничий цикл. Алгоритм прийняття рішення передбачає остаточний вибір оптимального варіанту дій. Це включає процес повідомлення керівництва та виконавців про прийняте рішення. У багатьох організаціях на цей випадок існують встановлені процедури для інформування та документообігу.

Етап 7. Формулювання завдань. Глибокий та продуманий процес розробки алгоритму прийняття рішень включає етап постановки завдань. Кожна мета – це стратегія, і для досягнення цієї мети необхідна тактика. Конкретні завдання виступають як програма досягнення мети, розбиваючи її на окремі кроки. Виконавцям потрібно пропонувати конкретні завдання, які допоможуть досягти загальної мети. Кожен співробітник отримує своє завдання у формі конкретних дій, які йому необхідно виконати. Постановка завдань передбачає відповідь на три ключові питання: чому, що і як. Співробітник повинен розуміти, чому це завдання важливе для нього, що саме від нього очікується, і яким чином це потрібно зробити. Задача повинна відповідати тим же критеріям, що і мета. Вона має бути конкретною, реалістичною, досяжною та обмеженою за часом виконання [12].

Етап 8. Впровадження. Після ухвалення рішення настає фаза його впровадження, яка є однією з ключових складових виробничого процесу. На цьому етапі важливо забезпечити інформаційну та організаційну підтримку для втілення прийнятого рішення. Підходи до реалізації рішень можуть значно відрізнятися в залежності від конкретних особливостей діяльності організації та її галузі. На цьому етапі виконавці повинні чітко виконувати поставлені завдання, дотримуючись встановлених стандартів і термінів.

Етап 9. Контроль за виконанням рішення. Під час впровадження рішення менеджер повинен здійснювати постійний контроль за цим процесом. Його завдання полягає в оцінці відповідності виконання запланованим цілям, вчасному виконанні завдань і раціональному використанні ресурсів. Менеджер на цьому етапі має готовність коригувати завдання, якщо вони втрачають відповідність загальним цілям у зв'язку з будь-якими обставинами.

Етап 10. Оцінка ефективності та звітність. Після того, як рішення буде виконано, менеджеру необхідно оцінити вірність його виконання і якість самого рішення. Наскільки воно дозволило усунути наявну проблему, з якою і починався процес прийняття рішення? Для оцінки керівник застосовує критерії, що задаються на етапі 3. Після того як зроблений остаточний аналіз рішення та ефективності його реалізації, зазвичай менеджер складає звіт, у якому фіксує позитивні й негативні результати процесу прийняття і виконання рішення [9].

Рішення приймаються через динамічний та внутрішньо пов'язаний процес, що включає різні функції прийняття рішень. Формулювання рішення передбачає два ключових етапи. По-перше, це усвідомлення необхідності прийняття рішення, а по-друге, проведення діагностики та аналізу ситуації. Процес прийняття рішень розпочинається з чіткої постановки завдання і завершується в момент виконання цієї мети, коли прийняте рішення впроваджується [3].

Так, потреба в ухваленні рішення може виникнути через проблему або можливість. Проблема може виникнути, коли досягнення організацією поставлених цілей ускладнюється непритомністю досягнення певних результатів, що вимагають удосконалення діяльності. Можливість натомість означає, що менеджери вбачають потенціал для покращення організаційної діяльності, що дозволить перевищити поточні цілі. Усвідомлення проблеми або можливості становить перший етап процесу ухвалення рішень. Це вимагає дослідження зовнішнього та внутрішнього середовища для виявлення несподіваних відхилень та визначення перспектив, які варто враховувати керівництву. Цей процес схожий на військовий розвідку: менеджери аналізують навколишній світ, щоб з'ясувати, чи досягає організація своїх цілей [8].

Пошук альтернатив під час прийняття рішень – це процес вивчення зовнішнього та внутрішнього оточення організації для отримання необхідної інформації. Ця інформація використовується для створення набору можливих варіантів рішень, які, на цей час, здається, можуть сприяти досягненню поставленої мети або завдання. Пропоновані альтернативи рішень мають бути реалістичними, тобто відповідати умовам зовнішнього та внутрішнього середовища організації, які в теорії прийняття рішень називаються обмеженнями. Обмеження – це умови, які впливають на досягнення цілей організації та визначаються зовнішнім середовищем та ресурсами організації.

Оскільки процес досягнення цілей організації більшою мірою залежить від зовнішнього середовища, обмеження можуть обмежувати можливості внутрішнього середовища організації [4]. Подальший аналіз і контроль дозволяють переконатися, що обране рішення відповідає вимогам, які поставило керівництво й сприятиме досягненню очікуваних результатів, що визначили початок процесу ухвалення рішення. Під час оцінки, менеджер повинен аналізувати, як його рішення виконується і наскільки ефективно воно спрямоване на досягнення поставлених цілей. Зворотний зв'язок відіграє ключову роль у процесі впровадження рішень, оскільки прийняття рішень – це послідовний і неперервний процес. Завдяки зворотному зв'язку, до керівництва надходять відомості, які можуть викликати необхідність нових дій. Далі наведемо алгоритм прийняття рішень методом лінійної згортки для вибору засобу масової інформації для укладання рекламного контракту. Схему даного алгоритму наведено на рис. 1. [7].

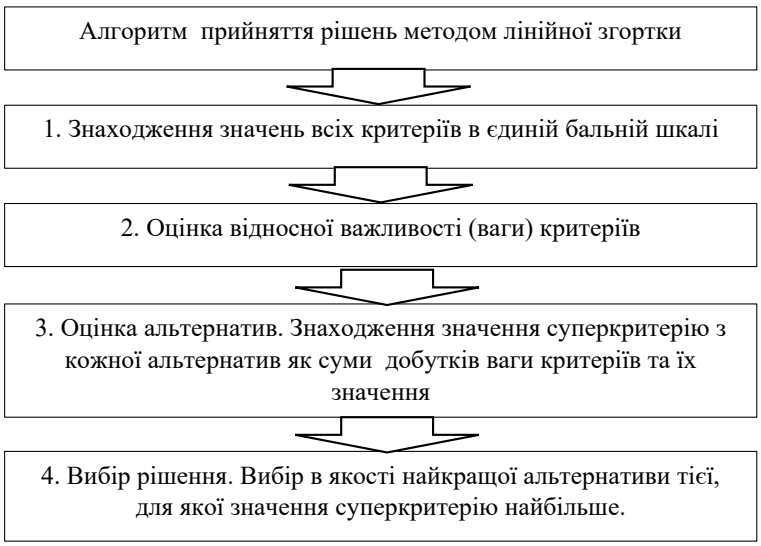

Рис. 1. Алгоритм прийняття рішень методом лінійної згортки [7]

Проблема полягає у виборі найкращого ЗМІ для розміщення рекламного контракту. Компанія розглядає пропозиції щодо створення та розміщення реклами від семи ЗМІ: A, B, C, E, H, I, K. Тому необхідно розглянути питання вибору ЗМІ для укладення рекламного контракту [2].

У даному контексті, задача прийняття рішення стосується вибору ЗМІ для укладення рекламного контракту. Однак, визначення оптимальної альтернативи у цьому випадку ускладнюється тим, що критерії порівняння не є однозначно визначеними. Пропозиції ЗМІ мають різноманітні параметри, такі як ціна, якість реклами, розмір аудиторії тощо. Це означає, що система переваг альтернатив на цей час не є повною, і неможливо однозначно порівняти альтернативи між собою за відсутності чітких критеріїв для оцінки [10]. Знаходження значень всіх критеріїв в єдиній бальній шкалі. Для зіставлення впливу різних критеріїв на загальну якість альтернативи, використовується єдина бальна шкала для вимірювання їх значень. Якщо значення критерію виміряно в абсолютній шкалі, його поділяють на інтервали, і кожному інтервалу присвоюється певна кількість балів для оцінки значення критерію. Кількість інтервалів на абсолютній шкалі та їх діапазони, як правило, визначаються суб'єктивно залежно від конкретного контексту аналізу.

Розглянемо значення критеріїв, що наведені в табл. 1 та 2 [4].

Таблиця 1

Значення критеріїв проблеми вибору ЗМІ для укладання рекламного контракту (розроблено автором)

Критерії	Альтернативи						
	A	**B**	**C**	**E**	**H**	**I**	**K**
Ціна контракту *P*, тис. дол.	60	65	60	75	55	93	55
Розмір цільової аудиторії *R*, %	25	25	12	22	20	40	14
Якість реклами *Q*, бали	7	6	7	4	5	9	4

Таким чином, кожній з 7 альтернатив виявився співставленим тримірний числовий вектор (рис. 2).

A ⟶		(60, 25, 7)
B ⟶		(65, 25, 6)
C ⟶		(60, 12, 7)
E ⟶		(75, 22, 4)
H ⟶		(55, 20, 8)
I ⟶		(93, 40, 9)

Рис. 2. Формалізований опис альтернатив вибору ЗМІ (розроблено автором)

Визначмо множину альтернатив ЗМІ, які є оптимальними з точки зору принципу Парето. Припустимо, що ми розглядаємо альтернативи *A* та *B*. Виявляється, що альтернатива *B* гірша за альтернативу *A* за параметрами ціни контракту та якості реклами, проте має однакове значення з розміром цільової аудиторії. Таким чином, альтернатива *B* є менш вигідною за *A* у двох показниках. Продовживши таке порівняння з іншими альтернативами, ми визначимо множину рішень, які відповідають принципу Парето і потребують подальшого аналізу [11].

Таблиця 2

Множина парето-оптимальних рішень (розроблено автором)

Критерії	Альтернативи		
	A	**H**	**I**
Ціна контракту *P*, тис. дол.	60	55	93
Розмір цільової аудиторії *R*, %	25	20	40
Якість реклами *Q*, бали	7	5	9

Нам потрібно провести переведення значень критеріїв з різних шкал в одну бальну шкалу. Наприклад, якщо у нас є три критерії зі значеннями в абсолютній шкалі та критерій у бальній шкалі, нам потрібно привести всі значення до спільної шкали, наприклад, від 1 до 3, де 1 – найгірше значення, 3 – найкраще. Таким чином, ми зможемо порівняти різні критерії за однаковими критеріями оцінки (табл. 3)

Таблиця 3

Оцінка критеріїв в єдиній бальній шкалі (розроблено автором)

Критерії	Альтернативи		
	А	**Н**	**І**
Ціна контракту *P*, тис. дол.	2	3	1
Розмір цільової аудиторії *R*, %	2	1	3
Якість реклами *Q*, бали	2	1	3

Отже, залишаються лише альтернативи, що відповідають принципу Парето. Кожна з цих альтернатив завжди перевершує інші за одним критерієм, але при цьому вона може бути гіршою за іншими критеріями.

Аналіз, який враховує ступінь відхилення значень критеріїв від їх найкращих значень, можна виконати через застосування пропорційної шкали. Наприклад, використовуючи десятибальну або стобальну шкалу, де 10 оцінює найкраще можливе значення критерію, а 1 – найгірше, можна здійснити більш точне відображення різниці між альтернативами за кожним критерієм.

Наприклад, за критерієм – ціна контракту найкраще значення 55 тис. дол. досягається в альтернативах *Н* та *К* – оцінка 10 балів, найгірше 93 тис. дол. – альтернатива І – оцінка 1 бал.

Ціна одного балу:

$$\text{Критерій } P: \ \Delta P = \frac{93-55}{9} = 4{,}2 \text{ тис. дол.}$$

$$\text{Критерій } R: \ \Delta R = \frac{40-12}{9} = 3{,}1\%$$

$$\text{Критерій } Q: \ \Delta Q = \frac{9-4}{9} = 0{,}55 \text{ бала}$$

Значення всіх критеріїв в єдиній десятибальній шкалі наведено в табл. 4.

Таблиця 4

Оцінка критеріїв в єдиній десятибальній шкалі (розроблено автором)

Критерії	Альтернативи		
	А	**Н**	**І**
Ціна контракту *P*, тис. дол.	9	10	1
Розмір цільової аудиторії *R*, %	5	3	10
Якість реклами *Q*, бали	6	2	10

Отже, в результаті залишаються альтернативи, які є оптимальними за принципом Парето. Кожна з цих альтернатив є кращою за одним критерієм, але водночас гіршою за іншим.

Висновки. Таким чином, для ефективного розв'язання будь-якої задачі, ключовим є чітке визначення проблеми, яка потребує вирішення. Саме у формулюванні цієї проблеми фахівці бачать можливості знайти вихід. У випадку індивідуального прийняття рішення керівник може також користуватися своїм власним досвідом у схожих обставинах. При групових методах зазвичай кожен експерт вносить свій внесок у розуміння контексту, в якому буде прийматися рішення. У даній статті була представлена задача прийняття рішення, яка стосувалась вибору ЗМІ для укладення рекламного контракту. Однак, визначення оптимальної альтернативи у цьому випадку ускладнюється тим, що критерії порівняння не є однозначно визначеними. Пропозиції ЗМІ мають різноманітні параметри, такі як ціна, якість реклами, розмір аудиторії тощо. Це означає, що система переваг альтернатив на цей час не є повною, і неможливо однозначно порівняти альтернативи між собою за відсутності чітких критеріїв для оцінки.

Список використаних джерел:

1. Алгоритм прийняття рішень: методи, розробка завдань та реалізація планів. [Електронний ресурс] URL: https://what.com.ua/algoritm-priiniattia-rishen-m/ (дата звернення 26.12.2023).

2. Бондар Н. П. Сучасний простір медіаграмотності та перспективи його розвитку (інформаційно-методичні матеріали). Херсон : Видавництво Навчально-методичного центру освіти у Херсонській області, 2020. 204 с.

3. Застосування методу лінійної згортки для прийняття рішення. [Електронний ресурс] URL: https://ela.kpi.ua/bitstream/123456789/5393/1/4.pdf (дата звернення 26.12.2023).

4. Копистинська І. М. Проблематика засобів масової комунікації та реклами. Навчальний посібник. Івано-Франківськ. 2022. 204 с.

5. Основи прийняття фінансових та інвестиційних рішень. [Електронний ресурс] URL: https://otherreferats.allbest.ru/finance/00089639_0.html (дата звернення 26.12.2023).

6. Прийняття рішень. [Електронний ресурс] URL: https://studopedia.com.ua/1_224983_priynyattya-finansovih-rishen.html (дата звернення 26.12.2023).

7. Прийняття рішення за допомогою побудови супер критерію методом лінійної згортки. [Електронний ресурс] URL: https://studfile.net/preview/6342097/page:12/ (дата звернення 26.12.2023).

8. Пушкар О. І., Грабовський Є. М. Культура цифрових медіа [Електронний ресурс] : навчальний посібник. Харків : ХНЕУ ім. С. Кузнеця, 2022. 164 с.

9. Makedon, V., Zaikina, H., Slusareva, L., Shumkova, O., Zhmaylova, O. Rebranding in the Enterprise Market Policy. Proceedings of the 34rd International Business Information Management Association Conference, IBIMA 2019: Vision 2025: Education Excellence and Management of Innovations. through Sustainable Economic Competitive Advantage: pp. 9472-9476

10. Nightingale. V. The handbook of media audiences. Wiley-Blackwell, 2011. 550 p.

11. Patarakin E., Burov V., Remorenko I. Scaffolding educational community of practice using visual storytelling. In: Proceedings of the 10th International Conference on Theory and Practice of Electronic Governance. New York : Romania, 2017. P. 355–358.

12. Shelukhin M., Kupriichuk V., KyrylkoN., Makedon V., Chupryna N. Entrepreneurship Education with the Use of a Cloud-Oriented Educational Environment. International Journal of Entrepreneurship. 2021. Volume 25. Issue 6. URL: https://www.abacademies.org/articles/entrepreneurship-education-with-the-use-of-a-cloudoriented-educational-environment-11980.html (дата звернення 27.12.2023).

References:

1. Alhorytm pryynyattya rishen': metody, rozrobka zavdan' ta realizatsiya planiv [Decision-making algorithm: methods, development of tasks and implementation of plans]. Retrieved from: https://what.com.ua/algoritm-priiniattia-rishen-m/ (accessed date: 26 December 2024). [in Ukrainian].

2. Bondar, N. P. (2020). Suchasnyy prostir mediahramotnosti ta perspektyvy yoho rozvytku (informatsiyno-metodychni materialy) [Modern space of media literacy and prospects for its development (informational and methodological materials)]. Kherson: Vydavnytstvo Navchal'no-metodychnoho tsentru osvity u Khersons'kiy oblasti. [in Ukrainian].

3. Zastosuvannya metodu liniynoyi z·hortky dlya pryynyattya rishennya [Application of the linear convolution method for decision-making]. Retrieved from: https://ela.kpi.ua/bitstream/123456789/5393/1/4.pdf (Accessed: 26 December 2024). [in Ukrainian].

4. Kopystyns'ka, I. M. (2022). Problematyka zasobiv masovoyi komunikatsiyi ta reklamy. Navchal'nyy posibnyk [Problems of mass communication and advertising. Tutorial]. Ivano-Frankivs'k. [in Ukrainian].

5. Osnovy pryynyattya finansovykh ta investytsiynykh rishen' [Basics of making financial and investment decisions]. URL: https://otherreferats.allbest.ru/finance/00089639_0.html (accessed date: 26 December 2024). [in Ukrainian].

6. Pryynyattya rishen' [Making decisions]. Retrieved from: https://studopedia.com.ua/1_224983_priynyattya-finansovih-rishen.html (accessed date: 26 December 2024). [in Ukrainian].

7. Pryynyattya rishennya za dopomohoyu pobudovy super kryteriyu metodom liniynoyi z·hortky [Making a decision using the construction of a super criterion by the method of linear convolution]. Retrieved from: https://studfile.net/preview/6342097/page:12/ (accessed date: 26 December 2024). [in Ukrainian].

8. Pushkar, O. I., Hrabovs'kyy, YE. M. (2022). Kul'tura tsyfrovykh media [Elektronnyy resurs] : navchal'nyy posibnyk [Culture of digital media [Electronic resource]: educational guide]. Kharkiv: KHNEU im. S. Kuznetsya. [in Ukrainian].

9. Makedon, V., Zaikina, H., Slusareva, L., Shumkova, O., Zhmaylova, O. (2019). Rebranding in the Enterprise Market Policy. Proceedings of the 34rd International Business Information Management Association Conference, IBIMA 2019: Vision 2025: Education Excellence and Management of Innovations. through Sustainable Economic Competitive Advantage, 9472-9476.

10. Nightingale, V. (2011). The handbook of media audiences. Wiley-Blackwell.

11. Patarakin, E., Burov, V., Remorenko, I. (2017). Scaffolding educational community of practice using visual storytelling. In: Proceedings of the 10th International Conference on Theory and Practice of Electronic Governance. New York : Romania, 355–358.

12. Shelukhin, M., Kupriichuk, V., Kyrylko, N., Makedon, V., Chupryna, N. (2021). Entrepreneurship Education with the Use of a Cloud-Oriented Educational Environment. International Journal of Entrepreneurship. Volume 25, Issue 6. Retrieved from: https://www.abacademies.org/articles/entrepreneurship-education-with-the-use-of-a-cloudoriented-educational-environment-11980.html (accessed date: 27 December 2024).

УДК 519.681.2
DOI https://doi.org/10.32689/maup.it.2023.5.7

Олексій ПІСКУНОВ
кандидат фізико-математичних наук, старший науковий співробітник, доцент кафедри прикладної математики, Національний авіаційний університет, просп. Любомира Гузара, 1, Київ, Україна, індекс 03058 (oleksii.piskunov@npp.nau.edu.ua)

ORCID: 0000-0002-9200-3422

Наталя ТУПКО
кандидат фізико-математичних наук, доцент, доцент кафедри прикладної математики, Національний авіаційний університет, просп. Любомира Гузара, 1, Київ, Україна, індекс 03058 (natalia.tupko@npp.nau.edu.ua)

ORCID: 0000-0003-0625-3271

Іван ПЕТРЕНКО
інженеренер-програміст, Splunk, вулиця Добрий пастир, 122б, кв. 37, Краків, Польща, індекс 31-416 (humty.ua@gmail.com)

ORCID: 0009-0000-3409-0022

Oleksii PISKUNOV
Candidate of Physical and Mathematical Sciences, Senior Research Fellow, Associate Professor at the Department of Applied Mathematics, National Aviation University, 1, Lubomyra Huzara Ave, Kyiv, Ukraine, postal code 03058 (oleksii.piskunov@npp.nau.edu.ua)

Natalia TUPKO
Candidate of Physical and Mathematical Sciences, Associate Professor, Associate Professor at the Department of Applied Mathematics, National Aviation University, 1, Lubomyra Huzara Ave, Kyiv, Ukraine, postal code 03058 (natalia.tupko@npp. nau.edu.ua)

Ivan PETRENKO
Software Engineer, Splunk, 122b, app. 37, Dobryi pastyr St, Krakow, Poland, postal code 31-416 (humty.ua@gmail.com)

Бібліографічний опис статті: Піскунов, О., Тупко, Н., Петренко, І. (2023). Алгебраїчне проєктування програмного забезпечення. *Інформаційні технології та суспільство*, 5 (11), 50–59. DOI: https://doi.org/10.32689/maup.it.2023.5.7

Bibliographic description of the article: Piskunov, O., Tupko, N., Petrenko, I. (2023). Alhebraichne proiektuvannia prohramnoho zabezpechennia [Algebraic software design]. *Informatsiini tekhnolohii ta suspilstvo – Information technology and society,* 5 (11), 50–59. DOI: https://doi.org/10.32689/maup.it.2023.5.7

АЛГЕБРАЇЧНЕ ПРОЄКТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Анотація*. У статті розглянуто алгебраїчний підхід до проєктування та тестування програмного забезпечення.* ***Метою*** *статті є розробка класу, який реалізує арифметику Пеана. Арифметика Пеана є однією з базових конструкцій при аксіоматичному побудуванні математики. Клас арифметики, який представлений в статті є півгрупою натуральних чисел і є першим в ієрархії числових класів (група цілих чисел, кільце цілих чисел, поле раціональних чисел) для демонстрування небезпечності універсального поліморфізму підтипів та порушення принципу підстановки Лісков.* ***Методи дослідження:*** *під час дослідження використовуються базові положення методу проєктування по контракту Бертрана Меєра та методу формальної розробки RAISE, які дозволяють застосовувати формальну логіку.* ***Наукова новина дослідження*** *полягає в тому, що змінене трактування до функціонального типу методів класу та більш широке використання вимог до класу у вигляді алгебраїчних рівностей, якими описуються вимоги до інваріантів, передумов та післяумов. Об'єднання вимог до функціональних типів і цих рівностей логічною зв'язкою 'і' дозволяє однозначно прогнозувати порушення принципу підстановки та вимагає змінення правила підстановки (subsumption). Крім цього, звертається увага на властивість категоричності відповідної алгебраїчної моделі, яка робить неможливими некоректні реалізації, на прямі аналогії між аксіоматичним викладенням математичної теорії та розробкою технічного завдання програмістам. Також, даний підхід значно спрощує підбір тестів для димового тестування (smoke testing) програмного забезпечення.* ***Висновки.*** *Алгебраїчне проєктування та тестування базується на математичних принципах, що дозволяє: уникати двозначності і неоднозначності в описі функціональності; забезпечувати точність та однозначність у формулюванні вимог до програми; автоматизувати процес генерації тестових випадків та перевірки роботи, відповідно підвищувати надійність; виявляти*

і усувати помилки ще на стадії розробки та прискорювати розробку програмного забезпечення. Запропоновані в статті доповнення до алгебраїчного підходу проєктування по контракту та методу формальної розробки RAISE демонструють універсальність алгебраїчного проєктування, яке дозволяє перейти від мистецтва програмування та мистецтва тестування програм до формальних технологічних прийомів.

Ключові слова: *алгебраїчне проєктування, алгебраїчне тестування, метод RAISE, димове тестування, проєктування за контрактом, принцип підстановки, небезпечність універсального поліморфізму підтипів.*

ALGEBRAIC SOFTWARE DESIGN

Abstract. *The article discusses an algebraic approach to designing and testing software. The purpose of the article is to develop a class that implements Peano arithmetic. Peano arithmetic is one of the fundamental constructs in axiomatic mathematics. The arithmetic class presented in the article represents a semigroup of natural numbers and this class is the first example in the hierarchy of numerical classes (integers, integer rings, rational number fields) that demonstrates the potential unsafety of universal inclusion polymorphism and violations of the Liskov substitution principle.* ***Research methods.*** *During the study, basic principles of the Bertrand Meyer's design by contract method and the formal development method RAISE are used, which allow applying formal logic.* ***Scientific novelty.*** *The modified interpretation of the functional type of class methods and the consistent use of requirements in the form of algebraic equalities make it possible at the time of design to indicate the unsafety of universal inclusion polymorphism. Additionally, attention is drawn to the categoricality (rigitity) of algebraic model, which makes incorrect implementations impossible, and to direct analogies between the axiomatic presentation of mathematical theory and the development of specifications. Furthermore, this approach significantly simplifies the of smoke test design.* ***Conclusions.*** *Algebraic design and testing are based on mathematical principles, allowing for the avoidance of ambiguity and uncertainty in functionality descriptions, ensuring accuracy and unambiguity in formulating specifications, automating the process of test cases design and verification of software requirements, thereby makes it easier to detect and correct a design and coding errors.*

Key words: *unsafety of universal inclusion polymorphism, functional type of class method, design by contract, RAISE method, algebraic equalities.*

Постановка проблеми та аналіз останніх досліджень і публікацій. Поліморфізм підтипів (шаблонів) згідно з правилом підстановки, який використовується в класичних мовах програмування, наприклад C++ та C#, дозволяє передавати змінні похідного класу, тобто наступного, у функції, які написані в термінах базового класу [7]. В загальному розумінні, ця властивість постулюється як основна перевага мов об'єктно-орієнтованого програмування (ООП) над мовами, які не належать до ООП. Стверджується, що при цьому можна розробляти алгоритми, які однаково успішно працюють як зі змінними базового класу, так і зі змінними похідного. Відповідно у розроблені за такими алгоритмами поліморфній функції можна безпечно передавати змінні довільних похідних класів. При цьому досить давно в роботах Б. Лісков [21] було зауважено, що таке використання поліморфної функції є безпечним, якщо тип похідної функції збігається з типом базового класу. Один із широко відомих прикладів, який згадується в роботах Б. Мартіна з квадратами та прямокутниками [14], підтверджує спостереження Б. Лісков. Наведені факти заперечують думку розробників компіляторів класичних ООП мов, що успадкування являє собою формування підтипу та безпосередньо вказує на наявність проблеми. При цьому не пояснюється наперед як потрібно проєктувати базовий та похідний клас, щоб і успадкування, і використання поліморфних функцій було безпечним. Відповідно не зрозуміло, в яких випадках порушується принцип підстановки Б. Лісков, а в яких – ні. До того ж, автори прикладів (Р. Мартін) не замислюються про обговорення такої властивості як категоричність. Наприклад, у монографіях К. Дейт та Б. Меєр [10; 15] успадковують квадрати від прямокутників. З вище сказаного, напрошуються висновок: інструментальні засоби розробки без попереджень дозволяють використовувати небезпечний поліморфізм підтипів, який приводить до неочікуваного порушення специфікацій і виникає питання: яким чином правильно проєктувати потрібні класи?

Виклад основного матеріалу. Перехід до формального обговорення проблеми вимагає формальних визначень для понять, що використовуються. Згідно з джерелами [1; 2], термін «тип» буде вважатись синонімом до терміну «абстрактний тип даних». Термін «клас», за Б. Меєром [15], трактуватиметься як «запрограмований (реалізований) абстрактний тип даних». В якості властивості, яка має виконуватись для змінної базового класу [21], щоб формально дотримуватись принципу підставки, буде обрано наступну властивість: алгоритм проєктованої поліморфної функції задовольняє вимогам, що пред'являються до нього, тобто задовольняє своїй специфікації.

З цієї формальної точки зору, на основі загальних положень проєктування за контрактом Б. Меєра та формального методу розробки RAISE [12] приклад Р. Мартіна було розглянуто в роботі [2]. У ході зворотної розробки для класів прямокутників та квадратів були представлені відповідні їм абстрактні типи даних, а саме, як пояснюється в роботі [15], введена невизначена множина даних Figure та множина невід'ємних дійсних чисел UReal, для них задані функції «взяти довжину однієї сторони» (getWifth: Figure → UReal), «задати довжину другої сторони» (setHeight: Figure × UReal → Figure). При цьому, властивості цих функцій (і всіх інших необхідних у прикладі) задавалися у вигляді алгебраїчних рівностей:

$$\forall r : \mathrm{UReal}, f : \mathrm{Figure} \cdot \mathrm{getWidth}\big(\mathrm{setHeight}(f, r)\big) = \mathrm{getWidht}(f)$$

Тобто, для довільної фігури виконується вимога: яку б довжину однієї сторони не задавали, довжина другої сторони не змінюється. Отже, вимоги до значень параметрів функцій, до властивостей значень, які повертаються, та самих функцій визначаються за допомогою алгебраїчних рівнянь. Список усіх множин, сигнатур функцій та алгебраїчних рівностей утворював у термінах Б. Меєра [15] «контракт». Цей контракт повинен виконуватись при спадкуванні. Але в розглянутому прикладі це виявилося не так, контракт виявився порушеним. В об'єктах одного класу при зміні довжини однієї сторони змінювалася довжина іншої. В об'єктах другого класу довжина другої сторони не змінювалась. Зрозуміло, що це призводило до порушень роботи поліморфних функцій. Тобто методи, які змінюють вміст свого об'єкту, виявлялися причиною порушення роботи поліморфної функції.

Надалі, при розгляді цього прикладу довелося дещо змінити трактування поняття типу методів та наблизити його до типу функцій відповідного абстрактного класу. До списку класів формальних параметрів і класу значення, що повертається (які беруться з сигнатури методу) був доданий клас до якого належить метод. Наприклад, замість сигнатури методу f класу c (метод не змінює об'єкт this):

```
class c{
  int fv(T v);
}
```

Розглядається його функціональний тип з явним додаванням ще однієї множини допустимих значень для прихованого параметра this:

$$f : dom(c) \times dom(T) \to dom(int).$$

Зрозуміло, що у функціональному типі (методів, які приводять до порушення принципу підстановки) домен класу з'являється зліва і справа від функціональної стрілки.

Зауваження.

dom(c) – це множина допустимих кортежів, які можуть з'являтися в об'єктах класу c. Згідно з Л. Карделлі [8] (Objects as records), математична парадигма «об'єкти-як-записи» (тобто, як кортежі з прямих добутків деяких множин) цілком достатня для опису всіх основних властивостей об'єктів. Зважаючи на те, що функція є певним підмножиною прямого добутку області визначення та множини значень, будь-яка функція також може розглядатися як компонента такого кортежу. Отже, кортеж може містити числа, символи, рядки і, крім того, функції. Це впритул підводить поняття об'єкта до поняття абстрактного автомата. Що в свою чергу, означає: твердження Д. Парнаса [17]) «Змінна – це автомат» слід розуміти буквально. Спадкування абстрактних автоматів розглядалося в [18].

Далі буде потрібне визначення таких понять: метод, функція стану або переходу, конструктор, функція виходу.

Метод – це функція, яка визначена в області видимості класу. Функціональний тип будь-якого методу, крім статичного, повинен мати додатковий параметр порівняно з його сигнатурою ліворуч та/або праворуч від функціональної стрілки.

Функція переходу (у розумінні переходу з одного стану об'єкта в інший) – це функція, в сигнатурі якої домен класу з'являється ліворуч і праворуч від функціональної стрілки (→ для всюди визначеної функції або ↦ для частково визначеної функції). Згідно з [12] така функція називається генератором (generator).

Конструктор – це функція, в сигнатурі якої домен класу з'являється тільки праворуч від функціональної стрілки і відсутній ліворуч.

Функція виходу (функція – спостерігач (observer) – це функція, в сигнатурі якої домен класу з'являється лише ліворуч від функціональної стрілки і відсутній справа [12].

Аксіоматика Пеано та напівгрупа натуральних чисел

У цьому розділі до множини чисел, які визначаються аксіомами Пеано, додані аксіоми для функції односпрямованого ітератора: взяти наступне (successor) suc і функції додавання (summator) sum. Отже:

1. P_0 : існує число 1, яке не слідує ні за яким натуральним числом.
2. P_1 : кожному натуральному числу n однозначне відповідає безпосередньо наступне за ним натуральне число n'.
3. P_2 : будь-яке натуральне число n, за винятком 1, безпосередньо слідує за одним і тільки одним натуральним числом.
4. P_3 : Якщо твердження S доведено для 1 і якщо з припущення, що воно вірне для натурального числа n, випливає, що воно вірне для безпосередньо наступного натурального числа n', то твердження S виконується для всіх натуральних чисел.

Аксіома математичної індукції P_3 , вже реалізована в багатьох мовах програмування у вигляді ітерації (операторів циклу) та/або рекурсії, тому надалі не буде згадуватися. Дещо в іншому вигляді, з цією ж специфікацію на мові RSL можна ознайомитись в [22], розділ 'Example: The Natural Numbers'.

Множина натуральних чисел задовольняє перерахованим аксіомам і позначимо її через $\mathfrak{N}$. Специфікація для ітератора має вигляд:

$$P_4 : suc : \mathfrak{N} \to \mathfrak{N} \wedge \forall n \in \mathfrak{N} \cdot suc(n) = n'$$

Специфікація для суматора Грассмана [5] має вигляд:

$$P_5 : sum : \mathfrak{N} \times \mathfrak{N} \to \mathfrak{N} \wedge \forall a, r \in \mathfrak{N}:$$

$$sum(a,1) = suc(a) \wedge sum(a, suc(r)) = suc(sum(a,r))$$

Крім того, з двох останніх аксіом видно, що вони постулюють операцію порівняння двох чисел, тобто введення символу '='.

Дуже важливою характеристикою такої специфікації для функцій suc і sum виявляється наступне:

– неявність специфікації функцій, що є наслідком використання алгебраїчних рівностей [15] (ця неявність є ключовим аспектом лаконічного опису абстрактних типів даних та їх майбутніх аналогів – класів);

– аксіоми для відносин між функціями дають лаконічний опис властивостей специфікованого типу в суто математичних термінах без допомоги імперативних міркувань.

Якщо у роботах Б. Меєра процес вибору аксіоми для відносин між функціями згадуються не надто явно, то у авторів методу формальної розробки RAISE аксіоматичний опис попарного застосування функцій з'являється вже як обов'язкова вимога методу розробки:

– For each possible combination of non-derived observer and non-derived generator, define an axiom expressing the relation between them. We have three non-derived generators and two non-derived observers, so we have six such axioms. These axioms are called observer-generator axioms [12].

– Для кожної можливої пари складеної з функції – спостерігача (observer), яка не виводиться (тобто не може бути запрограмована за допомогою інших функцій класу) і функції- генератора (generator), яка не виводиться, визначити аксіому, що виражає відношення між цими функціями.

Самі розробники методу формальної розробки RAISE з питання алгебраїчного підходу до аксіоматичного опису відносин між функціями відсилають до робіт Johna Guttag [11], який, у свою чергу, відсилає до Хоара та Флойда: «The algebraic approach used here owes much to the work of Hoare (which in turn owes much to Floyd)».

Насправді, аксіоматичний опис відношення двох функцій, що специфікуються, ще раніше з'явився в алгебрі. Наприклад, саме так виглядає аксіома дистрибутивності $\mathbb{Z}_{VIII}$ в аксіоматиці кільця цілих чисел [20]:

$$\forall a,b,c \in \mathbb{Z} \cdot (a+b) \cdot c = a \cdot c + b \cdot c .$$

У функціональному записі ця аксіома виглядає наступним чином:

$$sum : \mathbb{Z} \times \mathbb{Z} \to \mathbb{Z}$$

$$mult : \mathbb{Z} \times \mathbb{Z} \to \mathbb{Z}$$

$$\forall a,b,c \in \mathbb{Z} \cdot mult(sum(a,b),c = sum(mult(a,c),mult(b,c))$$

Тобто, нічим не відрізняється від специфікації абстрактного типу даних. Треба визнати, що програмісти повторюють шлях математиків дев'ятнадцятого та початку двадцятого століть.

Далі, між алгебраїстами і Б. Меєром з одного боку і розробниками методу RAISE з іншого, можна виявити наступну відмінність: перша сторона, на відмінну від другої, специфікує відносини не тільки між парами функцій стану і функцій виходу, а також і між двома функціями стану. В роботі [15] Б. Меєр дає аксіому для пари функцій стану (non-derived generator) remove і put. У представлених вище аксіомах для арифметики Пеано також є аксіома для спільного опису додавання sum і ітератора suc. Зрозуміло, що обидві ці функції теж є функціями стану.

Крім того, при алгебраїчному підході ніхто не змушує задавати властивості функцій обов'язково попарно. Тобто функцій може бути одна (складання будь-якого числа з нулем дає нуль) або, скажімо, чотири. Так наприклад, визначаються функції синуса та косинуса:

$$sum(mult(sin(x),sin(x)),mult(cos(x),cos(x))) = 1$$

Тут символом mult позначається функція добутку дійсних чисел. Звичайно, такі складні рівності ускладнюють тестування. Налагодження та тестування потрібно починати з більш простих рівностей.

Відповідь на питання, чи є достатньо повною така аксіоматична специфікація типу, яка використовується в методах формальної розробки ПЗ, залишимо на авторитеті Б. Меєра та J. Guttag-а: «A detailed look at sufficient-completeness is contained in Guttag».

Специфікація арифметики Пеано мовою формальних специфікацій Z

Проєкт із програмуванням арифметики Пеано є невеликим. Мова Z [19] використовується у ньому з наміром освоєння мов формальних специфікацій у закінчених проєктах.

Це полегшує необхідність робити під час розробки, на думку Д. Парнаса, «невеликі кроки, щоб забезпечити відповідність між абстрактним уявленням користувачів про систему і конкретним працюючим кодом» [17]. ASCII-версія специфікації арифметики Пеано очевидно може бути записана мовою формальних специфікацій Z [3;19]. Початкову ASCII – версію специфікації, придатну для безпосереднього використання в коментарях коду програм, утилітою ZTC [13], можна відконвертувати в LaTex-версію, у форму звичну для математиків.

Приклад ASCII-версії специфікації з аксіомами визначення функцій suc і sum:

```
spec
   generic[T]
      one: T;
      suc: T fun T;
      sum: T & T fun
   where
      forall a,b: T @ sum(a,one) = suc(a)
          and sum(a,suc(b)) = suc(sum(a,b))
      end generic
      end spec
```

Відкомпільована в LaTex версія специфікації виглядає так:

one : T
suc : T → T
sum : T × T → T
∀ a, b : T · (sum(a, one) =
suc(a) ∧ sum(a, suc(b)) =
suc(sum(a, b))

Отримана специфікація повністю збігається з рекомендаціями Б. Меєра [15].

Обґрунтування існування алгоритму для суматора Грассмана

Перед тим як розпочати розробку поставленої задачі, важливо переконатися, що необхідний алгоритм існує і може бути реалізований (чого зазвичай програмісти не роблять). Запропонований в статті проєкт начебто створений для застосування частково-рекурсивних функцій Черча на практиці. Ітератор Пеано збігається з найпростішою функцією Черча з тою ж назвою suc – це перше правило з визначення суматора Грассмана. Друге правило записується через композицію ітератора та проєктування кортежу з трьох елементів на третю координату. Значить правило може бути виражене через елементарні операції над функціями і задовольняє вимогам рекурсії:

$$\mathrm{sum}(x,\mathrm{suc}(k)) = \mathrm{suc}(\mathrm{sum}(x,k)) = \mathrm{suc}\left(pr_3^3(x,k,\mathrm{sum}(x,k))\right)$$

Використовуючи примітивну функцію проєктування кортежу з 3 чисел на третій співмножник – pr_3^3, формулу обчислення суматора вдалося подати у вигляді оператору рекурсії Черча. Відповідно суматор можна запрограмувати.

Алгоритм ітератора

При описі алгоритму використовуються наступні позначення:

- dgts – це множина цифр від 0 до 9.
- seq1 – позначення типу непустих послідовностей.
- front – уся послідовність без останнього елемента.

– last – останній елемент послідовності.
– функція suc – на непустій послідовністю цифр (seq1 dgts) будує іншу.
– Послідовності записуються за допомогою подвоєних кутових дужок – << … >> в ASCII-стилі.
– Функція suc1 обчислює наступне значення останнього елемента послідовності і повертає пару: послідовність довжиною 1 із, можливо, зміненої цифри << n + a >> та ознаки переповнення.
– Функція suc2 – це крок рекурсії.

Функція suc змінює останній елемент послідовності і застосовує себе до меншої непустої послідовності без останнього елемента, з можливо зміненою ознакою переповнення. Символ ^ означає операцію конкатенацію послідовностей. Якщо послідовність була точно з одного елемента, то у разі переповнення до зміненого останнього елемента спереду додається 1.

```
---- File:./z/suc.zsl
spec
  input suc0.zcl
  schema suc
    suc : seq1  dgts          fun   seg1 dgts;
    suc2 : seq1  dgts  &0..1 fun   seg1 dgts;
    suc1 : gts           &0..1 fun  dgts &0..1;
  where
    forall s : seq1  dgts@  suc(s) = suc2(s,0);
    forall s : seq1  dgts;a : 0..1@
    suc2(s,a) = (let na == suc1(last s,a);n1 == ≪ first na ≫;a1 == second na  @ if  front s ≠ ≪≫
              then suc2(front s,a1)   ^n1
              else if  a1 = 0
              then n1
              else ≪ 1 ≫ ^ n1
              );
    forall n : dgts;          a : 0..1@
              suc1(n,a) = if a = 1 then
              if n + a > 9  then (0,1)  else (n + a,0)
              else (n,0)
  end schema
  end spec
---- End Of File:./z/suc.zsl
```

Специфікація алгоритму ітератора на LaTex:

```
dgts : P N
dgts = 0 . . 9

suc
suc : seq1 dgts → seq1 dgts
suc2 : seq1 dgts × 0 . . 1 → seq1 dgts
suc1 : dgts × 0 . . 1 → dgts × 0 . . 1
∀ s : seq1 dgts · suc(s) = suc2(s, 0)
∀ s : seq1 dgts; a : 0 . . 1 · suc2(s, a) =
   (let na == suc1(last s, a); n1 == first na ; a1 == second na · if
      front s /= then suc2(front s, a1) ^ n1
        else (if a1 = 0 then n1 else 1^ n1))
∀ n : dgts; a : 0 . . 1 · suc1(n, a) =
    if a = 1 then (if n + a > 9 then (0, 1) else (n + a, 0))
     else (n, 0)
```

Ця спроба описати алгоритм мовою Z виглядає незручною і марною, бо в результаті програмування текст виявився чи не довшим за текст методу [5]. Відсутність аналога змінних і виразів подібних до операторів циклів (які присутні в мові формальних специфікацій RAISE (далі – RSL) [22] ускладнює практичну роботи програмістів і робить її менш зручною порівняно з RSL.

Трохи зручніша версія специфікації ітератора може виглядати наступним чином:

```
dgts : P N
─────────────
dgts = 0 . . 9

inc == ( λ old, new : dgts · if (old = 9 ∧ new = 0) then 1 else 0)

mdf == (λ s : seq1 dgts; i : 0 . . 1 · if i = 1 then 0 ^a s else s)

suc ──────────────────────────────────
suc : seq1 dgts → seq1 dgts
──────────────────────────────────
∀ s : seq1 dgts ·
∀ i : 1 . . #(suc(s)) − 1 ·
   let ms == mdf(s, #(suc(s)) − #s) ·
   (last(suc(s)) = (last s + 1) div 10
   ∧ suc(s)(i) = ms(i) + inc(ms(i + 1), suc(s)(i + 1)))
──────────────────────────────────
```

Як і у випадку схеми для факторіалу цикл замінюється синтаксичною конструкцією з квантором загальності. Короткі функції визначаються через лямбда – вирази. Функція inc визначає наявність переповнення, воно виникає якщо вихідна цифра дорівнювала цифрі 9, а нова – дорівнювала цифрі 0. Функція mdf змінює вихідну послідовність цифр, дописуючи їй лідируючий нуль. Це потрібно зробити, якщо нова послідовність виявиться довшою за вихідну. Така ситуація виникає, коли вихідна послідовність починалася з цифри 9. ms – це позначення можливо подовженої вихідної послідовності, яке вводиться конструкцією let. Тобто, ms це застосування функції mdf до вихідної та різниці довжин вихідної та нової suc(s) послідовностей. В результаті послідовності ms і suc(s) гарантовано мають однакову довжину. Причому остання цифра у suc(s) рахується додаванням 1 до останньої цифри послідовності s та діленням на 10. А кожна наступна цифра нової послідовності suc(s)(i) визначається як сума поточної цифри старої та результату переповнення на попередньому (i + 1) кроці.

Аксіоматика комутативної напівгрупи мовою Z

Аксіоматика комутативної напівгрупи виглядає наступним чином [1]:

```
[T] ═══════════════════════════
  f : T × T → T
──────────────────
∀ x, y : T · (∃1 z : T · f(x, y) = z)
∀ x, y, z : T · f(x, f(y, z)) = f(f(x, y), z)
∀ x, y : T · f(x, y) = f(y, x)
──────────────────
```

Множина T з функцією f називається комутативною напівгрупою, якщо виконуються умови:

1. f – всюди визначена (стрілка, яка використовується в сигнатурі функції є символом всюди визначеної функції);
2. функція f задовольняє умову асоціативності;
3. функція f задовольняє умову комутативності.

Множина чисел Пеано та функція суматора Грасмана, згідно доведеного вище, задовольняють першу умову визначення напівгрупи. Докладно про доведення асоціативності та комутативності суматора Грассмана описано в роботі [5].

Таким чином, реалізований у проєкті [5] клас natural можна вважати напівгрупою натуральних чисел за додаванням.

Функція віднімання

Після обговорення суматора потрібно було розглянути формули для порівняння двох чисел (менше, рівно, більше), сформувати двосторонній натуральний ряд, та ввести операцію віднімання. Для цілей дослідження успадкування в поточному проєкті була реалізована функція віднімання, способом схожим на порівняння Грассмана з використанням ітератора. В силу того, що функція не є повністю визначеною на натуральних числах, вона не є операцією. Для деяких двох чисел a і b третє число, яке задовольняє умові sum(c, b) = a), буде називатися різницею і позначатися знаком dif(a, b). І це означає введення частково певної функції, яка буде називатися віднімання:

$$\mathrm{dif} : T \times T \mapsto \mathrm{sum}(\mathrm{dif}(a, b), b) = a$$

Різниця визначена не для всіх елементів з натуральних чисел, тому функція виявляється частково визначеною і позначається іншою функціональною стрілкою. Додаткові вимоги для специфікації напівгрупи натуральних чисел виглядають наступним чином:

$$[T]$$

$$\mathrm{sum} : T \times T \to T$$

$$\mathrm{dif} : T \times T \mapsto T$$

$$\forall x, y : T \cdot (\exists_1 z : T \cdot \mathrm{sum}(x, y) = z)$$

$$\forall x, y, z : T \cdot \mathrm{sum}(x, \mathrm{sum}(y, z)) = \mathrm{sum}(\mathrm{sum}(x, y), z)$$

$$\forall x, y : T \cdot \mathrm{sum}(x, y) = \mathrm{sum}(y, x)$$

$$\forall x, y : T \cdot (x, y) \in \mathrm{dom\ dif} \Rightarrow \mathrm{sum}(\mathrm{dif}(x, y), y) = x$$

Далі, саме функція віднімання в обох можливих спробах наслідування (як у бік звуження домену класу – від групи до напівгрупи, так і у бік розширення домену класу – від напівгрупи до групи [6]) призводила до порушення в роботі поліморфних функцій. Спільним між цим прикладом і прикладом Р. Мартіна є те, що віднімання є функцією стану, а домен її класу знаходиться з лівої і правої сторін функціональної стрілки.

Тестування

Подробиці з розробки та тестування класу natural викладені в роботі [5]. Зокрема, алгебраїчні рівності sum(a, one) = suc(a) та sum(a, suc(b)) = suc(sum(a, b)) були використані для розробки відповідних драйверів тестів [5]. У цих тестах просто підставляються різні значення для пошуку першої відмови. Програмування цих тестів виглядає значно простіше, ніж розробка тестів за технологіями чорного та/або білого ящиків [16]. При цьому важливо зазначити, що створення алгебраїчних тестів не є мистецтвом, а є досить механічним прийомом.

Можна відзначити, що тести з використанням алгебраїчних рівностей широко використовуються в монографії W. Cody і W. Waite-а [9] для роботи з функціями дійсного аргументу.

Висновки

– Алгебраїчний підхід до розробки та тестування ПЗ виглядає досить універсальним та перспективним засобом.

– Властивості функцій типу даних (як і методів класу), які задаються алгебраїчними рівностями не враховуються компіляторам при застосуванні універсального поліморфізму включення.

– Наявність функції стану (і відповідного методу в класі) у типі даних робить небажаною зміну домену класу. Поява домену класу одночасно з лівої та з правої сторони функціональної стрілки в силу одночасної ко- та контра-варіантності функціонального типу дозволяє успадкування, але робить небезпечним використання універсального поліморфізму включення

– Алгебраїчний підхід дозволяє перейти від мистецтва (в сенсі ‘Мистецтво програмування для ЕОМ’ Д. Кнута і ‘Мистецтво тестування програм’ Г. Майєрса) до досить простих, практично технічних прийомів програмування.

– Наявність у мові RSL синтаксичних конструкцій (Repetitive Expressions) для багаторазового повторення операторів дозволяє робити на мові RSL ‘невеликі кроки, щоб забезпечити відповідність між абстрактним уявленням про систему і конкретним кодом’. Таким чином, специфікації на мові RSL є більш зрозумілими та близькими до практичного кодування, ніж на мові Z.

Список використаних джерел:

1. Піскунов О.Г. Типи, множини та класи. 2011. С. 19. URL: https://www.researchgate.net/publication/334174126 (дата звернення: 01.02.2024).

2. Піскунов О.Г. Про відмінності між поняттями типу та. *Вісник Київського національного університету імені Тараса Шевченка*. Серія : Фізико-математичні науки. 2015. № 3. С. 106–114.

3. Піскунов О.Г. LaTex та вимоги державного стандарту. 2022. С. 74. URL: https://www.researchgate.net/publication/359860334 (дата звернення: 01.02.2024).

4. Піскунов О.Г., Жултинська А.К. Документування процесу розробки програмного забезпечення. 2024. С. 324. URL: https://www.researchgate.net/publication/377261513 (дата звернення: 01.02.2024).

5. Піскунов О.Г., Рудик В.І., Петренко І.А. Арифметика Пеано: від специфікації до класу. 2022. С. 45. URL: https://www.researchgate.net/publication/365979331 (дата звернення: 01.02.2024).

6. Піскунов О.Г., Мічуда А.М. Переозначення додавання: небезпечне наслідування в групі цілих. 2023. С. 36. URL: https://www.researchgate.net/publication/366867037 (дата звернення: 01.02.2024).

7. Cardelli L., Abadi M. A theory of objects. New York: Springer-Verlag, 1996. P. 396.

8. Cardelli L. A semantics of multiple inheritance. Information and Computation. 1988. № 76. P. 138–164.

9. Cody W., Waite W. Software manual for the elementary functions. New Jersey: Prentice-Hall, 1980. P. 289.

10. Date, C.J. An Introduction to Database Systems, 7th Edition. London, UK: Addison-Wesley, 2000. 938 p.

11. Guttag J.V. Abstract Data Types and the Development of Data Structures. Communications of the ACM. 1977. Vol. 20. № 6. P. 396-404.

12. Haxthausen A. Lecture Notes on The RAISE Development Method. Kongens Lyngby: DTU, 1999. P. 20.

13. Jia X. ZTC: A Type Checker for Z Notation. User's Guide (Version 2.03).Chicago: DePaul University, USA, 1998. P. 44.

14. Martin R. Clean Architecture: A craftsman's guide to software structure and design. Boston, U.S: Prentice-Hall, 2018, 378 p.

15. Meyer B. Object-Oriented Software Construction. Second Edition. London: Pearson Education, 2022. P. 1024.

16. Myers G., Sandler C., Badgett T. The art of software testing, 3rd ed. New Jersey, USA:J. Wiley & Sons, Inc, 2012. 240 p.

17. Parnas D.L. Really rethinking 'formal methods'. New York, U.S: IEEE Computer Society, Computer, 2010, N 43, pp. 28–34.

18. Piskunov A.G. Inheritance of Abstract Automata. *Вісник Київського національного університету імені Тараса Шевченка*. Серія : Кібернетика. 2011. № 11. С. 40-44.

19. Spivey J.M. The Z Notation: A Reference Manual, 2nd edition. New Jersey: Prentice Hall International Series in Computer Science, 1992. P. 158.

20. Stepanov A.A., Ros D.E. From Mathematics to Generic Programming. London, UK: Addison-Wesley, 2015, 285 p.

21. Wing J., Liskov B. Family Values: A Behavioral Notion of Subtyping. Pittsburgh,U.S: ACM, ACM Trans. Program. Lang. Syst., 16(6), 1994. pp 1812-1841 (дата звернення: 01.02.2024).

22. The RAISE Language Group. The RAISE SPECIFICATION LANGUAGE. Kongens Lyngby, Denmark: Prentice Hall Europe, 1992, 396 p.

23. Guttag, J.V. and Horning, J.J., The algebraic specifications of abstract data types. URL: https://www.semanticscholar.org/paper/The-algebraic-specification-of-abstract-data-types-Guttag-Horning/e4c8b1db0c839a07a833db51c5ac00e6ffd5a922 (дата звернення: 01.02.2024).

References:

1. Piskunov O.G. (2011). Typy, mnozhyny ta klasy [Types, sets and classes]. *ResearchGate, P. 19.* Retrieved from https://www.researchgate.net/publication/334174126.

2. Piskunov O.G. (2015). Pro vidminnosti mizh poniattiamy typu ta klasu [On the differences between the concepts of type and class]. *Visnyk Kyivskoho natsionalnoho universytetu imeni Tarasa Shevchenka. Seriia : Fizyko-matematychni nauky – Bulletin of Taras Shevchenko Kyiv National University. Series: Physical and mathematical sciences,* 3, *106–114* [in Ukrainian].

3. Piskunov O.G. (2022). LaTex ta vymohy derzhavnoho standartu. LaTex and the requirements of the state standard. *ResearchGate, P.* 74. Retrieved from https: //www.researchgate.net/publication/359860334.

4. Piskunov O.G., Zhultynska A.K. (2024). Dokumentuvannia protsesu rozrobky prohramnoho zabezpechennia [Documentation of the software development process]. *ResearchGate, P.* 324. Retrieved from https://www.researchgate.net/publication/377261513.

5. Piskunov O.G., Rudyk V.I., Petrenko I.A. (2022). Aryfmetyka Peano: vid spetsyfikatsii do klasu [Peano Arithmetic: From Specification to Class]. *ResearchGate, P.* 45. Retrieved from https://www.researchgate.net/publication/365979331.

6. Piskunov O.G., Michuda A.M. (2023). Pereoznachennia dodavannia: nebezpechne nasliduvannia v hrupi tsilykh [Redefining Addition: Dangerous Imitation in a Group of Integers]. *ResearchGate, P.* 36. Retrieved from: https://www.researchgate.net/publication/366867037.

7. Cardelli L., Abadi M. A. (1996). Theory of objects. New York, U.S.: Springer-Verlag.

8. Cardelli L. A. (1988). Semantics of multiple inheritance. *Information and Computation*,76.

9. Cody W., Waite W. (1980)/ Software manual for the elementary functions. New Jersey, U.S.: Prentice-Hall.

10. Date, C.J. (2000). An Introduction to Database Systems, 7th Edition. London, UK: Addison-Wesley.

11. Guttag J.V. (1977). Abstract Data Types and the Development of Data Structures. *Communications of the ACM,Vols. 20, 6, 396-404.*

12. Haxthausen A. (1999). Lecture Notes on The RAISE Development Method. Kongens Lyngby: DTU.

13. Jia X. (1998). ZTC: A Type Checker for Z Notation. User's Guide (Version 2.03).Chicago, U.S.: DePaul University.

14. Martin R. (2018). Clean Architecture: A craftsman's guide to software structure and design. Boston, U.S: Prentice-Hall.

15. Meyer B. (2022). Object-Oriented Software Construction. Second Edition. London, UK: Pearson Education.

16. Myers G., Sandler C., Badgett T. (2012). The art of software testing, 3rd ed. New Jersey, USA:J. Wiley & Sons, Inc.

17. Parnas D.L. (2010). Really rethinking 'formal methods'. *IEEE Computer Society, Computer, 43, 28–34.*

18. Piskunov A.G. (2011). Inheritance of Abstract Automata. *Visnyk Kyivskoho natsionalnoho universytetu imeni Tarasa Shevchenka. Seriia : Kibernetyka – Bulletin of Taras Shevchenko Kyiv National University. Series: Cybernetics, 11, 40-44.*

19. Spivey J.M. (1992). The Z Notation: A Reference Manual, 2nd edition. New Jersey, USA: Prentice Hall International Series in Computer Science.

20. Stepanov A.A., Ros D.E. (2015). From Mathematics to Generic Programming. London, UK: Addison-Wesley.

21. Wing J., Liskov B. (1994). Family Values: A Behavioral Notion of Subtyping. *ACM Trans. Program. Lang. Syst., 16(6), 1812-1841.*

22. The RAISE Language Group (1992). The RAISE SPECIFICATION LANGUAGE. Kongens Lyngby, Denmark: Prentice Hall Europe.

23. Guttag, J.V. and Horning, J.J. The algebraic specifications of abstract data types. Retrieved from https://www.semanticscholar.org/paper/The-algebraic-specification-of-abstract-data-types-Guttag- Horning/e4c8b1db0c839a07a833db51c5ac00e6ffd5a922.

УДК 003.26:004.056.5
DOI https://doi.org/10.32689/maup.it.2023.5.8

Володимир БРОДКЕВИЧ
кандидат економічних наук, доцент кафедри комп'ютерно-інформаційних систем і технологій Міжрегіональної Академії управління персоналом, вул. Фрометівська, 2, Київ, Україна, індекс 03039 (v.brodkevych@gmail.com)
ORCID: 0000-0003-4282-8888

Дарина ЯРЕМЕНКО
викладач кафедри комп'ютерних наук та інтелектуальних систем Міжрегіональної Академії управління персоналом, вул. Фрометівська, 2, Київ, Україна, індекс 03039 (dashayaremenko17@gmail.com)
ORCID: 0000-0002-6294-9698

Віталій КИРИЧЕНКО
аспірант кафедри комп'ютерно-інформаційних систем і технологій Міжрегіональної Академії управління персоналом, вул. Фрометівська, 2, Київ, Україна, індекс 03039 (vp_kirichenko@ukr.net)
ORCID: 0009-0005-5411-4315

Андрій ШЛАПАК
аспірант кафедри комп'ютерно-інформаційних систем і технологій Міжрегіональної Академії управління персоналом, вул. Фрометівська, 2, Київ, Україна, індекс 03039 (Andreii.shlapak@gmail.com)
ORCID: 0009-0001-7563-4871

Олег ТИЩЕНКО
аспірант кафедри комп'ютерно-інформаційних систем і технологій Міжрегіональної Академії управління персоналом, вул. Фрометівська, 2, Київ, Україна, індекс 03039 (0987651234um@gmail.com)
ORCID: 0009-0001-2763-579X

Volodymyr BRODKEVYCH
Candidate of Economic Sciences, Associate Professor at the Department of Computer Information Systems and Technologies of the Interregional Academy of Personnel Management, 2, Frometivska St, Kyiv, Ukraine postal code 03039 (v.brodkevych @gmail.com)

Daryna YAREMENKO
Lecturer at the Department of Computer Science and Intelligent Systems of the Interregional Academy of Personnel Management, 2, Frometivska St, Kyiv, Ukraine postal code 03039 (dashayaremenko17@gmail.com)

Vitalii KYRYCHENKO
Postgraduate Student at the Department of Computer Information Systems and Technologies of the Interregional Academy of Personnel Management, 2, Frometivska St, Kyiv, Ukraine postal code 03039 (vp_kirichenko@ukr.net)

Andrii SHLAPAK
Postgraduate Student at the Department of Computer Information Systems and Technologies of the Interregional Academy of Personnel Management, 2, Frometivska St, Kyiv, Ukraine postal code 03039 (Andreii.shlapak@gmail.com)

Oleh TYSHCHENKO
Postgraduate Student at the Department of Computer Information Systems and Technologies of the Interregional Academy of Personnel Management, 2, Frometivska St, Kyiv, Ukraine postal code 03039 (0987651234um@gmail.com)

Бібліографічний опис статті: Бродкевич, В., Яременко, Д., Кириченко, В., Шлапак, А., Тищенко О. (2023). Застосування шифрування даних в управлінській діяльності. *Інформаційні технології та суспільство,* 5 (11), 60–66. DOI: https://doi.org/10.32689/maup.it.2023.5.8

Bibliographic description of the article: Brodrevych, V., Yaremenko, D., Kyrychenko, V., Shlapak, A., Tyshchenko, O. (2023). Zastosuvannia shyfruvannia danykh v upravlinskii diialnosti [Application of data encryption in administrative activities]. *Informatsiini tekhnolohii ta suspilstvo – Information technology and society,* 5 (11), 60–66. DOI: https://doi.org/10.32689/maup.it.2023.5.8

ЗАСТОСУВАННЯ ШИФРУВАННЯ ДАНИХ В УПРАВЛІНСЬКІЙ ДІЯЛЬНОСТІ

Анотація. *У сучасному цифровому світі, де велика частина бізнес-операцій і обміну даними відбувається в електронному форматі, шифрування стає незамінним інструментом для забезпечення безпеки. Дана стаття присвячена опису розробленого прикладного програмного забезпечення, що реалізує алгоритми шифрування даних. Програмний застосунок може бути використаний в управлінській діяльності для забезпечення достатнього рівня захисту даних, враховуючи потенційні загрози. До чутливих даних компанії, що потребують особливої уваги з точки зору безпеки відносяться наступні: комерційна таємниця, фінансова інформація, внутрішня кореспонденція, персональні дані співробітників й клієнтів тощо. Шифрування цих типів даних дозволяє компаніям захистити свою комерційну інформацію, знизити ризик фінансових та репутаційних втрат. При розробці програмного забезпечення використовувалися алгоритми ChaCha20 і Poly1305 в декілька ключових етапів. Спочатку реалізовано основні функції, такі як QuarterRound і ChaChaBlock, які виконують перетворення стану ChaCha20. Далі створено механізм шифрування відкритого тексту, розбивши його на блоки і використовуючи XOR для обробки кожного блоку згенерованим потоком ключів. Крім того, імплементовано функцію аутентифікації повідомлень Poly1305, що генерує тег для перевірки цілісності даних. Завершальним етапом було інтегрування обох частин системи – шифрування та аутентифікації, щоб забезпечити конфіденційність та цілісність переданих даних. Також проведено тестування розробленого програмного забезпечення, що показало коректність його роботи. Розроблений застосунок легко інтегрується в майже будь-яку IT-інфраструктуру компанії, може працювати в реальному часі для шифрування внутрішньої кореспонденції або повідомлені по мережі компанії. Завдяки відкритому коду, програмне забезпечення може буде вдосконалено під умови замовника (наприклад для шифрування документів в різних форматах для довготривалого збереження та/або переказу по відкритому каналу зв'язку).*

Ключові слова: *прикладне програмне забезпечення, комерційна таємниця, шифрування даних, цілісність та автентичність.*

APPLICATION OF DATA ENCRYPTION IN MANAGEMENT ACTIVITIES

Abstract*. In today's digital world, where a significant portion of business operations and data exchange takes place in electronic format, encryption becomes an indispensable tool for ensuring security. This article is dedicated to the description of the developed application software that implements data encryption algorithms. The software application can be used in managerial activities to provide an adequate level of data protection, considering potential threats. Sensitive company data that require special attention in terms of security include the following: trade secrets, financial information, internal correspondence, as well as personal data of employees and clients, etc. Encrypting these types of data allows companies to protect their commercial information, reduce the risk of financial and reputational losses. In developing the software, the ChaCha20 and Poly1305 algorithms were used at several key stages. Initially, the core functions, such as QuarterRound and ChaChaBlock, which perform the transformation of the ChaCha20 state, were implemented. Then, a mechanism for encrypting plain text was created by dividing it into blocks and using XOR to process each block with the generated key stream. In addition, the Poly1305 message authentication function, which generates a tag for verifying data integrity, was implemented. The final stage was the integration of both parts of the system – encryption and authentication, to ensure the confidentiality and integrity of the transmitted data. Testing of the developed software was also conducted, demonstrating its correct operation. The developed application is easily integrated into almost any company's IT infrastructure, can operate in real-time for encrypting internal correspondence or company network messages. Thanks to the open-source code, the software can be refined under customer conditions (for example, for encrypting documents in various formats for long-term storage and/or transmission over an open communication channel).*

Key words**:** *software, encryptions algorytms, trade secrets, finance information, personal data.*

На сьогодні у сфері управлінської діяльності існує декілька різновидів «чутливих» даних, що потребують ретельного збереження та/або переказу. До таких видів можна віднести:

– персональні дані співробітників (ім'я, адреса, номер соціального страхування, банківські реквізити, медична інформація тощо);

– фінансова інформація (банківські рахунки компанії, звіти про прибутки та збитки, інвестиційні стратегії, аудиторські висновки тощо);

– внутрішня кореспонденція та документація (електронні листи, звіти, протоколи зборів, внутрішні настанови та політики);

– дані про клієнтів (контактна інформація, історія покупок, персональні уподобання та потреби);

– договори та угоди (контракти з партнерами, постачальниками, клієнтами, умови ліцензійних угод);

– дані про продукцію та послуги (описи, технічні характеристики, ціни, плани виробництва).

Особливу увагу також слід приділяти комерційній таємниці. Комерційна таємниця є ключовим активом для будь-якої організації, оскільки вона включає в себе відомості, що мають комерційну вартість через те, що вони невідомі широкому колу осіб і перед якими їх власник здійснює заходи щодо збереження конфіденційності. Це можуть бути формули, рецепти, проектні документи, стратегії розвитку, бази даних клієнтів, виробничі секрети, унікальні рецептури, технологічні процеси, маркетингові

стратегії та будь-яка інша інформація, яка допомагає компанії зберегти та посилити свої конкурентні переваги.

Захист комерційної таємниці має критичне значення. Наведемо декілька аргументів щодо цього ствердження. По перше це конкурентна перевага на ринку. Унікальна інформація, яка не доступна конкурентам, може надавати значну перевагу, дозволяючи пропонувати унікальні продукти чи послуги.

Другим аспектом може бути фінансова стабільність. Інформація, що становить комерційну таємницю, може впливати на доходи та рентабельність компанії. Її втрата або несанкціонований доступ може призвести до фінансових збитків. Також слід відмітити інвестиції акціонерів та подальший розвиток компанії. Компанії, що інвестують у дослідження та розробку, для захисту своїх інвестицій потребують гарантій того, що результати цих діяльностей залишаться винятково у їх розпорядженні.

Також слід відмітити законодавчі вимоги. В багатьох юрисдикціях існують законодавчі акти, що зобов'язують компанії захищати персональні дані клієнтів та іншу конфіденційну інформацію. Отже, задача захисту даних в управлінської діяльності є актуальною. Сучасні методи захисту повинні бути комплексними й охоплювати як технічні, так і організаційні заходи. До них ми будемо відносити наступні:

Шифрування даних. Використання сучасних методів шифрування для захисту електронних документів та баз даних забезпечує, що інформація залишається недоступною для несанкціонованих осіб.

- Контроль доступу. Обмеження доступу до інформації через фізичні та електронні системи контролю доступу дозволяє забезпечити, що тільки уповноважені особи мають доступ до комерційних таємниць.
- Юридичні заходи. Використання конфіденційних угод (NDA), трудових контрактів та інших юридичних інструментів допомагає захистити інформацію на законодавчому рівні.
- Фізична безпека. Захист фізичних носіїв інформації та важливих об'єктів компанії, таких як офіси, виробничі площі та лабораторії, є необхідним елементом загальної стратегії безпеки.

Дана стаття пропонує до уваги опис програмного рішення алгоритму шифрування даних. Після проведеного огляду сучасних алгоритмів шифрування [1, с. 48-51] було обрано шифр Шифр ChaCha20. Розглянемо його детальніше.

Шифр ChaCha20 – це високошвидкісний потіківий шифр, який був спочатку описаний у документі [2, с. 78-81]. Цей шифр є значно швидшим, ніж AES [3, с. 21] у програмних реалізаціях, що робить його близько втричі швидшим на платформах, де відсутнє спеціалізоване обладнання AES. Крім того, ChaCha20 не чутливий до атак з урахуванням часу. Щодо практичного застосування – ChaCha20 широко використовується в сучасних криптографічних протоколах, таких як TLS, SSH, IPsec [2, с. 6] та інші, як альтернатива AES.

Для досягнення нашої мети будемо імплементувати функцію аутентифікації повідомлень Poly1305, що генерує тег для перевірки цілісності даних. Під час цього етапу будемо використовувати бібліотеку NACL для створення тега Poly1305.

Аутентифікатор Poly1305 – це високошвидкісний аутентифікатор повідомлень, який використовується для перевірки цілісності та автентифікації повідомлень [4, с. 2]. Його реалізація також досить проста та не вимагає спеціальних обчислювальних ресурсів. Poly1305 часто використовується разом з різними шифрами для забезпечення конфіденційності та цілісності даних, зокрема в AEAD конструкціях.

Розглянемо конструкцію CHACHA20-POLY1305 AEAD. Це аутентифікована шифрувальна конструкція з асоційованими даними (AEAD), яка комбінує шифр ChaCha20 і аутентифікатор Poly1305 для забезпечення конфіденційності, цілісності та автентифікації повідомлень та їх асоційованих даних. Ця конструкція широко використовується в сучасних протоколах безпеки, наприклад, таких як TLS 1.3, з метою захисту комунікацій в мережах Інтернету.

Розглянемо більш детально роботу алгоритму. Алгоритм ChaCha20 використовує функцію блока для перетворення стану шляхом виконання кількох чвертей обертання.

Вхідними параметрами алгоритму є:

– 256-бітний ключ, який розглядається як конкатенація восьми 32-бітних малих ендіанів;

– 96-бітний нонс, який розглядається як конкатенація трьох 32-бітних малих ендіанів;

– 32-бітний параметр кількості блоків, який розглядається як 32-бітний малий ендіан.

Вихідним значенням алгоритму є 64 випадкових байтів.

Початковий стан ChaCha20 ініціалізується наступним чином:

– Перші чотири слова (0-3) є константами: 0x61707865, 0x3320646e, 0x79622d32, 0x6b206574.

– Наступні вісім слів (4-11) беруться з 256-бітного ключа, читаючи байти в малих ендіанів, в 4-байтових чанках.

– Слово 12 є лічильником блоків. Оскільки кожен блок має розмір 64 байти, 32-бітне слово достатньо для 256 гігабайтів даних.

– Слова 13-15 є нонсом, який МАЄ не повторюватися для одного ключа. 13-те слово є першими 32 бітами вхідного нонса, взятими як малий ендіан, тоді як 15-те слово є останніми 32 бітами.

Алгоритм ChaCha20, як видно з назви, складається з 20 раундів, які чергуються між «колонковими раундами» та «діагональними раундами». Кожен раунд складається з чотирьох чвертей обертання, і вони виконуються наступним чином. Чверті обертання 1-4 є частиною «колонкового» раунду, тоді як 5-8 є частиною «діагонального» раунду.

На кінці 20 раундів (або 10 ітерацій вищезазначеного списку) ми додаємо початкові вхідні слова до вихідних слів і серіалізуємо результат, впорядкувавши слова одне за одним в малих ендіанах.

Poly1305 – це одноразовий аутентифікатор, розроблений D. J. Bernstein [4, с. 3]. Poly1305 приймає 32-байтний одноразовий ключ і повідомлення, і генерує 16-байтний тег. Цей тег ми будемо використовувати для аутентифікації повідомлення. Poly1305 має назву «Код аутентифікації повідомлень Poly1305-AES", і там функція MAC вимагає 128-бітний ключ AES, 128-бітний «додатковий ключ» і 128-бітний (не секретний) nonce. Алгорим AES використовується там для шифрування nonce, щоб отримати унікальний (і секретний) 128-бітний рядок. При необхідності можна замінити AES на довільну ключову функцію з довільним набором nonce до 16-байтних рядків [5, с. 20].

Незалежно від того, як створюється ключ, ключ розділяється на дві частини, які називаються "r" і "s". Пара (r, s) повинна бути унікальною і НЕПРЕДСКАЗУЄМОЮ для кожного виклику (тому вона спочатку отримується шифруванням nonce), в той час як "r" МОЖЕ бути сталим, але потребує змін, перш ніж використовуватися. ("r" трактується як 16-октетне число little-endian):

– r[3], r[7], r[11] і r[15] повинні мати свої верхні чотири біти встановлені в нуль (бути меншими за 16);

– r[4], r[8] і r[12] повинні мати свої нижні два біти встановлені в нуль (бути кратними 4).

Вхідними параметрами для Poly1305 є:

– 256-бітний одноразовий ключ;

– Повідомлення довільної довжини;

Вихідне значення – це 128-бітний тег.

Спочатку значення "r" стискається.

Далі, встановлюється постійне просте число "P", яке дорівнює 2^130-5: 3ffffffffffffffffffffffffffffffffb. Також встановлюється змінну "accumulator" на нуль.

Після цього повідомлення розділяється на блоки по 16 байт. Останній може бути коротшим:

– Читаємо блок як число little-endian.

– Додаємо один біт поза числом октетів. Для блоку з 16 байт це еквівалентно додаванню 2^128 до числа. Для коротшого блоку це може бути 2^120, 2^112 або будь-яка ступінь двійки, яка рівномірно ділиться на 8, аж до 2^8.

– Якщо блок не має довжини 17 байтів (останній блок), заповнюємо його нулями. Це не має значення, якщо ми трактуємо блоки як числа.

– Додаємо це число до аккумулятора.

– Помножуємо на "r".

– Встановлюємо аккумулятор на результат modulo p. Загалом: Acc = ((Acc + block) * r) % p.

Нарешті, значення секретного ключа "s" додається до аккумулятора, а 128 менш значущих бітів серіалізуються у порядку little-endian для формування тегу.

Далі розглянемо опис програмного забезпечення.

При розробці програмного забезпечення для шифрування даних управлінської діяльності, спочатку було реалізовано функцію QuarterRound. Ця функція виконує одну ітерацію четвертого обертання в алгоритмі ChaCha20. Потім виконано реалізацію функції ChaChaBlock, яка використовує кілька ітерацій QuarterRound для перетворення стану ChaCha.

Ця функція приймає ключ, nonce та лічильник блоків як вхідні параметри і генерує 64 байти випадково виглядаючих даних.

Наступним кроком стала реалізація функції для шифрування відкритого тексту. Вхідний текст було розбито на блоки і застосована операція XOR зі згенерованим потоком ключів. Після успішного шифрування знадобилось також забезпечити аутентифікацію повідомлення.

Для цього було використано бібліотеку NACL для створення тега Poly1305. Під час шифрування згенерований 32-бітний ключ та зашифрований текст передавався в функцію Poly1305, щоб згенерувати тег. Варто зазначити, що під час розшифрування ця функція приймала «відкритий текст», який насправді є зашифрований, аби перевірити, чи не був він змінений під час транспортування.

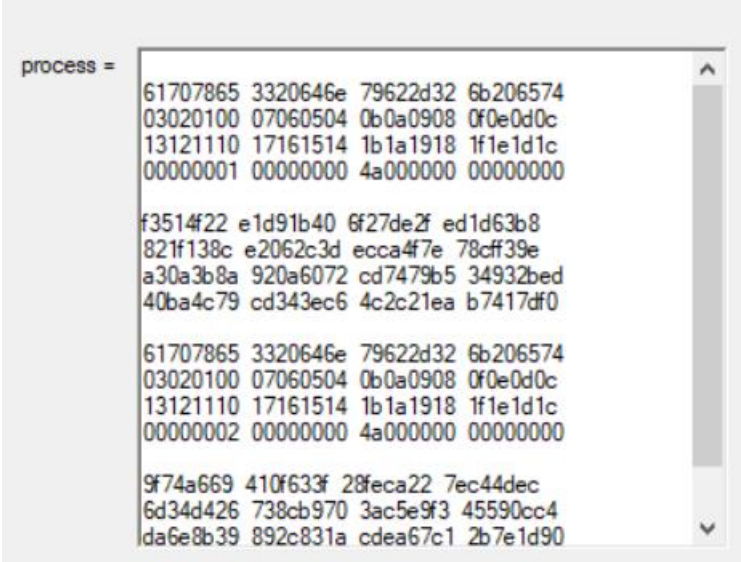

Рис. 1. Вхідні блоки в ChaCha20Block та після проходу 20 раундів

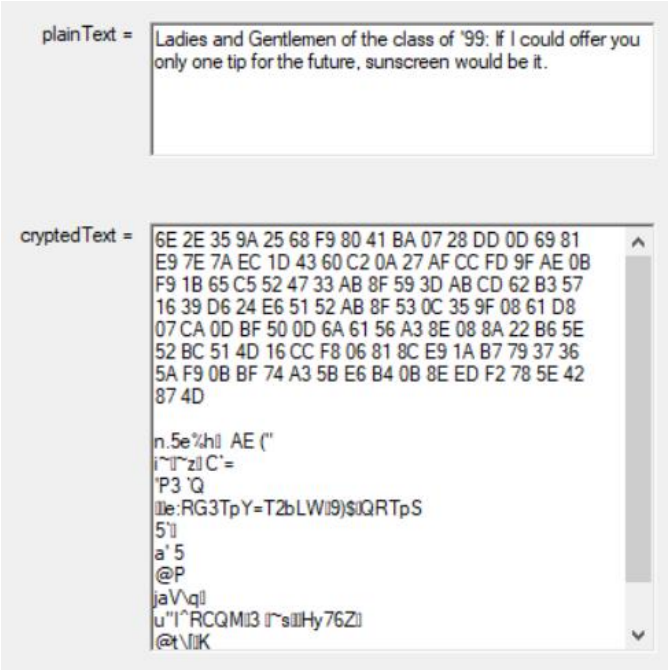

Рис. 2. Відкритий текст та зашифрований

Висновки

1. Шифрування цих типів даних дозволяє компаніям захистити свою комерційну інформацію, знизити ризик фінансових та репутаційних втрат. У сучасному цифровому світі, де велика частина бізнес-операцій і обміну даними відбувається в електронному форматі, шифрування стає незамінним інструментом для забезпечення безпеки.

2. Розроблено прикладне програмне забезпечення для шифрування конфеденційної інформації для управлінської діяльності. Проведено його тестування, показано коректність та продуктивність роботи алгориитмів.

3. Дане програмне забезпечення дозволяє забезпечити достатній рівень захисту даних, враховуючи потенційні загрози. Сумісне з ІТ-інфраструктурою будь-якої компанії та може бути змінено під потреби замовника завдяки відкритому коду.

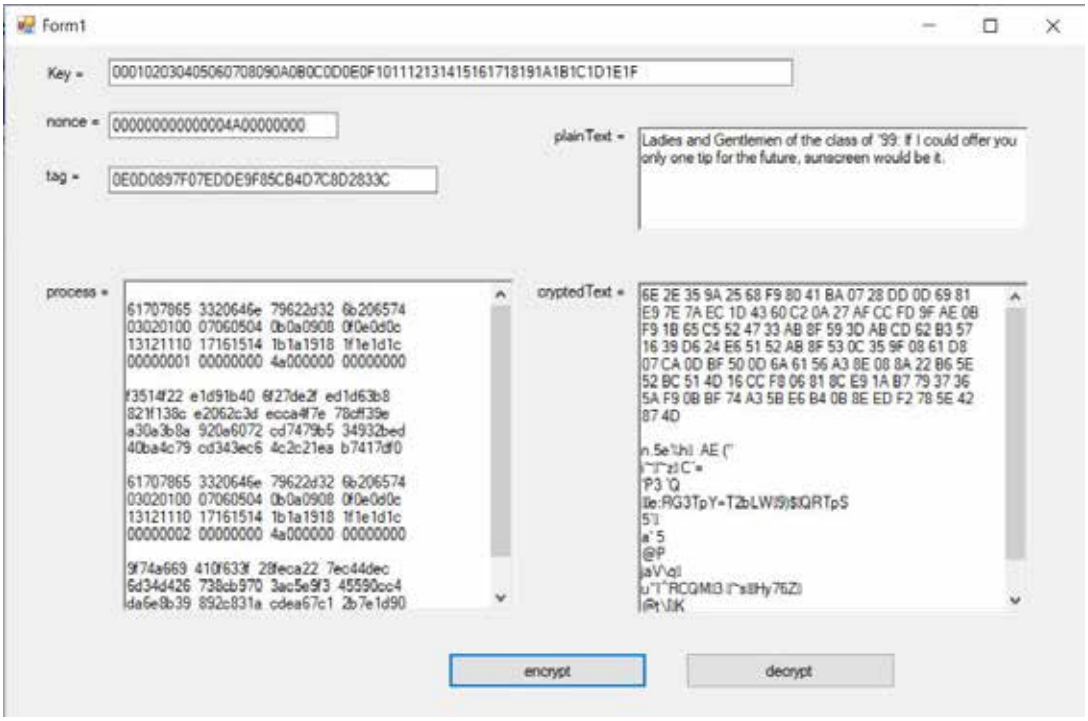

Рис. 3. Шифрування тексту

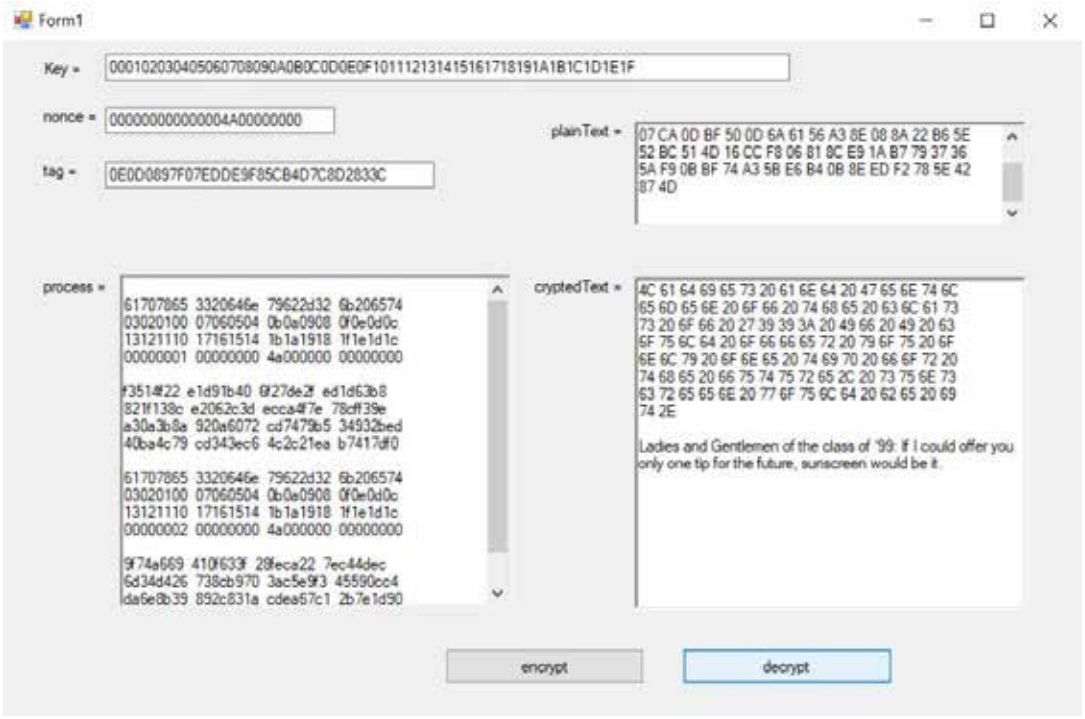

Рис. 4. Розшифрування тексту, перевірка тегу

Під час розробки програми для шифрування і аутентифікації повідомлень з використанням алгоритмів ChaCha20 і Poly1305 я вивчила та реалізувала кілька ключових етапів. Спочатку я розробила основні функції, такі як QuarterRound і ChaChaBlock, які виконують перетворення стану ChaCha20. Далі я створила механізм шифрування відкритого тексту, розбивши його на блоки і використовуючи XOR для обробки кожного блоку згенерованим потоком ключів.

Список використаних джерел:

1. Горбенко І. Д. Прикладна криптологія: Теорія. Практика. Застосування / І. Д. Горбенко, Ю. І. Горбенко. Харків: Форт, 2013. 80.
2. Совин Я. Р., Хома В. В., Отенко В. І., Порівняння AEAD-алгоритмів для вбудованих систем інтернету речей. 2019. с. 76-91. URL: https://science.lpnu.ua/sites/default/files/journal-paper/2020/feb/21055/var1ksm-19-78-93.pdf
3. AES Encrypter/Decrypter [Електронний ресурс]: ECE 5760: Final Project / A. Laxminarayana, A. Ravani, M. Venkatraman. URL: http://people.ece.cornell.edu/land/courses/ece5760/FinalProjects/s2015/ar856/ECE560webpage/ECE5760%20webpage/webpage_file s.html
4. Bernstein, D.J.: Stronger security bounds for wegman-carter-shoup authenticators. In: Cramer, R. (ed.) EUROCRYPT 2005. LNCS, vol. 3494, pp. 164–180. Springer, Heidelberg. 2005. http://cr.yp.to/papers.html#securitywcs, ID2d603727f69542f30f7da2832240c1ad
5. Nir Y. ChaCha20 and Poly1305 for IETF Protocols [Електронний ресурс] / Y. Nir, A. Langley // Google, Inc. 2018.

References:

1. Horbenko I. D. (2013). Applied cryptology: Theory. Practice. Application / I. D. Horbenko, Yu. I. Horbenko. Kharkiv: Fort. 880.
2. Sovin Y. R., Khoma V. V., Otenko V. I. (2019). Comparison of AEAD algorithms for embedded systems of the Internet of Things. pp. 76-91. Retrieved from https://science.lpnu.ua/sites/default/files/journal-paper/2020/feb/21055/var1ksm-19-78-93.pdf
3. AES Encrypter/Decrypter [Electronic resource]: ECE 5760: Final Project / A. Laxminarayana, A. Ravani, M. Venkatraman. Retrieved from http://people.ece.cornell.edu/land/courses/ece5760/FinalProjects/s2015/ar856/ECE560webpage/ ECE5760%20webpage /webpage_file s.html
4. Bernstein, D.J. (2005). Stronger security bounds for wegman-carter-shop authenticators. In: Cramer, R. (ed.) EUROCRYPT 2005. LNCS, vol. 3494, pp. 164–180. Springer, Heidelberg. Retrieved from http://cr.yp.to/papers.html#securitywcs,ID2d603727f69542f30f7da2832240c1ad
5. Nir Y. (2018). ChaCha20 and Poly1305 for IETF Protocols [Electronic resource] / Y. Nir, A. Langley // Google, Inc.

НОТАТКИ

НАУКОВЕ ВИДАННЯ

ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ ТА СУСПІЛЬСТВО

INFORMATION TECHNOLOGY AND SOCIETY

ВИПУСК 5 (11)
ISSUE 5 (11)

2023

Коректура
Ірина Чудеснова

Комп'ютерна верстка
Андрій Філатов

Формат 60x84/8. Гарнітура Cambria.
Папір офсет. Цифровий друк. Ум. друк. арк. 7,91. Замов. № 0324/194. Наклад 300 прим.

Видавництво і друкарня – Видавничий дім «Гельветика»
65101, Україна, м. Одеса, вул. Інглезі, 6/1
Телефон +38 (095) 934 48 28, +38 (097) 723 06 08
E-mail: mailbox@helvetica.ua
Свідоцтво суб'єкта видавничої справи
ДК No 7623 від 22.06.2022 р.