

ідеї. Крім цього, актуалізуються й готовність кожного учасника STEM-проекту приймати відповідальність за прийняте ним рішення, уміння розв'язувати проблеми, а також здатність до командної роботи.

Таким чином, STEM-освіта як інтегрована модель навчання на рівні сучасної базової школи (5–9 класи) здатна забезпечити цілеспрямований розвиток таких ключових компетентностей як-от: «вільне володіння державною мовою», «математична компетентність», «компетентності у галузі природничих наук, техніки і технологій», «інноваційність», «екологічна компетентність», «інформаційно-комунікаційна компетентність» та «підприємливість і фінансова грамотність». Розвиток решти чотирьох ключових компетентностей у рамках STEM-освіти відбувається несвідомо або дотично.

Список використаних джерел

1. Барна О. В., Балик Н. Р. Впровадження STEM-освіти у навчальних закладах: етапи та моделі. Тези доповідей I регіональної науково-практичної веб-конференції «STEM-освіта та шляхи її впровадження в навчально-виховний процес» (Тернопіль, 24 травня, 2017 р.). Тернопіль : ТОКІППО, 2017. С. 3–8. URL: <http://elar.ippo.edu.te.ua:8080/handle/123456789/4559> (дата звернення: 17.10.2025).
2. Бирка М. Ф. Бар'єри і виклики на шляху успішного впровадження STEM освіти в Україні. STEM-освіта та шляхи її впровадження в навчально-виховний процес: збірник матеріалів I регіональної науково-практичної веб-конференції (Тернопіль, 24 травня, 2017 р.). Тернопіль : ТОКІППО, 2017. С. 9–13.
3. Бирка М. Ф. Бар'єри, виклики та принципи ефективної реалізації STEM освіти в Україні. *Наукові записки Малої академії наук України*. Серія: Педагогічні науки, 2018. С. 6–24.
4. Державний стандарт базової середньої освіти. Постанова КМУ № 898 від 30.09.2020 року. URL: <http://surl.li/zfthkl> (дата звернення: 19.10.2025).
5. Колток Л., Іваник Н. Впровадження STEM-освіти в освітній процес Нової української школи. *Актуальні питання гуманітарних наук*, 2020. Т. . № 27. С. 133–136.

ОСОБЛИВОСТІ ВИВЧЕННЯ МОВ ПРОГРАМУВАННЯ УЧНЯМИ ЗАКЛАДІВ ЗАГАЛЬНОЇ СЕРЕДНЬОЇ ОСВІТИ (7–9 КЛАСИ)

Валігура Михайло Ігорович

здобувач другого рівня вищої освіти, спеціальність Середня освіта (Інформатика)
Тернопільський національний педагогічний університет імені Володимира Гнатюка
vmisha200113@gmail.com

Мартинюк Сергій Володимирович

кандидат фізико-математичних наук, доцент кафедри інформатики та методики її навчання
Тернопільський національний педагогічний університет імені Володимира Гнатюка
sergmart65@tnpu.edu.ua

Сучасна загальна середня освіта переживає період інтенсивної цифровізації, у якому вміння працювати з інформаційними технологіями та розуміти основи алгоритмічного мислення стає не просто корисним, а необхідним компонентом базової компетентності школяра. Водночас шкільний курс інформатики часто зосереджений на репродуктивних навичках: запам'ятовуванні термінів, відтворенні прикладів з підручника та виконанні ізольованих вправ. Це призводить до ситуації, коли учні формально опановують синтаксис мов програмування або окремі

конструкції, але не набувають стійкої здатності проєктувати алгоритми, відлагоджувати код і застосовувати набуті знання для вирішення практичних і творчих завдань.

У навчальному процесі з інформатики для учнів 7–9 класів особливо гостро виявляються протиріччя між потребою формувати у школярів практичні програмістські уміння і наявними методичними підходами, що часто не враховують вікових психологічних особливостей підлітків, їхню мотивацію та потреби в діяльнісному засвоєнні матеріалу. Традиційні методи недостатньо стимулюють розвиток алгоритмічного мислення і творчого підходу, а також не забезпечують належні умови для поступового переходу від розуміння базових понять до самостійної розробки проєктів. У результаті багато учнів втрачають інтерес до предмета або залишаються неготовими до подальшого вивчення програмування в старшій школі [1].

Отже, виникає нагальна потреба в розробці й апробації методик, які б поєднували поетапність засвоєння знань з діяльнісним підходом, забезпечували б системну диференціацію завдань за рівнем складності та включали б механізми рефлексії й самооцінки. Особливо актуальним є створення такої методичної моделі, яка б на базі перевірених педагогічних принципів адаптувала класичні підходи (зокрема, ідеї таксономії Блума) до специфіки вивчення мов програмування у середній школі, зробивши акцент на практичній спрямованості, мотивації та поступовій інтеграції навчальних проєктів.

Таким чином, проблемою даного дослідження є пошук ефективної методики формування умінь програмування в учнів 7–9 класів, яка б пододала обмеження традиційного підходу, сприяла послідовному розвитку алгоритмічного та творчого мислення і давала змогу виробити в учнів практичні навички програмування, необхідні для успішного навчання на подальших етапах здобуття освіти. Ця проблема визначила мету дослідження – розробити, впровадити й експериментально перевірити адаптовану модель навчання, орієнтовану на поетапне формування умінь програмування.

У процесі дослідження було розроблено та апробовано методику формування умінь програмування учнів 7–9 класів, побудовану на засадах поетапного навчання, діяльнісного підходу та принципах адаптованої таксономії програмування. Її концепція ґрунтувалася на тому, що ефективне оволодіння програмуванням передбачає поступовий перехід від розуміння теоретичних основ до самостійного створення програмних продуктів [2].

Методика включала п'ять послідовних етапів формування умінь програмування. Перший етап – усвідомлення – був спрямований на засвоєння базових понять, синтаксису мови, основних типів даних і структур керування. На цьому рівні акцент робився на поясненні, демонстрації, роботі з простими прикладами, що допомогло сформувати у школярів початкове розуміння логіки програмування.

Другий етап – застосування – передбачав написання учнями простих програм за зразком, виконання алгоритмів, поданих у псевдокоді або блок-схемах. Учні вчилися реалізовувати базові конструкції мови програмування та виконувати завдання, аналогічні до розглянутих на уроці.

Третій етап – конструювання – був спрямований на розвиток самостійності мислення. Учні створювали програми, у яких поєднували кілька логічних структур, самостійно обирали способи розв’язання задач, аналізували отримані результати.

На четвертому етапі – аналіз і вдосконалення – школярі навчалися знаходити й виправляти помилки у власних програмах, оптимізувати код, працювати з відлагодженням програми. Це сприяло формуванню критичного мислення, уважності та навичок самоконтролю.

П’ятий етап – творче застосування – передбачав розробку власних мініпроектів. Учні отримували змогу проявити ініціативу, розробляючи невеликі програми або інтерактивні застосунки, які мали практичне або навчальне значення. Саме на цьому рівні найповніше розкривалася особистісна мотивація до програмування.

Практична реалізація методики здійснювалася під час педагогічного експерименту в 8 класі, який було розділено на дві підгрупи – контрольну й експериментальну. Контрольна підгрупа навчалася за традиційною методикою, тоді як експериментальна працювала за розробленою моделлю. Упродовж навчального семестру проводилося спостереження за навчальною діяльністю, виконання тестових і проєктних завдань, а також анкетування учнів для виявлення змін у мотивації.

Результати підсумкового оцінювання показали, що учні експериментальної групи продемонстрували істотно вищі показники сформованості алгоритмічного мислення, умінь аналізувати код і розробляти власні програми. Крім того, спостерігалось зростання інтересу до предмета, підвищення самостійності та впевненості під час виконання завдань.

Отримані дані підтвердили, що впровадження поетапної методики сприяє глибшому розумінню навчального матеріалу, підвищенню навчальної мотивації та розвитку творчого потенціалу учнів. У порівнянні з традиційним підходом, запропонована модель забезпечує системність формування вмінь програмування, дозволяючи школярам поступово перейти від простих дій за зразком до самостійного створення повноцінних проєктів.

Проведене дослідження підтвердило ефективність розробленої методики формування умінь програмування в учнів 7–9 класів, побудованої на основі адаптованої таксономії програмування. У ході експерименту встановлено, що поетапне формування знань і навичок – від усвідомлення базових понять до творчого застосування – забезпечує глибше засвоєння матеріалу, розвиток алгоритмічного мислення та підвищення інтересу до предмета.

Учні, які навчалися за запропонованою методикою, не лише краще засвоїли синтаксис і структуру мови програмування, але й продемонстрували здатність самостійно розробляти прості програми, аналізувати власні помилки й оптимізувати код. Важливою перевагою методики стало поєднання навчальної діяльності з елементами творчості, що сприяло зростанню мотивації й позитивного ставлення до програмування.

Таким чином, запропонована модель може бути ефективно використана у навчальному процесі закладів загальної середньої освіти для формування практичних навичок програмування та розвитку ключових компетентностей учнів у галузі інформаційних технологій. Отримані результати дають підстави стверджувати, що така методика є доцільною для впровадження в навчальні програми з інформатики

середньої школи та може стати основою для подальшого вдосконалення підходів до навчання програмування.

Список використаних джерел

1. Кобильник Т., Когут У., Жидик В. Методичні аспекти вивчення основ алгоритмізації і програмування мовою Python у шкільному курсі інформатики у старших класах». *Фізико-математична освіта*, 2021, Т. 31. № 5. С. 36–44.

2. Валігура М. І., Мартинюк С. В. Формування алгоритмічного мислення учнів 7–9 класів засобами мов програмування. *Сучасні цифрові технології та інноваційні методики навчання: досвід, тенденції, перспективи*: матеріали XV Міжнародної науково-практичної інтернет-конференції, м. Тернопіль, 10 квітня, 2025 р. Тернопіль: ТНПУ ім. В. Гнатюка, 2025. С. 250–252.

ІНТЕРАКТИВНІ ОНЛАЙН-ІНСТРУМЕНТИ ДЛЯ ОЦІНЮВАННЯ РІВНЯ НАВЧАЛЬНИХ ДОСЯГНЕНЬ УЧНІВ: ДОСВІД І ПЕРСПЕКТИВИ ВИКОРИСТАННЯ В КУРСІ ІНФОРМАТИКИ

Вербіцький Іван Володимирович

здобувач другого рівня вищої освіти, спеціальність Середня освіта (Інформатика)
Тернопільський національний педагогічний університет імені Володимира Гнатюка
ivan.verbitskij@gmail.com

Мартинюк Сергій Володимирович

кандидат фізико-математичних наук, доцент кафедри інформатики та методики її навчання,
Тернопільський національний педагогічний університет імені Володимира Гнатюка
sergmart65@tnpu.edu.ua

Цифрова трансформація освіти відкриває нові можливості для організації навчального процесу, зокрема для контролю та оцінювання рівня навчальних досягнень здобувачів освіти. Якщо раніше система оцінювання базувалася переважно на усних або письмових формах контролю, то сьогодні вона дедалі активніше інтегрується з інтерактивними цифровими інструментами. Це зумовлено не лише розвитком технологій, а й зміною парадигми навчання: від традиційного викладання до активного, особистісно орієнтованого та компетентнісного підходів. Як зазначає В. Биков, розвиток цифрового освітнього середовища є ключовим фактором модернізації навчального процесу в Україні [2, с. 12].

Актуальність теми зумовлена потребою в адаптації сучасних методик оцінювання до умов дистанційного та змішаного навчання, що стало особливо помітним у період пандемії COVID-19 і подальших воєнних викликів, які змусили українську школу швидко перейти на нові цифрові платформи [3, с. 25]. Водночас такі виклики стимулювали педагогів шукати інноваційні рішення, здатні забезпечити безперервність навчання та підтримувати інтерес учнів до предмету. Як підкреслює О. Пінчук, інтерактивні технології в умовах цифрової трансформації освіти не лише підвищують ефективність навчання, але й створюють основу для формування нової культури взаємодії між учителем і учнем [4, с. 53].

Оцінювання перестає бути лише формальним інструментом перевірки знань – воно стає важливою складовою навчального діалогу, що забезпечує зворотний зв'язок, формує мотивацію та допомагає учневі усвідомити власні досягнення й