

**Ничипорчук Валерій,**  
 здобувач першого (бакалаврського)  
 рівня вищої освіти  
 фізико-математичного  
 факультету, Житомирський державний  
 університет імені Івана Франка,  
 м. Житомир  
**Яценко Оксана**  
 асистент кафедри комп'ютерних  
 наук та інформаційних технологій,  
 Житомирський державний  
 університет імені Івана Франка,  
 м. Житомир

### **ХМАРНІ ТЕХНОЛОГІЇ У ВИКЛАДАННІ ПРОГРАМУВАННЯ**

Викладання програмування змінюється разом зі світом. Якщо ще десять років тому студентам достатньо було мати робочий комп'ютер і базове середовище розробки, то сьогодні цього вже замало. Зміни, які принесла цифрова трансформація освіти прискорена пандемією і, як

результат, розвитком дистанційного навчання, стимулювали пошук більш гнучких і технологічно зручних рішень. І тут на перший план виходять хмарні технології.

В наш час хмара – це не лише про зберігання файлів. У програмуванні – це робоче місце, яке завжди зі студентом чи розробником. Це доступ до інтегрованого середовища розробки (ICP) у браузері, командної співпраці, автоматичних збірок і тестування коду. І все це без прив'язки до конкретного комп'ютера. Хмарні середовища поступово стали стандартом у викладанні програмування, особливо в умовах дистанційного або змішаного навчання. Замість класичного підходу, коли студенти встановлюють ICP на власні комп'ютери, усе частіше використовуються вебплатформи, що дозволяють почати писати код без жодних технічних налаштувань.

Такі середовища стали не просто зручними інструментами, а докорінно змінили підхід до навчання. Студент більше не прив'язаний до аудиторії чи комп'ютерного класу. Він може навчатись з дому, у коворкінгу чи під час подорожі – головне, щоб був доступ до Інтернету. Це не лише розширює освітні можливості, але й розвиває у студентів гнучкість, самостійність і навички роботи в розподілених командах, що важливо для сучасної ІТ-індустрії.

У багатьох закладах вищої освіти вже застосовують такі інструменти як *Replit*, *GitHub Codespaces*, *Google Colab* або *Visual Studio Code* у браузері через *Gitpod*. Так широко використовується платформа *Replit* (<https://replit.com/>). *Replit* – це онлайн-платформа для створення програмного забезпечення, яка дозволяє писати, запускати та ділитися кодом без потреби встановлювати будь-яке програмне забезпечення на комп'ютер. Її назва походить від аббревіатури REPL (Read-eval-print loop – цикл читання-обчислення-друку), що означає середовище, у якому користувач уводить програмний код і відразу отримує обчислення та виведений результат. Ця платформа дозволяє студентам працювати навіть зі смартфонів або малопотужних ноутбуків. Викладачі можуть створити класи (Teams) всередині платформи та слідкувати за прогресом кожного учня.

Дослідження проведене в College of Computing and Informatics, University Technology MARA Sarawak Branch [1] показало, що студенти високо оцінили *Replit* за його доступність, інтерактивність та здатність сприяти командній роботі над проектами. Дослідники зробили висновок, що використання *Replit* сприяє підвищує рівень зацікавленості студентів та збільшує їх залученість в процес навчання.

Інший приклад – *Google Colab* (<https://colab.research.google.com/>), який дозволяє запускати складні обчислення на віддалених серверах

Google, що особливо важливо для роботи з великими масивами даних. Під час використання *Colab* студенти можуть легко ділитися своїми роботами з викладачами для перевірки завдань, що значно спрощує комунікацію. Платформа підтримує підключення до Google Drive, дозволяє імпортувати бібліотеки, використовувати GPU або TPU для обчислень, а також автоматично зберігає всі зміни. Платформа особливо зручна для виконання лабораторних і домашніх завдань: можна додавати текстові пояснення, графіки, завантажувати дані та навіть працювати над проектами у групі завдяки спільному доступу.

Загалом Google Colab являє собою потужне та зручне для навчання хмарне середовище для роботи з Jupyter Notebook, що дозволяє програмістам і дослідникам проводити обчислення, аналіз даних, машинне навчання та інші завдання прямо в хмарі, не витрачаючи час на налаштування власної інфраструктури. Colab підтримує популярні бібліотеки Python, має зручний інтерфейс і надає можливість спільної роботи над проектами. Для програмістів і дослідників Google Colab стає незамінним інструментом для підвищення ефективності роботи та спрощення процесу розробки і досліджень [2]. Результати дослідження [3] показують, що використання Google Colab значно підвищує інтерес до програмування, а інтеграція GenAI відповідає їхнім очікуванням з точки зору освітньої підтримки.

У межах командного навчання все більшої популярності набуває *GitHub Codespaces* (<https://github.com/features/codespaces>) – сервіс від GitHub, що дозволяє розгортати контейнерне хмарне середовище виконання репозиторію, в якому можна не лише розробляти, але й запускати код. Ця платформа дозволяє кожному студенту мати свій окремий контейнер із попередньо налаштованим середовищем розробки. Це зручно в рамках роботи над груповими проектами, коли потрібно одночасно писати код, тестувати його та зберігати історію змін.

Проведене дослідження [4] показує, що використання GitHub Codespaces для вивчення мов програмування зменшує труднощі як для студентів, так і для викладачів, дозволяє викладачам охоплювати більше відповідного матеріалу курсу, а також підвищує залученість та задоволення студентів від заняття, а також впевненість у кодуванні. Водночас, GitHub Classroom є ефективним інструментом для розповсюдження початкового коду шаблону Codespace серед студентів, а також допомагає викладачам краще організовувати та оцінювати подані завдання.

Деякі навчальні платформи, наприклад *Stepik* або *Coursera*, інтегрують хмарні редактори безпосередньо у навчальні модулі. Це означає, що студент може читати теорію та одразу виконувати практичні

завдання в інтерактивному середовищі без переходу між вкладками чи встановлення додаткового ПЗ.

З появою хмарних середовищ змінюється й роль викладача. Тепер він не лише передає знання, а й стає фасилітатором – тим, хто організовує середовище, створює виклики, стимулює роботу в команді. Викладач має змогу бачити, як студент працює з кодом у реальному часі, підказати, скоригувати, дати швидкий зворотний зв'язок. Це знижує рівень фрустрації студентів, особливо новачків, і сприяє формуванню більш дружнього та продуктивного навчального простору.

Попри ентузіазм, хмарні рішення не позбавлені проблем. Наприклад, не всі студенти мають стабільний інтернет – а без нього жодне хмарне середовище не працюватиме. Іноді виникають питання захисту даних – особливо в тих випадках, коли працюють з конфіденційними матеріалами або API-ключами. Ще одним зі значних є можливість несподіваної монетизації сервісу – в будь-який момент компанія може почати стягувати плату за послуги та можливості, які були безкоштовними. Крім того не можна ігнорувати певну «розмитість» навичок: працюючи в браузері, учні можуть не дізнатися, як налаштувати локальне середовище, що також важливо. Тому хмарні технології варто сприймати не як панацею, а як інструмент, який варто грамотно поєднувати з традиційними підходами.

Проаналізувавши зазначене вище можна зробити висновок, що найбільший плюс хмарних технологій – це гнучкість. Їх можна інтегрувати у дистанційне, змішане або навіть очне навчання. Студенти та учні отримують можливість працювати над спільними проєктами, вчать співпраці, а також мають змогу легко демонструвати свої результати однокласникам, вчителям та навіть потенційним роботодавцям.

Хмарні технології – це не мода, а необхідність у сучасній освіті, особливо у сфері програмування. Вони дозволяють зробити навчання гнучким, доступним і технологічно актуальним. Проте ключовим залишається баланс: використання хмарних рішень має йти разом із критичним аналізом їх ефективності та готовністю адаптувати методику викладання до нових умов. Викладачеві XXI століття важливо не лише знати про нові інструменти, а й вміти їх інтегрувати в реальний навчальний процес.

### **Список використаних джерел та літератури**

1. Engkamat A., Yii M. L., Gran S. S. Replit: a simple approach to real-time collaborative coding. *International Teaching Aid Competition 2023*. Kedah, 2023. P. 882–886. URL: <https://ir.uitm.edu.my/id/eprint/84984/> (дата звернення: 10.05.2025).

2. Налаштування середовища виконання в Google Colab: поради та лайфхаки. URL: <https://foxminded.ua/google-colab/> (дата звернення: 10.05.2025).

3. Innovations in Introductory Programming Education: The Role of AI with Google Colab and Gemini / J. Llerena-Izquierdo et al. *Education Sciences*. 2024. Vol. 14, no. 12. P. 19. URL: <https://doi.org/10.3390/educsci14121330> (date of access: 10.05.2025).

4. Snowberger A. D., You K. Creating a Standardized Environment for Efficient Learning Management using GitHub Codespaces and GitHub Classroom. *Journal of Practical Engineering Education*. 2024. Vol. 16, no. 3. P. 267–274. URL: <https://doi.org/10.14702/JPEE.2024.267> (date of access: 10.05.2025).