

ЛАВРЕНЧУК С. В.

Луцький національний технічний університет  
ORCID ID: 0000-0002-5453-3924  
lavrsveet@gmail.com

ЗДОЛБИЦЬКА Н. В.

Луцький національний технічний університет  
ORCID ID: 0000-0002-1345-3581  
ninazdolb@gmail.com

ХАМУЛА Н. М.

Луцький національний технічний університет  
nadya.synuyk0907@gmail.com

## ПРОГРАМНИЙ КОМПЛЕКС ДЛЯ ВІЗУАЛІЗАЦІЇ АЛГОРИТМІВ НА ГРАФАХ

Розглянуто алгоритми на графах, що представлені за допомогою різних структур даних. Програмний комплекс має модульний web-інтерфейс. Реалізовано представлення графа як множини вершин у вигляді пронумерованих кругів та зв'язків між ними (графічне зображення); за допомогою динамічних зв'язаних списків (списки суміжності); у вигляді матриці суміжності. Користувачі даного проекту мають змогу інтерактивно впливати та спостерігати за покроковим виконанням алгоритмів на графах, що дасть змогу набуття дослідницьких навичок та компетентностей в області використання дискретних структур.

Ключові слова: граф, дискретні структури, візуалізація, способи подання графа, програмний комплекс, алгоритм Прима, алгоритм Дейкстри, алгоритм Крускала, алгоритм Флойда, пошук вглиб, пошук вшир.

SVITLANA LAVRECHUK, NINA ZDOLBITSKA, NADIIA KHAMULA  
Luts National Technical University

## SOFTWARE COMPLEX FOR GRAPH ALGORITHMS VISUALIZATION

Algorithms on graphs represented by graphical structures are offered. The software complex has a modular web interface. A representative graph is implemented as a set of vertices in the form of numbered circles and links between them (graphic image); using dynamically linked lists (adjacency lists); using an adjacency matrix. The project of this project allows the use of interactive algorithms for step-by-step calculations and algorithms on graphical images to obtain the necessary research results and competencies in the use of discrete structures. The project was implemented using HTML, CSS, JavaScript, which allows visualizing the application and interactively working with algorithms on graphs represented by different data structures. Graphics File Algorithm Programming provides web-based and interactive algorithms created by the used DHTML itself, creating a software project in the form of a site. Each page is dedicated to a separate algorithm and structurally consists of a header, container, footer. The website design uses HTML and cascading CSS stylesheets, to create an interactive parsing process and to transform methods in graphic files – based on JavaScript, which allows third-party scripts to be processed and rendered. The user can vibrate the image type (for orientation), the number of nodes, the presentation method, generate this graph, indicate the starting point for starting the search algorithm, observe the operational operation of the algorithm. The user can adjust the animation speed. The development of a set of programs is meant for interactive demonstration and visualization of the operation of algorithms in the study of graph theory.

Keywords: graph, discrete structures, visualization, methods of graph representation, software complex, Prim's algorithm, Dijkstra's algorithm, Kruskal's algorithm, Floyd's algorithm, depth search, breadth search.

### Постановка проблеми

Теорія графів використовується в багатьох сферах для дослідження множини об'єктів (подій або процесів) та відношень між ними, адже вона дозволяє змоделювати та інтерпретувати окремі явища, реалізувати їх наочне представлення без необхідності фізичного проектування. Візуалізація роботи алгоритмів на графах – актуальне завдання, яке має практичну та наукову цінність, оскільки такі алгоритми використовуються для дослідження проблем в різних галузях науки та виробництва.

Розглянувши алгоритми на графах, що представлені за допомогою різних структур даних, запропоновано створити програмний комплекс, що матиме модульний web-інтерфейс. Необхідно реалізувати представлення графа як множини вершин у вигляді пронумерованих кругів та зв'язків між ними (графічне зображення); за допомогою динамічних зв'язаних списків (списки суміжності); у вигляді матриці суміжності. Користувачі даного проекту матимуть змогу інтерактивно впливати та спостерігати за покроковим виконанням алгоритмів на графах, що дасть можливість набуття дослідницьких навичок та компетентностей в області використання дискретних структур.

### Аналіз останніх джерел

Теорія графів, як розділ дискретної математики, застосовується для моделювання різних (лінійних та нелінійних) зв'язків та залежностей між об'єктами [1]. Ця теорія має велике практичне значення, адже вона використовується при розв'язуванні задач кібернетики, програмування [2–4], теорії автоматів та теорії ігор, робототехніки та електротехніки [5, 6], штучного інтелекту [7], стеганографії [8, 9], картографії, економіко-математичних задач, логістики [10], оптимізації [10, 11], планування і керування виробництвом [12].

Застосування графів дає змогу змоделювати та дослідити генеалогічні зв'язки, взаємодії в соціальних мережах [13], спроектувати електричні або комп'ютерні мережі, телекомунікаційні системи,

схеми водопостачання або електрифікації. Інструментарій теорії графів дозволяє проаналізувати взаємозв'язки та взаємовпливи між об'єктами при дослідженні екосистем, ієрархії живих організмів, ланцюгів харчування, генетичних закономірностей, поширення хвороб; розповсюдження заборонених речовин; зобразити моделі конкуренції, корупції [14]. За допомогою графів здійснюється інтерпретація різноманітних проблем теорії управління, теорії інформації, тощо.

Широке застосування алгоритмів на графах зумовлює потребу в розробці програмних засобів для візуалізації їх роботи. Розглянемо найвідоміші програми, які призначені для візуального представлення об'єктів дослідження у вигляді графів.

Програма GraphBuilder [15] дозволяє візуалізувати поширені алгоритми на графах: Флойда-Воршелла, Дейкстри, Прима, Крускала. У процесі візуалізації відображається номер ітерації, відбувається підсвічування вершин і ребер різними кольорами.

Графічний редактор графів програми MaxFlow [16] дає змогу зобразити вершини та дуги орієнтованого графа, вказати назви вершин та ваги дуг, при потребі можна переміщувати елементи графа. Створений граф зберігається в txt-файлі як список суміжності. У програмі реалізовано наступні алгоритми пошуку: найкоротшого шляху, максимального потоку, оптимального шляху (найкоротший шлях із максимальним потоком), компонент сильної зв'язності, мінімального кістякового дерева.

За допомогою візуального середовища Графоаналізатор [17] можна вирішити багато інженерних задач, адже воно містить близько 20 алгоритмів обробки графів. Це проект з відкритим кодом, що дає змогу користувачам розширювати функціонал та адаптувати його для розв'язування власних задач.

Програма Graph Magics [18] містить функції для ефективної побудови та модифікації графів, реалізовано 17 різних алгоритмів, є можливість експорту та імпорту графів, які можна подати у вигляді різних структур: список ребер, список суміжності, матриця суміжності.

Існуючі програми візуалізації роботи алгоритмів на графах працюють під ОС Windows, що ускладнює можливість їх використання на різних девайсах. Тому метою роботи є створення кросплатформенного програмного комплексу з web-інтерфейсом, що дає змогу інтерактивно відобразити роботу алгоритмів на графах, представляти їх за допомогою різних структур даних.

### Виклад основного матеріалу

Щоб проект був web-орієнтованим та інтерактивним, використаємо DHTML. Застосування HTML та CSS для оформлення сайту дозволить створювати web-сторінки різних форм та розмірів. Для реалізації алгоритмів на графах, забезпечення інтерактивності візуалізації та взаємодії користувача з проектом використовуватимемо JavaScript, яка є динамічною, об'єктно-орієнтованою та забезпечує виконання сценаріїв на стороні клієнта.

Структура проекту модульна, функціонально він складається з таких частин:

- html-файли:
  - головна сторінка, що містить меню сайту;
  - інформація про проект;
  - інтерфейс кожного алгоритму забезпечують 6 html-файлів;
- 2 CSS-файли для форматування зовнішнього вигляду модулів;
- JavaScript-файли:
  - реалізація логіки алгоритмів – 6 файлів;
  - елементи візуалізації (круги, лінії, тощо) – 5 файлів;
  - керування процесом анімації створено – 9 файлів.

Відповідно до принципів динамічного програмування, html-сторінки інкапсулюють скрипти для забезпечення функціоналу та анімації процесу візуалізації. Скрипти, що забезпечують динаміку візуалізації окремих елементів, є спільними для всіх сторінок. Для реалізації логіки роботи кожного алгоритму створено окремий скрипт, який підключено до файлу інтерфейсу алгоритму (наприклад, до файлу DFS.html підключено DFS.js).

На рис. 1 зображено вигляд головної сторінки проекту, а на рис. 2 – способи подання графу в програмі: графічний (вузли та дуги); матриця суміжності; динамічні зв'язані списки (списки суміжності).



Рис. 1. Головна сторінка проекту

Інтерфейс типової сторінки сайту наведено на рис. 3. Користувачеві надається можливість вибирати тип графа (неорієнтований або орієнтований) та спосіб його подання (граф, списки суміжності, матриця

суміжності), згенерувати новий граф та запустити алгоритм візуалізації. Для алгоритмів пошуку найкоротшого шляху або обходу вершин графа також можна вказати початкову вершину. Можна керувати візуалізацією роботи алгоритму та змінювати швидкість анімації.

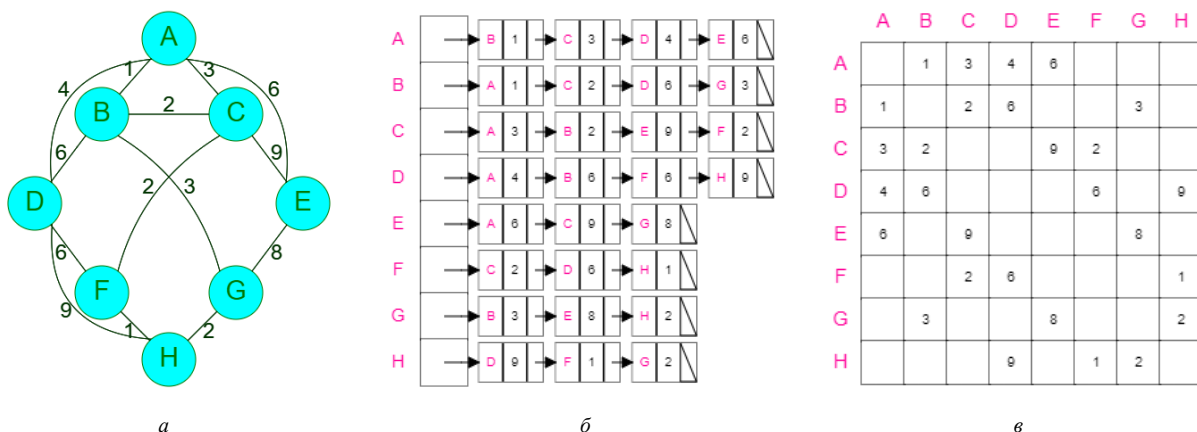


Рис. 2. Приклади подання зваженого графу в програмі: а) графічне зображення; б) списки суміжності; в) матриця суміжності

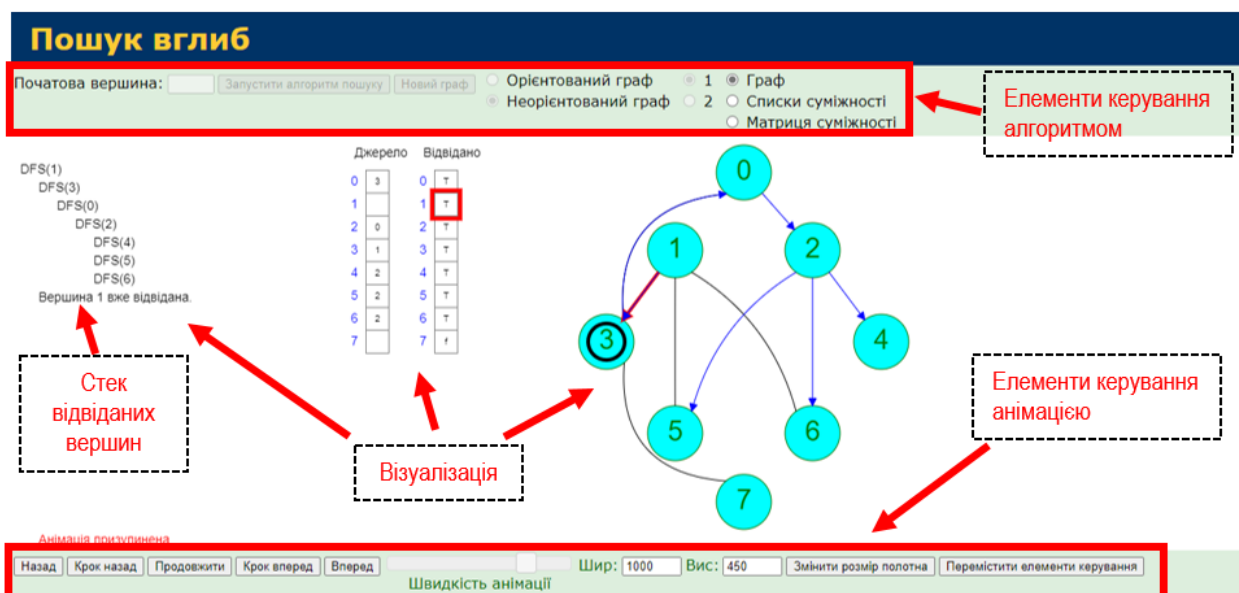


Рис. 3. Інтерфейс типової сторінки сайту

Результатом обходу графа вглиб або вишир є каркасне дерево (граф без циклів), у процесі побудови якого підсвічуються кольором активна вершина та ребро, по якому відбувається обхід у конкретний момент часу (див. рис.3).

Прикладом практичного застосування такого алгоритму обходу графа [6] є проходження роботом лабіринту (рис. 4). Щоб запрограмувати алгоритм руху робота, необхідно зробити модель лабіринту (рис. 5). Як бачимо з рис. 5, щоб здійснити проходження по даному лабіринту, достатньо пройти вершини в наступному порядку: 1 (вхід), 3, 0, 2, 6 (вихід), а вершини 4, 5 та 7 є тупиковими і їх відвідувати не потрібно (відповідно немає необхідності роботу заходити в ці частини лабіринту). Порядок проходження елементів лабіринту (пронумерованих вузлів графа) також відображається в стеку відвіданих вершин (див. рис. 3), який динамічно формується під час процесу візуалізації роботи алгоритму пошуку вглиб.

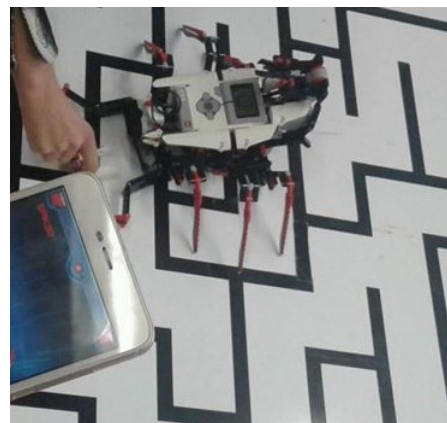


Рис. 4. Рух робота по лабіринту

У програмному комплексі реалізовано також відомі алгоритми побудови мінімального кістякового дерева: Крускала та Прима. Продемонструємо варіант візуалізації алгоритму Крускала, відповідно до якого на першому етапі будується вироджений ліс, який складається з N одновершинних дерев. На другому етапі відбувається

об'єднання двох дерев за ознакою найменших ребер (для цього в програмі схематично позначаємо впорядковані ребра – рис. 6, а) до тих пір, поки не отримаємо мінімальне кістякове дерево. На рис. 6, б наведено результат знаходження мінімального кістякового дерева. Як видно з рис. 6, залишилися не задіяні ребра, що дало змогу уникнути циклів у графі, але при цьому відвідати всі вершини.

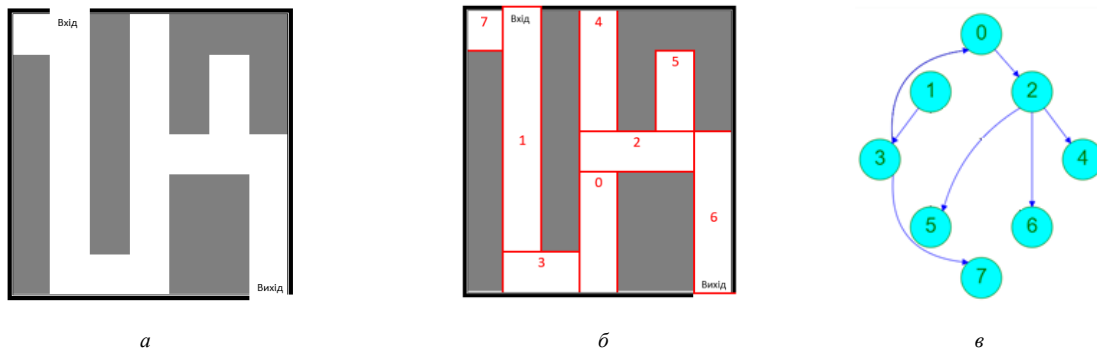


Рис. 5. Моделювання лабіринту за допомогою графа: а) лабіринт; б) дискретизація шляху; в) граф переходів

**Алгоритм Крускала**

Залучити алгоритм   1 Граф  
 2 Списки суміжності  
 Матриця суміжності

Мін. шлях(B) = D    Мін. шлях(G) = G  
 Вершини в різних деревах. Додати ребро до дерева: Об'єднання (D, G)

Непересічні множини

A	D
B	D
C	D
D	4
E	-1
F	-1
G	-1
H	-1

A	6	E
B	6	D
C	6	H
D	4	G
E	4	F
F	4	E
G	8	F
H	5	F
H	5	G
H	5	H

Анімація призулилення

Назад Крок назад Продовжити Крок вперед Вперед Шир: 1000 Швидкість анімації

Рис. 6. Алгоритм Крускала: а) процес візуалізації; б) результат роботи алгоритму

**Висновки**

Розроблено проєкт у вигляді програмного комплексу для візуалізації роботи алгоритмів на графах, при цьому є можливість подавати граф різними способами. Проєкт має модульний інтерфейс, що дасть змогу легко доповнювати його в майбутньому новими алгоритмами. Створено JS-сценарії, які забезпечують анімацію роботи алгоритмів та їх реалізацію. Розроблений програмний комплекс можна використовувати для вивчення теорії графів та наочного демонстрування алгоритмів на графах, які є важливими для розв'язування багатьох практичних задач.

**Література**

1. Кузьменко І. М. Теорія графів : навчальний посібник для здобувачів ступеня бакалавра за спеціальністю 122 «Комп'ютерні науки» [Електронний ресурс] / Ігор Миколайович Кузьменко. – КПІ ім. Ігоря Сікорського, 2020. – Режим доступу : <https://ela.kpi.ua/handle/123456789/35854>.
2. Шевченко Г. В. Дискретна математика : навчально-методичний посібник. / Г. В. Шевченко, В. В. Шкапа. – Київ : ДУТ, 2018. – 158 с.
3. Остапок З. В. Застосування графів для відображення життєвого циклу сутностей під час розроблення системи опрацювання відгуків безпосередніх користувачів програмних продуктів / З. В. Остапок, Т. О. Коротєєва // Науковий вісник НЛТУ України. – 2019. – т. 29, № 9. – С. 147–152.
4. Ришковець Ю. В. Алгоритмізація та програмування. Частина 2: навчальний посібник / Ю. В. Ришковець, В. А. Висоцька. – Львів : Новий Світ-2000, 2020. – 320 с.
5. Шолом П. С. Пропорційно-інтегрально-диференціальне управління стабільністю руху мобільного агента Зрі по заданій траєкторії / П. С. Шолом, Н. В. Здолбіцька // Комп'ютерно-інтегровані технології: освіта, наука, виробництво : міжвузівський збірник. – Луцьк : Видавництво ЛНТУ, 2012. – № 8. – С. 125–129.

6. Шолом П. С. Аналіз алгоритмів обходу графа для задачі трасування маршруту / П. С. Шолом, Н. В. Здолбіцька // Комп'ютерно-інтегровані технології: освіта, наука, виробництво : міжвузівський збірник. – Луцьк : Видавництво ЛНТУ, 2011. – № 3. – С. 204–207.
7. Лавренчук С. В. Дослідження технології обробки природної мови та машинного навчання при створенні chat-bot засобами Python / С. В. Лавренчук, Р. С. Люшик. // Комп'ютерно-інтегровані технології: освіта, наука, виробництво : міжвузівський збірник. – Луцьк : Видавництво ЛНТУ. – 2019. – № 37. – С. 36–42.
8. Graph Theory Towards New Graphical Passwords In Information Networks [Електронний ресурс] / [Y. Bing, S. Hui, W. Hongyu та ін.]. – 2018. – Режим доступу : <https://arxiv.org/abs/1806.02929>.
9. Борисенко І. І. Застосування теорії графів в задачах створення стеганографічних повідомлень / І. І. Борисенко // Сучасна спеціальна техніка. – 2015. – № 2. – С. 26–33.
10. Жигаревич О. К. Методи та засоби проектування та розробки системи оптимізації транспортних маршрутів / О. К. Жигаревич // Комп'ютерно-інтегровані технології: освіта, наука, виробництво : міжвузівський збірник. – Луцьк : Видавництво ЛНТУ, 2013. – № 11. – С. 16–23.
11. Optimal Transport for structured data with application on graphs [Електронний ресурс] / [T. Vayer, L. Chapel, R. Flamary та ін.]. – 2019. – Режим доступу : <https://arxiv.org/abs/1805.09114>.
12. Бізюк А. В. Застосування теорії графів до аналізу структурно-логічної схеми навчального плану / А. В. Бізюк, В. П. Ткаченко, В. В. Бізюк // Тези доповідей IV міжнародної науково-технічної конференції «Поліграфічні, мультимедійні та web-технології» (PMW-2019), 14–17 травня 2019 р., Харків. – Харків : Друкарня Мадрид, 2019. – С. 238–239.
13. Бобрицька Г. С. Прикладне застосування теорії графів у різних сферах життя суспільства та окремої особистості / Г. С. Бобрицька // Фізико-математична освіта: науковий журнал. – 2017. – № 3. – С. 26–30.
14. Новожилова Л. М. Моделирование графами дискретных систем и процесов : учебное пособие / Л. М. Новожилова. – СПб : СПбГУ, 2017. – 16 с.
15. Graph Builder [Електронний ресурс]. – Режим доступу : <https://www.softportal.com/software-19403-graph-builder.html>
16. Обзор MaxFlow [Електронний ресурс]. – Режим доступу : <https://soft.mydiv.net/win/download-MaxFlow.html>
17. Grafoanalizator [Електронний ресурс]. – Режим доступу : <https://sourceforge.net/projects/grafanalizator/>
18. Graph Magics [Електронний ресурс]. – Режим доступу : <http://www.graph-magics.com/>

#### References

1. Kuzmenko I. M. Teoriia hrafiv : navchalnyi posibnyk dlia zdobuvachiv stupenia bakalavra za spetsialnistiu 122 «Kompiuterni nauky» [Elektronnyi resurs] / Ihor Mykolaiovych Kuzmenko. – KPI im. Ihoria Sikorskoho, 2020. – Rezhym dostupu : <https://ela.kpi.ua/handle/123456789/35854>.
2. Shevchenko H. V. Dyskretna matematika : navchalno-metodychni posibnyk. / H. V. Shevchenko, V. V. Shkapa. – Kyiv : DUT, 2018. – 158 s.
3. Ostapiuk Z. V. Zastosuvannia hrafiv dlia vidobrazhennia zhyttievoho tsykladu sutnostei pid chas rozroblennia systemy opratsiuvannia vidhukiv bezposerednikh korystuvachiv prohramnykh produktiv / Z. V. Ostapiuk, T. O. Korotieieva // Naukovyi visnyk NLTU Ukrainy. – 2019. – t. 29, № 9. – S. 147–152.
4. Ryshkovets Yu. V. Alhorytmizatsiia ta prohramuvannia. Chastyna 2: navchalnyi posibnyk / Yu. V. Ryshkovets, V. A. Vysotska. – Lviv : Novyi Svit-2000, 2020. – 320 s.
5. Sholom P. S. Proporsiiino-integralno-dyferentsialne upravlinnia stabilnistiu rukhu mobilnogo ahenta 3pi po zadanii traiektorii / P. S. Sholom, N. V. Zdolbitska // Kompiuterno-intehrovani tekhnolohii: osvita, nauka, vyrobnytstvo : mizhvuzivskiyi zbiryk. – Lutsk : Vydavnytstvo LNTU, 2012. – № 8. – S. 125–129.
6. Sholom P. S. Analiz alhorytmiv obkhodu hrafu dlia zadachi trasuvannia marshrutu / P. S. Sholom, N. V. Zdolbitska // Kompiuterno-intehrovani tekhnolohii: osvita, nauka, vyrobnytstvo : mizhvuzivskiyi zbiryk. – Lutsk : Vydavnytstvo LNTU, 2011. – № 3. – S. 204–207.
7. Lavrenchuk S. V. Doslidzhennia tekhnolohii obrobky pryrodnoi movy ta mashynnoho navchannia pry stvorenni chat-bot zasobamy Python / S. V. Lavrenchuk, R. S. Iliushyk. // Kompiuterno-intehrovani tekhnolohii: osvita, nauka, vyrobnytstvo : mizhvuzivskiyi zbiryk. – Lutsk : Vydavnytstvo LNTU. – 2019. – № 37. – S. 36–42.
8. Graph Theory Towards New Graphical Passwords In Information Networks [Elektronnyi resurs] / [Y. Bing, S. Hui, W. Hongyu та ін.]. – 2018. – Rezhym dostupu : <https://arxiv.org/abs/1806.02929>.
9. Borysenko I. I. Zastosuvannia teorii hrafiv v zadachakh stvorennia stehanohrafichnykh povidomlen / I. I. Borysenko // Suchasna spetsialna tekhnika. – 2015. – № 2. – S. 26–33.
10. Zhyharevych O. K. Metody ta zasoby proektuvannia ta rozrobky systemy optymizatsii transportnykh marshrutiv / O. K. Zhyharevych // Kompiuterno-intehrovani tekhnolohii: osvita, nauka, vyrobnytstvo : mizhvuzivskiyi zbiryk. – Lutsk : Vydavnytstvo LNTU, 2013. – № 11. – S. 16–23.
11. Optimal Transport for structured data with application on graphs [Elektronnyi resurs] / [T. Vayer, L. Chapel, R. Flamary та ін.]. – 2019. – Rezhym dostupu : <https://arxiv.org/abs/1805.09114>.
12. Biziuk A. V. Zastosuvannia teorii hrafiv do analizu strukturo-lohichnoi skhemy navchalnogo planu / A. V. Biziuk, V. P. Tkachenko, V. V. Biziuk // Tezy dopovidei IV mizhnarodnoi naukovo-tekhnichnoi konferentsii «Polihrafichni, multymediini ta web-tekhнології» (PMW-2019), 14–17 travnia 2019 r., Kharkiv. – Kharkiv : Drukarnia Madryd, 2019. – S. 238–239.
13. Bobrytska H. S. Prykladne zastosuvannia teorii hrafiv u riznykh sferakh zhyttia suspilstva ta okremoi osobystosti / H. S. Bobrytska // Fyzyko-matematychna osvita: naukovyi zhurnal. – 2017. – № 3. – S. 26–30.
14. Novozhylova L. M. Modelyrovanye hrafamy diskretnykh system y protsesov : uchebnoe posobyie / L. M. Novozhylova. – Spb : SPbHU, 2017. – 16 s.
15. Graph Builder [Elektronnyi resurs]. – Rezhym dostupu : <https://www.softportal.com/software-19403-graph-builder.html>
16. Obzor MaxFlow [Elektronnyi resurs]. – Rezhym dostupu : <https://soft.mydiv.net/win/download-MaxFlow.html>
17. Grafoanalizator [Elektronnyi resurs]. – Rezhym dostupu : <https://sourceforge.net/projects/grafanalizator/>
18. Graph Magics [Elektronnyi resurs]. – Rezhym dostupu : <http://www.graph-magics.com/>