



МАТЕРІАЛИ VII ВСЕУКРАЇНСЬКОЇ НАУКОВО-ПРАКТИЧНОЇ КОНФЕРЕНЦІЇ

ІНФОРМАЦІЙНЕ СУСПІЛЬСТВО: «ПРОБЛЕМИ ТА ПЕРСПЕКТИВИ»



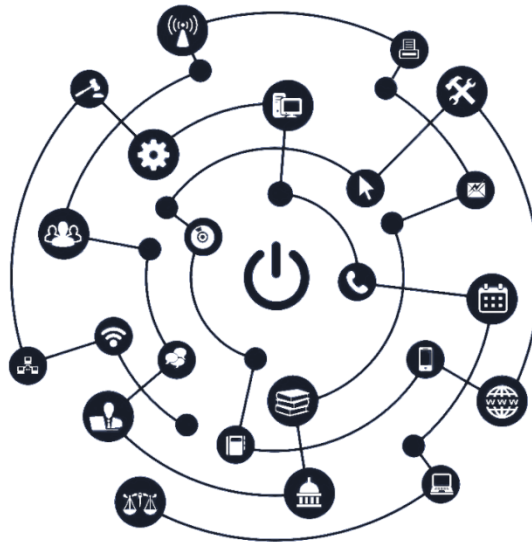
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
«ОДЕСЬКА ЮРИДИЧНА АКАДЕМІЯ»

ФАКУЛЬТЕТ КІБЕРБЕЗПЕКИ
ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

КАФЕДРА ІНФОРМАЦІЙНИХ
ТЕХНОЛОГІЙ

ОДЕСА
20 травня 2022 р.

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
Національний університет «Одеська юридична академія»
Факультет кібербезпеки та інформаційних технологій
Кафедра інформаційних технологій



ІНФОРМАЦІЙНЕ СУСПІЛЬСТВО: ПРОБЛЕМИ ТА ПЕРСПЕКТИВИ

МАТЕРІАЛИ VII ВСЕУКРАЇНСЬКОЇ НАУКОВО-ПРАКТИЧНОЇ КОНФЕРЕНЦІЇ

20 травня 2022 року, м. Одеса

Одеса, 2022

УДК 340.12:316.3

ББК 67.0

Відповідальний редактор:

Н. І. Логінова, завідувач кафедри інформаційних технологій, к.пед.н., доцент

Матеріали видано в авторській редакції.

Повну відповідальність за достовірність та якість поданого матеріалу несуть учасники конференції, їхні наукові керівники, рецензенти, які рекомендували ці матеріали до друку.

*Друкується за рішенням кафедри інформаційних технологій
Національного університету «Одеська юридична академія»
(протокол засідання кафедри № 11 від 09.05.2022 р.)*

Матеріали VII Всеукраїнської науково-практичної конференції «Інформаційне суспільство: проблеми та перспективи» (м. Одеса, 20 травня 2022 р.). Одеса, 2022. 151 с.
DOI: <https://doi.org/10.32837/11300.17736>

Всеукраїнська науково-практична конференція «Інформаційне суспільство: проблеми та перспективи» присвячена розгляду актуальних питань розвитку інформаційного суспільства в Україні.

До участі у конференції запрошені студенти, аспіранти, молоді вчені, викладачі та науковці. Конференція проводиться на базі Національного університету «Одеська юридична академія».

ВІЗУАЛІЗАЦІЯ АЛГОРИТМУ ДЕЙКСТРИ

Трофименко О. Г.

*кандидат технічних наук, доцент, доцент кафедри інформаційних технологій
Національного університету «Одеська юридична академія»*

Новіков Д. І.

*студент 2-го курсу факультету кібербезпеки та інформаційних технологій
Національного університету «Одеська юридична академія»*

Сьогодні для ІТ-освіти, і як наслідок для ІТ-галузі, характерні проблеми якості освіти і дефіцит спеціалістів водночас. Здобувачі ІТ-освіти мають багато вчитися і під час навчання, і під час професійної діяльності, щоб встигати за змінами

в ІТ-індустрії загалом та новими трендами у вузькоспеціалізованих напрямках [1].

Нині на співбесідах ІТ-компаній та оцінці кандидатів найму часто питання з алгоритмів рішення задач (Problem Solving) є ключовими. «Алгоритмічні» питання можуть становити від 60% до 100% завдань у співбесіді [2]. Здебільшого такі завдання полягають в неочевидному вирішенні, аніж у реалізації, і дозволяють з'ясувати здатність фахівця до розробки нового функціоналу програмного продукту та оптимізації наявного, що дозволить і надалі просувати компанії вгору.

Тому знання і вміння застосовувати ефективні алгоритми є важливою компетенцією для розробників програмного забезпечення (ПЗ). Алгоритми використовуються практично в усіх сферах інформаційних технологій: в криптографії, при аналізі текстів, зображень і відео, в біоінформатиці, аналізі великих даних, штучному інтелекті, задачах оптимізації тощо.

Переважно основні етапи проектування, розробки та аналізу алгоритмів вивчаються в рамках дисципліни «Алгоритми та структури даних» (АСД) у закладах вищої освіти України при підготовці студентів галузі 12 «Інформаційні технології». В університетах світу цей курс здебільшого має назву CS2. Набуті у цьому курсі навички сприятимуть написанню ефективного робочого коду при розробці проектів. Курс АСД (або CS2) є доволі складним для розуміння студентами, багато хто з них сприймає матеріал лише частково [3]. За даними [4], невдачі наприкінці цього курсу складають 5,7%. Серед найскладніших тем курсу дослідники називають алгоритми на графах і деревах, рекурсію, купу, алгоритми сортування та пошуку, аналіз швидкості алгоритмів, зв'язані списки, черги, карти і хеш-таблиці [5-7]. Причому перелік найскладніших тем CS2 суттєво відрізняється у різних дослідників, що може свідчити про наявність суб'єктивних чинників, зокрема: розподіл часу за темами, добір завдань для лабораторних робіт, попередній досвід вивчення програмування у студентів, педагогічна майстерність викладача тощо.

Ефективним підходом до вирішення проблем з опанування студентами складних тем програмного опрацювання графів є допоміжна візуалізація алгоритмів. Адже за даним досліджень [8] мозок людини обробляє зображення в 60 000 разів швидше, ніж текст, і 90% відсотків інформації, що передається мозку, є саме візуальною.

Для програмної реалізації візуалізації алгоритму Дейкстри було використано мову програмування C#, вибір якої зумовлений наявністю в .Net Framework потужних інструментів Windows Forms та Windows Presentation Foundation (WPF) для розробки графічного інтерфейсу користувача (GUI). WPF навіть дозволяє розробляти інтерфейс, побудований на векторній графіці, що дозволяє масштабувати елементи інтерфейсу без втрати якості та без спотворення розмірів.

Розроблений проєкт LOGMAP спрямований на візуальну інтерактивну побудову графів з метою наочної демонстрації ефективної роботи алгоритму Дейкстри для пошуку найкоротшого шляху між двома заданими користувачем

вершинами графа. Користувач може довільно задавати вершини графа на полі, просто клацаючи по ньому у відповідній позиції. Є можливість заповнення поля випадково сформованими вершинами. Поєднувати вершини графа можна у довільно вибраному користувачем порядку. Для економії часу є режим випадкового поєднання вершин графів, що особливо зручно в процесі навчального процесу. Кількість вершин та дуг графів може бути довільною. Можна для кожної вершини задавати назву, наприклад, якогось географічного об'єкта на мапі або назви торгової точки при практичній реалізації графа. Для дуг можна задавати вагу, наприклад, як значення відстані між двома вершинами. Значення ваги для графів може бути комплексною характеристикою, яка враховує декілька чинників, наприклад: відстань, якість дороги, наявність заторів на дорогах тощо. Якщо користувач не задає назви вершин, то вони формуються з усталеного набору, а значення ваги дуг формуються, залежно від відстані між вершинами. Є можливість вимикати відображення значень ваги та назв вершин, якщо не варто захирачувати зайвою деталізацією візуалізацію графа. Закладено можливість задавати односпрямовані дуги для орієнтованих графів, наприклад, якщо це стосується практичного застосування для доріг з односпрямованим рухом.

Окрім зазначених переваг, у проєкті LOGMAP закладено можливість вільної інтерактивної трансформації графа, з можливістю його вільного редагування за допомогою покажчика миші, і навіть масштабування, наприклад для того, щоб детально переглянути та відредагувати якийсь певний фрагмент масштабного графа.

Програмно алгоритм Дейкстри реалізовано в окремому класі з набором відповідних методів та властивостей. Його основний метод GetPath визначає найкоротший шлях поміж точками у формі списку з'єднань вершин графа (рис. 1).

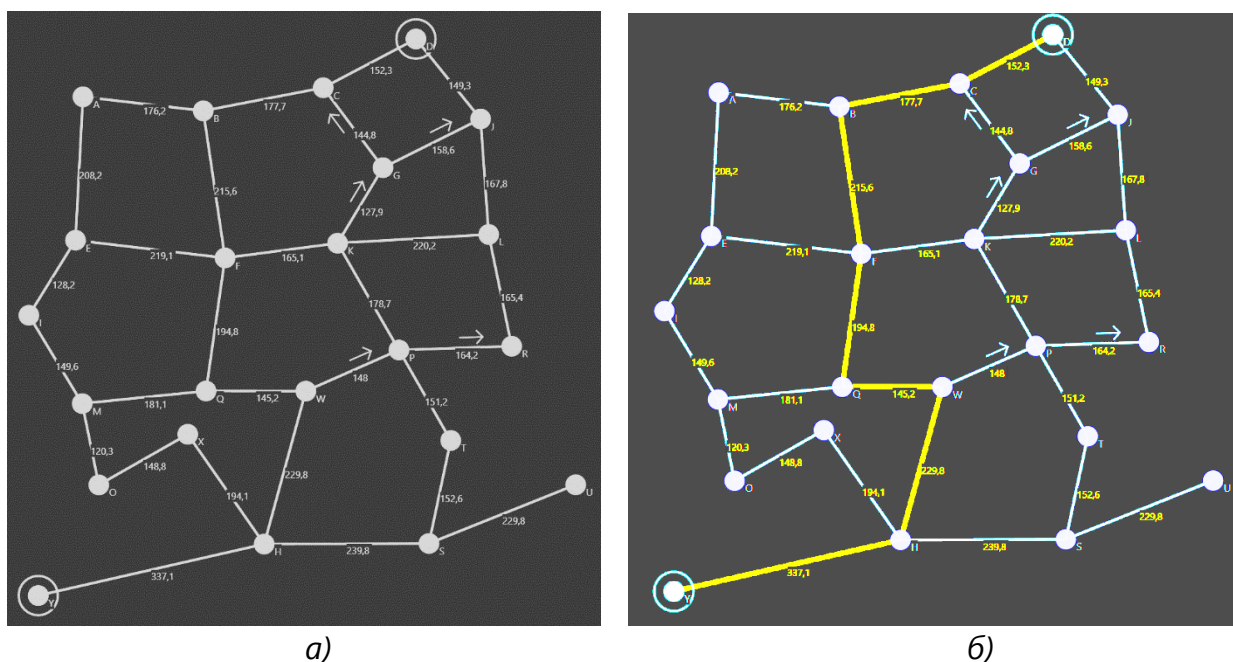


Рис. 1. Візуалізація алгоритму Дейкстри:

- а) задано граф і дві вершини (виділені кільцем) для пошуку найкоротшого шляху;
- б) визначено найкоротший шлях між двома заданими вершинами (жовта жирна лінія)

Аргументами цього методу є стартова та кінцева вершини для пошуку найкоротшого шляху поміж ними, а також список усіх вершин графа, які будуть аналізуватися для пошуку шляху від стартової вершини до кінцевої.

У методі GetPath формується словник словників, в якому зберігаються дані з'єднань між вершинами та вага кожного з них. Цей словник можна представити матрицею суміжності (рис. 2). Якщо з'єднання існує, то воно додається до словника.

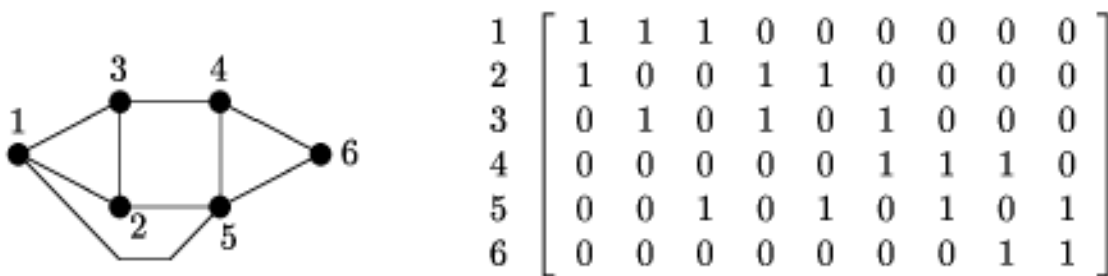


Рис. 2. Матриця суміжності графа

При реалізації пошуку ітераційно у циклі за допомогою метода GetSmallest() визначається найближча вершина, оновлюються дистанції до інших вершин методом UpdateDistances(), список невідвіданих вершин сортується, залежно від дистанцій, які зберігаються в словнику, а метод BuildPath() будує шлях у вигляді списку номерів визначених вершин.

Запропонована реалізація алгоритму доволі швидка. Наприклад, пошук найкоротшого шляху на графі на рис. 1 зайняв менше 1 мс. Звісно, що на швидкість пошуку впливають характеристики процесора, кількості вершин та дуг графа. Експериментальні тестові пошуки на графі з 422 вершин і 562 дуг між різними його вершинами показали час в діапазоні від 17 до 24 мс.

Практичне застосування розробленого проєкту LOGMAP під час пояснення роботи алгоритмів на графах у курсі АСД показав гарні показники засвоєння навчального матеріалу, про що свідчать позитивні оцінки успішності шляхом тестування з теми.

Візуалізація складних алгоритмів є сучасним та ефективним підходом для кращого засвоєння їх здобувачами вищої освіти. Важливість курсу АСД в процесі підготовки ІТ-фахівців змушує викладачів в університетах нашої країни та всього світу приділяти значну увагу вдосконаленню його викладання та шукати нові підходи викладання. Адже розуміння принципів роботи алгоритмів і структур даних під час розроблення програмного забезпечення дозволяє покращити продуктивність програм, поліпшити якість коду і прискорити його роботу, а тому є фундаментальним для становлення майбутніх програмістів.

Список використаних джерел:

1. Прокоп Ю.В., Трофименко О.Г., Задерейко О.В. Аналіз підходів у викладанні початкового курсу програмування в університетах. *Системні технології*. 2021.

- № 4(135). С. 73-84. DOI 10.34185/1562-9945-4-135-2021-08.
2. Прокоп Ю.В., Трофименко О.Г., Дикий О.В. Дослідження підходів до викладання курсу "Алгоритми та структури даних" для студентів ІТ-спеціальностей. *Вчені записки Таврійського національного університету імені В.І. Вернадського*. Серія «Технічні науки». 2021. Том 32 (71) Ч. 1 № 2. С. 216-220. DOI: 10.32838/2663-5941/2021.2-1/34.
 3. Yeomans L., Zschaler S., Coate K. Transformative and Troublesome? Students' and Professional Programmers' Perspectives on Difficult Concepts in Programming. *ACM Transactions on Computing Education*. Vol. 1, No. 1. P. 1-27. DOI: 10.1145/3283071.
 4. Layman L., Song Y., Guinn C. Toward Predicting Success and Failure in CS2: A Mixed-Method Analysis. *ACM Southeast Conference (ACM SE '20)*. New York, NY, USA, 2020. P. 218-225. DOI: 10.1145/3374135.3385277.
 5. Wittie L., Kurdia A., Peng J., Kelly J., Huggard M. Developing a Concept Inventory for Computer Science 2: What should it focus on and what makes it challenging? *IEEE Frontiers in Education Conference (FIE'20)*. 2020. P. 1-8. URL: http://www.tara.tcd.ie/bitstream/handle/2262/92988/CS2_Concept_Inventory_FIE_2020.pdf;jsessionid=BD4D1BE6611256FB0C6A3CF76556F59C?sequence=1.
 6. Zingaro D., Taylor C., Porter L., Clancy M., Lee C., Nam L., Webb K. Identifying student difficulties with basic data structures. In *Proceedings of the 14th ACM Conference on International Computing Education Research*. 2018. P.169-177. URL: <https://www.cs.oberlin.edu/~ctaylor/pubs/icer143-zingaroA.pdf>
 7. Tenenberg J., Murphy L. Knowing what I know: An investigation of undergraduate knowledge and self-knowledge of data structures. *Computer Science Education*. 2005. Vol. 15, No. 4. P. 297-315. DOI: 10.1080/08993400500307677
 8. Humans Process Visual Data Better. URL: <https://www.t-sciences.com/news/humans-process-visual-data-better>

ІНФОРМАЦІЙНЕ СУСПІЛЬСТВО: ПРОБЛЕМИ ТА ПЕРСПЕКТИВИ

**МАТЕРІАЛИ VII ВСЕУКРАЇНСЬКОЇ
НАУКОВО-ПРАКТИЧНОЇ КОНФЕРЕНЦІЇ**

*20 травня 2022 року
м. Одеса*

Відповідальній редактор: Н. І. Логінова

Дизайнер: Р. Р. Дробожур



**НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
«ОДЕСЬКА ЮРИДИЧНА АКАДЕМІЯ»**

**ФАКУЛЬТЕТ КІБЕРБЕЗПЕКИ
ТА ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ**

**КАФЕДРА ІНФОРМАЦІЙНИХ
ТЕХНОЛОГІЙ**

**МАТЕРІАЛИ VII ВСЕУКРАЇНСЬКОЇ
НАУКОВО-ПРАКТИЧНОЇ КОНФЕРЕНЦІЇ
ІНФОРМАЦІЙНЕ СУСПІЛЬСТВО:
«ПРОБЛЕМИ ТА ПЕРСПЕКТИВИ»**

20 травня 2022 року, м. Одеса