

Секція Управління	
УДК 004.451	
Дата першого надходження статті до видання	
Дата прийняття статті до друку після рецензування	
Дата публікації/оприлюднення	2026-03-30

Методичні засади навчання формальних моделей операційних систем у контексті теоретичної інформатики

Стоцький Іван Володимирович

Старший викладач кафедри комп'ютерних наук та інтелектуальних технологій,
факультет Інформаційних технологій,
Військовий інститут телекомунікацій та інформатизації імені Героїв Крут, Київ, Україна
e-mail: ivan.stotskyi@viti.edu.ua
<https://orcid.org/0000-0002-5898-3228>

Голуб Олександр Олександрович

Старший викладач кафедри комп'ютерних наук та інтелектуальних технологій,
факультет Інформаційних технологій,
Військовий інститут телекомунікацій та інформатизації імені Героїв Крут, Київ, Україна
e-mail: oleksandr.golub@viti.edu.ua
<https://orcid.org/0009-0006-1122-7667>

Налісний Геннадій Іванович

Викладач кафедри комп'ютерних наук та інтелектуальних технологій,
факультет Інформаційних технологій,
Військовий інститут телекомунікацій та інформатизації імені Героїв Крут, Київ, Україна
e-mail: hennadii.nalisnyi@viti.edu.ua
<https://orcid.org/0009-0000-5142-3677>

Костенко Віталій Петрович

Викладач кафедри комп'ютерних наук та інтелектуальних технологій,
факультет Інформаційних технологій,
Військовий інститут телекомунікацій та інформатизації імені Героїв Крут, Київ, Україна
e-mail: vitalij.kostenko@viti.edu.ua
<https://orcid.org/0000-0002-7180-233X>

Анотація. У статті досліджено методичні засади навчання формальних моделей операційних систем у контексті теоретичної інформатики та професійної підготовки майбутніх ІТ-фахівців. Актуальність дослідження зумовлена необхідністю підвищення рівня формалізації викладання дисципліни «Операційні системи» та інтеграції formal methods у сучасну ІТ-освіту. Проаналізовано сучасні підходи до використання автоматних моделей, графів ресурсів, сіток Петрі та мов формальної специфікації у процесі викладання операційних систем. Встановлено, що використання формалізованих моделей забезпечує поглиблене розуміння механізмів функціонування ОС, сприяє розвитку системного мислення, навичок логічного аналізу, верифікації системних станів та формалізованого опису процесів синхронізації і розподілу ресурсів.

У результаті проведеного порівняльного аналізу визначено дидактичні можливості та обмеження різних підходів до формалізації процесів функціонування

операційних систем. З'ясовано, що модель Холта забезпечує формування уявлень про взаємодію процесів і ресурсів та механізми виникнення тупикових ситуацій, state-machine approach сприяє формалізованому опису системних станів і перевірці властивостей системної поведінки, а використання сіток Петрі забезпечує наочне представлення процесів конкурентності та взаємного виключення. Обґрунтовано доцільність поєднання формальних моделей із практично орієнтованими формами навчання, зокрема лабораторним моделюванням системних процесів і використанням засобів model checking.

Доведено, що інтеграція formal methods у структуру дисципліни «Операційні системи» створює передумови для переходу від описового засвоєння системних механізмів до їх логічно та математично структурованого аналізу. Запропоновані методичні підходи можуть бути використані під час розроблення навчальних програм, лабораторних практикумів та цифрових засобів моделювання у закладах вищої освіти.

Ключові слова: операційні системи, formal methods, формальні моделі, скінченні автомати, сітки Петрі, model checking, теоретична інформатика, професійна підготовка ІТ-фахівців.

Methodological principles of teaching formal models of operating systems in the context of theoretical computer science

Stotskyi Ivan

Senior Lecturer, Department of Computer Science and Intellectual Technologies,
Faculty of Information Technologies,
Kruty Heroes Military Institute of Telecommunications and Information Technology, Kyiv, Ukraine
e-mail: ivan.stotskyi@viti.edu.ua
<https://orcid.org/0000-0002-5898-3228>

Holub Oleksandr

Senior Lecturer, Department of Computer Science and Intellectual Technologies,
Faculty of Information Technologies,
Kruty Heroes Military Institute of Telecommunications and Information Technology, Kyiv, Ukraine
e-mail: oleksandr.golub@viti.edu.ua
<https://orcid.org/0009-0006-1122-7667>

Nalisnyi Hennadii

Lecturer, Department of Computer Science and Intellectual Technologies,
Faculty of Information Technologies,
Kruty Heroes Military Institute of Telecommunications and Information Technology, Kyiv, Ukraine
e-mail: hennadii.nalisnyi@viti.edu.ua
<https://orcid.org/0009-0000-5142-3677>

Kostenko Vitalii

Teacher, Department of Computer Science and Intellectual Technologies,
Faculty of Information Technologies,
Kruty Heroes Military Institute of Telecommunications and Information Technology, Kyiv, Ukraine
e-mail: vitalij.kostenko@viti.edu.ua
<https://orcid.org/0000-0002-7180-233X>

Abstract. The article examines the methodological principles of teaching formal models of operating systems in the context of theoretical computer science and professional training of future IT specialists. The relevance of the study is due to the need to increase the level of formalization of teaching the discipline "Operating Systems" and the integration of formal

methods into modern IT education. Modern approaches to the use of automata models, resource graphs, Petri nets and formal specification languages in the process of teaching operating systems are analyzed. It was established that the use of formalized models provides an in-depth understanding of the mechanisms of OS functioning, contributes to the development of systems thinking, logical analysis skills, verification of system states and formalized description of synchronization and resource allocation processes. As a result of the comparative analysis, the didactic possibilities and limitations of various approaches to the formalization of operating systems functioning processes are determined. It was found that the Holt model provides the formation of ideas about the interaction of processes and resources and the mechanisms of the emergence of deadlocks, the state-machine approach contributes to the formalized description of system states and verification of the properties of system behavior, and the use of Petri nets provides a visual representation of the processes of competition and mutual exclusion. The feasibility of combining formal models with practically oriented forms of learning is substantiated, in particular, laboratory modeling of system processes and the use of model checking tools.

It is proven that the integration of formal methods into the structure of the discipline "Operating Systems" creates the prerequisites for the transition from descriptive mastering of system mechanisms to their logically and mathematically structured analysis. The proposed methodological approaches can be used when developing curricula, laboratory workshops and digital modeling tools in higher education institutions.

Keywords: (operating systems, formal methods, formal models, finite state machines, Petri nets, model checking, theoretical computer science, professional training of IT specialists.

Вступ

Актуальність проблеми. Стрімке ускладнення сучасних програмних систем, поширення багатопотокових і розподілених обчислень, розвиток хмарних сервісів та критично важливого програмного забезпечення зумовлюють необхідність підвищення рівня формалізації процесів проектування, аналізу та верифікації комп'ютерних систем. За таких умов особливого значення набувають формальні методи, які забезпечують математично коректний опис поведінки систем, перевірку властивостей алгоритмів, аналіз взаємодії процесів і виявлення потенційних помилок ще на етапі моделювання. У сучасних дослідженнях підкреслюється, що formal methods повинні розглядатися як фундаментальна складова підготовки фахівців у галузі computer science, а не як допоміжний або факультативний компонент освітніх програм [11].

Актуальність використання формальних моделей у підготовці майбутніх ІТ-фахівців посилюється тим, що сучасні операційні системи функціонують у середовищах із високим рівнем паралелізму, конкуренції процесів та складною організацією управління ресурсами. Традиційне описове викладання механізмів планування процесів, синхронізації, взаємного виключення або управління пам'яттю часто не забезпечує достатнього рівня розуміння внутрішньої динаміки функціонування ОС. У зв'язку з цим використання автоматних моделей, графових структур, сіток Петрі та мов формальної специфікації створює можливість переходу від декларативного засвоєння матеріалу до формалізованого аналізу поведінки системи та моделювання її станів. Практична значущість відповідного підходу підтверджується сучасними дослідженнями у сфері computer science education та досвідом застосування formal methods у промисловій розробці програмного забезпечення [4].

Водночас аналіз сучасних освітніх стандартів і навчальних програм свідчить про недостатній рівень інтеграції formal methods у структуру професійної підготовки майбутніх фахівців з комп'ютерних наук [5; 6], що актуалізує проблему пошуку методичних підходів до використання формальних моделей у викладанні дисциплін системного спрямування, зокрема курсу операційних систем, де моделювання процесів,

станів та взаємодії ресурсів виступає необхідною умовою формування системного мислення майбутніх ІТ-фахівців.

Аналіз останніх досліджень і публікацій. У сучасних дослідженнях з теорії та методики навчання комп'ютерних наук простежується тенденція до посилення ролі формальних методів у процесі професійної підготовки майбутніх ІТ-фахівців. Актуалізація відповідного напрямку пов'язана зі зростанням складності програмних і системних архітектур, що потребує використання формалізованих засобів опису, аналізу та верифікації процесів функціонування комп'ютерних систем. У контексті викладання операційних систем це зумовлює необхідність застосування моделей, здатних забезпечувати формалізоване представлення процесів синхронізації, планування, розподілу ресурсів та міжпроцесної взаємодії.

У дослідженні Y. Jiang, X. Wang та H. Zhang запропоновано підхід «state-machine first», у межах якого вивчення операційних систем здійснюється через використання скінченних автоматів і моделювання станів системи [8], який орієнтований на формування системного розуміння поведінки операційної системи шляхом аналізу переходів між станами, моделювання виконання процесів та верифікації логіки функціонування системних механізмів.

У дослідженнях, присвячених методиці викладання операційних систем, окрему увагу приділено використанню формальних графових моделей для пояснення механізмів конкурентності та взаємного виключення. Зокрема, U. Arronategui обґрунтував необхідність застосування сіток Петрі як засобу моделювання процесів синхронізації, взаємодії ресурсів та аналізу досяжності станів у курсі операційних систем [3]. Використання відповідного формалізму дозволяє забезпечити візуалізацію паралельних процесів і формалізований аналіз системних станів.

У вітчизняних науково-методичних працях також акцентується необхідність фундаменталізації підготовки майбутніх фахівців у галузі комп'ютерних наук та формалізації базових понять операційних систем. Зокрема, С. О. Семеріков розглядає операційну систему як механізм управління та розподілу ресурсів комп'ютерної системи й наголошує на необхідності використання формалізованих підходів у процесі пояснення принципів її функціонування [2].

Виділення невирішеної частини проблеми. Попри наявність окремих наукових і методичних напрацювань, проблема інтеграції формальних моделей у процес викладання операційних систем залишається недостатньо розробленою. У сучасних дослідженнях відсутній цілісний методичний підхід до використання formal methods у навчанні ОС, недостатньо конкретизовано педагогічні цілі їх застосування та не здійснено системного порівняльного аналізу існуючих моделей і методик формалізації процесів функціонування операційних систем у професійній підготовці майбутніх ІТ-фахівців.

Мета статті. Метою статті є аналіз методичних засад навчання формальних моделей операційних систем у контексті теоретичної інформатики

Наукова новизна полягає у систематизації та обґрунтуванні ключових компонентів методики: впровадження скінченних автоматів і графових моделей (петри-мереж, ресурсних графів) до курсу ОС як засобу фундаменталізації знань.

Практичне значення. Розроблені рекомендації можуть бути застосовані при розробці навчальних програм і курсів «Операційні системи» та «Теоретична інформатика» у закладах вищої освіти. Формалізація матеріалу сприятиме глибшому засвоєнню понять ОС, розумінню засад конкурентності і безпеки, а також покращенню навичок студентів у побудові моделей комп'ютерних систем.

Методологія

Методи дослідження. Методологічну основу дослідження становлять теоретичні методи аналізу, порівняння, систематизації та узагальнення науково-методичних підходів до навчання формальних моделей операційних систем. У процесі дослідження застосовано контент-аналіз наукових публікацій з проблематики formal methods і викладання дисципліни «Операційні системи», а також метод педагогічного моделювання для побудови структурних схем вивчення окремих тем курсу. Порівняльний аналіз використано для систематизації сучасних підходів до формалізації процесів функціонування операційних систем та визначення їх дидактичних особливостей.

Джерела даних. Джерельну базу дослідження становлять сучасні українські та зарубіжні наукові публікації, монографії, методичні матеріали, освітні стандарти та матеріали наукових конференцій 2010–2026 рр. У процесі відбору джерел пріоритет надавався первинним науковим публікаціям та офіційним матеріалам, представленим у наукових базах ACM, USENIX, IEEE, а також вітчизняним працям з дидактики інформатики та методики викладання операційних систем. Для аналізу підходів до формалізації моделей ресурсів операційної системи використано навчальний посібник, підготовлений у КПІ ім. Ігоря Сікорського [1].

Інструменти аналізу. Для дослідження використано інструменти теоретичного аналізу та педагогічного моделювання. Контент-аналіз застосовано з метою виявлення концептуальних підходів до навчання formal methods у курсі операційних систем. Порівняльний аналіз забезпечив систематизацію сучасних методик використання автоматних моделей, сіток Петрі, графових структур і мов формальної специфікації у процесі викладання ОС. Узагальнення результатів здійснювалося шляхом побудови порівняльних характеристик методичних підходів та розроблення прикладів лабораторних завдань, орієнтованих на інтеграцію формальних моделей у навчальний процес.

Обмеження дослідження. Обмеження дослідження характеризується недостатньою представленістю у відкритих джерелах матеріалів щодо практичного використання спеціалізованих засобів моделювання та особливостей реалізації окремих операційних систем у навчальному процесі. Додатковим обмеженням є відсутність уніфікованих методичних стандартів щодо навчання формальних моделей операційних систем, що зумовлює фрагментарність існуючих науково-методичних підходів. Водночас дослідження ґрунтується на систематизації сучасних теоретичних концепцій, узагальненні педагогічних практик та порівнянні підходів до формалізації процесів функціонування операційних систем.

Результати

У результаті проведеного порівняльного аналізу виокремлено основні підходи до викладання формальних моделей операційних систем, які відрізняються рівнем формалізації, типом використовуваних моделей та дидактичними можливостями їх інтеграції у процес професійної підготовки майбутніх ІТ-фахівців.

Одним із базових підходів є використання моделі Холта, що ґрунтується на представленні операційної системи у вигляді графа розподілу ресурсів. Зазначена модель застосовується для формалізованого аналізу взаємозалежностей між процесами та ресурсами, а також дослідження механізмів синхронізації й виникнення тупикових ситуацій [1]. У методичному вимірі використання відповідної моделі передбачає побудову студентами графів запитів і виділення ресурсів для заданих системних сценаріїв із подальшим аналізом циклів та умов виникнення deadlock. Застосування моделі Холта сприяє формуванню здатності ідентифікувати конфліктні стани системи, аналізувати механізми розподілу ресурсів та формалізовано описувати взаємодію процесів у середовищі операційної системи.

Іншим перспективним напрямом є state-machine approach, у межах якого операційна система розглядається як сукупність взаємопов'язаних скінченних автоматів. Відповідний підхід реалізовано у моделі MOSAIC, запропонованій Y. Jiang [7]. Методологічною основою зазначеного підходу є інтерпретація системних викликів, поведінки процесів та механізмів функціонування ОС через зміну системних станів і переходів між ними. У межах навчального процесу це передбачає моделювання мінімальних ОС-систем, побудову графів станів, аналіз алгоритмів взаємного виключення та використання model checking для перевірки коректності системної поведінки [12].

Дидактична цінність state-machine approach полягає у формуванні здатності студентів до формалізованого опису системних станів, аналізу переходів між ними та верифікації базових властивостей функціонування операційної системи. Використання відповідного підходу забезпечує перехід від описового засвоєння механізмів ОС до їх математично та логічно формалізованого аналізу, що сприяє розвитку системного мислення та навичок формального моделювання у майбутніх ІТ-фахівців [7, с. 931].

Перевагами state-machine approach є високий рівень інтерактивності навчання, наочність представлення системних процесів та можливість автоматизованої перевірки моделей і властивостей системи. Водночас обмеженням відповідного підходу виступає спрощений характер моделей, які не відображають повний спектр низькорівневих механізмів функціонування реальних операційних систем, що потребує додаткового методичного пояснення під час переходу від навчальних моделей до аналізу реальних системних архітектур [9].

Окремий напрям формалізації навчання операційних систем пов'язаний із використанням сіток Петрі як засобу моделювання паралельних процесів і механізмів синхронізації. Застосування відповідного формалізму забезпечує можливість представлення конкурентних процесів, критичних секцій та механізмів взаємного виключення у графічній формі. У межах навчального процесу студентам можуть пропонуватися завдання, пов'язані з побудовою сіток Петрі для паралельних алгоритмів, аналізом досяжності станів та виявленням умов виникнення тупикових ситуацій. Як зазначає U. Arronategui, використання сіток Петрі сприяє формуванню наочного розуміння динаміки функціонування операційних систем і процесів взаємодії ресурсів [3]. Перевагами зазначеного підходу є візуалізація процесів синхронізації та можливість формального аналізу досяжності системних станів. Водночас використання сіток Петрі ускладнюється зі зростанням масштабів системи та потребує сформованих навичок побудови й інтерпретації відповідних моделей.

Іншим напрямом формалізації навчання операційних систем є також інтеграція мов формальної специфікації, зокрема Z, TLA+ та Promela, для опису алгоритмів управління процесами, синхронізації та файлових систем. Використання відповідного підходу передбачає розроблення студентами формальних специфікацій окремих компонентів ОС із подальшою перевіркою їх властивостей та консистентності. Практична реалізація може включати формалізований опис спрощених механізмів файлових систем або алгоритмів взаємного виключення з використанням засобів model checking. Дидактична значущість відповідного підходу полягає у формуванні навичок формального опису алгоритмів, розвитку логічного мислення та засвоєнні принципів строгої специфікації системних процесів. Крім того, застосування мов формальної специфікації характеризується високим рівнем складності для студентів початкового рівня підготовки та потребує використання спеціалізованих програмних інструментів і середовищ верифікації моделей.

З метою систематизації сучасних методичних підходів до навчання формальних моделей операційних систем необхідним є проведення їх порівняльного аналізу за критеріями концептуальної основи моделювання, дидактичних цілей, способів

організації навчальної діяльності, характеру практичних завдань, а також педагогічних переваг і обмежень. Узагальнення відповідних підходів дозволяє визначити специфіку використання графових моделей, автоматних формалізмів і сіток Петрі у процесі викладання дисципліни «Операційні системи», а також оцінити їх дидактичний потенціал у формуванні системного мислення та навичок формалізованого аналізу процесів функціонування ОС. Результати порівняльного аналізу представлено в таблиці 1.

Таблиця 1. Порівняльна характеристика підходів до викладання формальних моделей операційних систем

Модель / підхід	Навчальна мета	Методичні засоби	Переваги	Обмеження
Модель Холта (граф ресурсів)	Аналіз взаємодії процесів і ресурсів, вивчення тупиків	Побудова ресурсних графів, моделювання deadlock	Наочність і формальна строгість	Обмежене відображення динаміки ОС
«State-machine first»	Формалізація станів ОС, аналіз синхронізації та планування	Побудова графів станів, model checking, моделювання алгоритмів	Інтерактивність, автоматизована перевірка моделей	Високий рівень абстракції
Сітки Петрі	Вивчення конкурентності та взаємного виключення	Побудова й аналіз мереж Петрі	Візуалізація паралельних процесів	Ускладнення моделей при масштабуванні

Сформовано на основі [1; 3; 7]

Отже, проведений порівняльний аналіз засвідчив, що сучасні підходи до викладання формальних моделей операційних систем відрізняються рівнем формалізації, типом використовуваних моделей та дидактичними можливостями їх застосування у навчальному процесі. Використання графових моделей, автоматних формалізмів і сіток Петрі забезпечує формування у студентів системного розуміння механізмів функціонування ОС, навичок аналізу синхронізації процесів, розподілу ресурсів та формалізованого опису системних станів. Водночас кожен із підходів характеризується специфічними методичними перевагами й обмеженнями, що зумовлює доцільність їх комбінованого використання у процесі професійної підготовки майбутніх ІТ-фахівців.

Обговорення

Інтерпретація результатів. Результати проведеного аналізу свідчать, що інтеграція формальних моделей у структуру курсу «Операційні системи» підвищує рівень концептуального розуміння механізмів функціонування ОС та сприяє формуванню у студентів навичок формалізованого аналізу системних процесів. Використання скінчених автоматів, графів станів і засобів model checking забезпечує можливість представлення поведінки операційної системи у формалізованому вигляді, що дозволяє аналізувати переходи між станами, механізми синхронізації та властивості системної поведінки [7]. У дослідженнях, присвячених проблематиці взаємного виключення та формалізації алгоритмів синхронізації, також підкреслюється, що застосування формальних моделей сприяє глибшому розумінню системної логіки та механізмів взаємодії процесів [10, с. 115].

Отримані результати узгоджуються із сучасними дослідженнями у сфері *formal methods education*, у яких наголошується, що використання формальних методів сприяє більш глибокому засвоєнню програмних концепцій і може інтегруватися у структуру професійної підготовки без порушення цілісності навчального курсу [4]. У зв'язку з цим, формальні моделі виступають засобом візуалізації системних процесів та інструментом розвитку логічного мислення, навичок верифікації та формалізованого опису поведінки операційних систем.

Порівняння отриманих результатів із традиційними підходами до викладання операційних систем засвідчує, що формалізовані методики доповнюють практико-орієнтовані моделі навчання, які переважно ґрунтуються на демонстрації реальних операційних систем або реалізації спрощених мікроядер. На відміну від *hands-on approaches*, формальні методи концентрують увагу студентів на інваріантних принципах функціонування ОС, зокрема механізмах синхронізації, розподілу ресурсів, взаємного виключення та управління процесами [4]. З огляду на це, найбільш результативним є поєднання формальних моделей із лабораторними та практичними завданнями, де перші забезпечують концептуально-теоретичне підґрунтя, а другі – демонструють прикладне використання відповідних моделей у контексті функціонування реальних систем.

Водночас результати аналізу свідчать про необхідність дотримання балансу між рівнем формалізації та прикладною спрямованістю навчання. Надмірна деталізація формальних моделей може ускладнювати процес засвоєння матеріалу та знижувати зв'язок між абстрактними моделями й реальними механізмами функціонування операційних систем. У дослідженні Y. Jiang відповідна проблема розглядається як ризик надмірної абстрагованості навчальних моделей від практичної архітектури реальних ОС [7]. У зв'язку з цим, використання *formal methods* доцільно поєднувати з практичними лабораторними завданнями, аналізом реальних системних механізмів та прикладами прикладного програмування.

Наукова новизна (розгорнуто). Наукова новизна дослідження полягає у теоретичному обґрунтуванні та систематизації методичних засад навчання формальних моделей операційних систем у контексті викладання дисципліни «Операційні системи». У дослідженні здійснено комплексний порівняльний аналіз сучасних підходів до використання графових моделей, автоматних формалізмів, сіток Петрі та мов формальної специфікації у процесі професійної підготовки майбутніх ІТ-фахівців, що дало можливість визначити їх дидактичні функції, рівень формалізації та методичні особливості інтеграції у навчальний процес.

Уточнено методичні підходи до формалізації процесів функціонування операційних систем шляхом поєднання теоретичних моделей із практично орієнтованими формами навчальної діяльності, зокрема лабораторним моделюванням системних процесів, побудовою графів станів, аналізом deadlock-ситуацій та використанням засобів *model checking*. На відміну від традиційних підходів, у межах яких вивчення операційних систем переважно орієнтується на опис архітектурних компонентів або відтворення готових системних рішень, запропонований підхід передбачає інтеграцію формальних апаратів у процес аналізу системної поведінки та моделювання механізмів функціонування ОС.

Набуло подальшого розвитку положення щодо використання *formal methods* як засобу формування системного мислення, логічного аналізу та навичок формалізованого опису системних процесів. Доведено, що застосування автоматних моделей, графів ресурсів і сіток Петрі забезпечує перехід від описового трактування механізмів функціонування операційних систем до їх логічно та математично структурованого аналізу, що сприяє формуванню у студентів здатності до верифікації

системних станів, аналізу синхронізації процесів та формалізації алгоритмів взаємодії ресурсів.

Удосконалено підходи до методичного структурування змісту навчання операційних систем шляхом інтеграції формальних моделей у вивчення процесів диспетчеризації, взаємного виключення, синхронізації та розподілу ресурсів, що забезпечує концептуальне поєднання положень теоретичної інформатики та прикладних аспектів системного програмування у структурі професійної підготовки майбутніх ІТ-фахівців.

Практичне значення (розгорнуто). Практична значущість дослідження полягає у можливості використання запропонованих методичних підходів у процесі викладання дисципліни «Операційні системи» у закладах вищої освіти. Результати дослідження можуть бути використані під час оновлення змісту навчальних курсів, розроблення лабораторних практикумів та інтеграції formal methods у професійну підготовку майбутніх ІТ-фахівців.

Запропоновані підходи до використання моделі Холта, автоматних моделей і сіток Петрі забезпечують підвищення рівня наочності системних процесів та сприяють формуванню у студентів навичок логічного моделювання, аналізу синхронізації процесів, розподілу ресурсів і формалізованого опису поведінки операційних систем.

Порівняння з іншими дослідженнями. Порівняно з традиційними підходами до викладання операційних систем, формалізовані методики не замінюють практичного ознайомлення з реальними програмними системами, а виконують функцію концептуально-теоретичного підґрунтя для їх глибшого розуміння. У сучасних дослідженнях підкреслюється, що formal methods сприяють формуванню системного бачення архітектури програмного забезпечення та забезпечують краще розуміння принципів проектування комп'ютерних систем [11]. Отримані результати підтверджують необхідність розгляду формальних моделей операційних систем у контексті теоретичної інформатики, оскільки використання автоматних, графових та мережевих формалізмів забезпечує встановлення концептуального зв'язку між абстрактними теоретичними конструкціями та практикою функціонування операційних систем.

Висновки

1. У результаті дослідження здійснено теоретичне узагальнення методичних засад навчання формальних моделей операційних систем у контексті теоретичної інформатики. Встановлено, що використання автоматних моделей, графів ресурсів, сіток Петрі та мов формальної специфікації забезпечує підвищення рівня формалізації навчального процесу, сприяє поглибленому розумінню механізмів функціонування ОС та формуванню у студентів навичок логічного аналізу, верифікації системних станів і моделювання процесів синхронізації та розподілу ресурсів.

2. Проведений порівняльний аналіз сучасних підходів до викладання формальних моделей ОС дозволив визначити їх дидактичні можливості, методичні переваги та обмеження. Доведено доцільність поєднання формальних методів із практично орієнтованими формами навчання, зокрема лабораторним моделюванням системних процесів і використанням засобів model checking. Запропоновані методичні підходи можуть бути використані для оновлення змісту дисципліни «Операційні системи» та інтеграції formal methods у професійну підготовку майбутніх ІТ-фахівців.

3. Перспективи подальших досліджень пов'язані з емпіричною перевіркою ефективності запропонованих методичних підходів у навчальному процесі, розробленням цифрових засобів моделювання та верифікації системних процесів, а також адаптацією методики навчання формальних моделей операційних систем до

різних рівнів підготовки здобувачів вищої освіти та спеціалізацій у галузі інформаційних технологій.

Список використаних джерел

1. Зайцев В. Г., Дробязко І. П. Операційні системи: навч. посіб. для студ. спеціальності 123 «Комп'ютерна інженерія». Київ : КПІ ім. Ігоря Сікорського, 2019. 240 с.
2. Семеріков С. О. Методичні основи вивчення теми «Операційні системи» у підготовці майбутнього вчителя. 2003. URL: https://lib.iitta.gov.ua/id/eprint/704179/1/stat_os.pdf (дата звернення: 12.02.2026).
3. Arronategui U. Petri nets as a tool for teaching operating systems. ACM SIGCSE Bulletin. 1998. Vol. 30, No. 3. P. 171–175. DOI: <https://doi.org/10.1145/290320.283109>.
4. Dongol B., Dubois C., Hallerstede S., Hehner E., Morgan C., Müller P., Ribeiro L., Silva A., Smith G., de Vink E. On formal methods thinking in computer science education. Formal Aspects of Computing. 2024. DOI: <https://doi.org/10.1145/3670419>.
5. FME (Formal Methods Europe). URL: <https://fmeurope.org> (date of access: 12.02.2026).
6. FME Formal Methods Teaching Committee. FME Education Course Database. URL: <https://fme-teaching.github.io/courses/> (date of access: 12.02.2026).
7. Jiang Y. The Hitchhiker's Guide to Operating Systems. In: 2023 USENIX Annual Technical Conference (USENIX ATC 23). Berkeley : USENIX Association, 2023. P. 929–944. URL: <https://www.usenix.org/system/files/atc23-jiang-yanyan.pdf> (date of access: 12.02.2026).
8. Jiang Y., Wang X., Zhang H., Li X., Li Z., Gu R. The Hitchhiker's Guide to Operating Systems. In: 2023 USENIX Annual Technical Conference (USENIX ATC 23). Berkeley: USENIX Association, 2023. P. 823–837. URL: <https://www.usenix.org/system/files/atc23-jiang-yanyan.pdf> (date of access: 12.02.2026).
9. Lane M. G. Teaching operating systems and machine architecture - more on the hands-on laboratory approach. In: Proceedings of the Twelfth SIGCSE Technical Symposium on Computer Science Education (SIGCSE '81). New York : Association for Computing Machinery, 1981. P. 28–36.
10. Peterson G. L. Myths about the mutual exclusion problem. *Information Processing Letters*. 1981. Vol. 12, No. 3. P. 115–116.
11. ter Beek M., Broy M., Dongol B. The role of formal methods in computer science education. *ACM Inroads*. 2024. Vol. 15, No. 4. URL: <https://inroads.acm.org/article.cfm?aid=3702231> (date of access: 12.02.2026).
12. Yang J., Sar C., Engler D. EXPL0DE: A lightweight, general system for finding serious storage system errors. In: 7th USENIX Symposium on Operating Systems Design and Implementation (OSDI 06). Seattle : USENIX Association, 2006.

References

1. Zaitsev V. H., Drobiazko I. P. (2019) Operatsiini systemy [Operating systems: a textbook for students of specialty 123 "Computer engineering"]: navch. posib. dlia stud. spetsialnosti 123 «Kompiuterna inzheneriia». Kyiv : KPI im. Ihoria Sikorskoho.
2. Semerikov S. O. (2003) Metodychni osnovy vyvchennia temy «Operatsiini systemy» u pidhotovtsi maibutnoho vchytelia [Methodological foundations of studying the topic "Operating systems" in the training of future teachers].. URL: https://lib.iitta.gov.ua/id/eprint/704179/1/stat_os.pdf (data zvernennia: 12.02.2026).
3. Arronategui U. (1988) Petri nets as a tool for teaching operating systems. ACM SIGCSE Bulletin, 30 (3). 171–175. DOI: <https://doi.org/10.1145/290320.283109>.

4. Dongol B., Dubois C., Hallerstede S., Hehner E., Morgan C., Müller P., Ribeiro L., Silva A., Smith G., de Vink E. (2024) On formal methods thinking in computer science education. *Formal Aspects of Computing*. DOI: <https://doi.org/10.1145/3670419>.
5. FME (Formal Methods Europe). URL: <https://fmeurope.org> (date of access: 12.02.2026).
6. FME Formal Methods Teaching Committee. FME Education Course Database. URL: <https://fme-teaching.github.io/courses/> (date of access: 12.02.2026).
7. Jiang Y. (2023) The Hitchhiker's Guide to Operating Systems. In: 2023 USENIX Annual Technical Conference (USENIX ATC 23). Berkeley : USENIX Association, 929–944. URL: <https://www.usenix.org/system/files/atc23-jiang-yanyan.pdf> (date of access: 12.02.2026).
8. Jiang Y., Wang X., Zhang H., Li X., Li Z., Gu R. (2023) The Hitchhiker's Guide to Operating Systems. In: 2023 USENIX Annual Technical Conference (USENIX ATC 23). Berkeley: USENIX Association, 823–837. URL: <https://www.usenix.org/system/files/atc23-jiang-yanyan.pdf> (date of access: 12.02.2026).
9. Lane M. G. (1981) Teaching operating systems and machine architecture - more on the hands-on laboratory approach. In: *Proceedings of the Twelfth SIGCSE Technical Symposium on Computer Science Education (SIGCSE '81)*. New York : Association for Computing Machinery, 28–36.
10. Peterson G. L. (1981) Myths about the mutual exclusion problem. *Information Processing Letters*, 12 (3), 115–116.
11. ter Beek M., Broy M., Dongol B. (2024) The role of formal methods in computer science education. *ACM Inroads*, 15 (4). URL: <https://inroads.acm.org/article.cfm?aid=3702231> (date of access: 12.02.2026).
12. Yang J., Sar C., Engler D. (2006) EXPLODE: A lightweight, general system for finding serious storage system errors. In: 7th USENIX Symposium on Operating Systems Design and Implementation (OSDI 06). Seattle : USENIX Association.