

Секція: Початкова освіта	
УДК 378.147:373.3.011.3-051:004.021	
Дата першого надходження статті до видання	2026-08-16
Дата прийняття статті до друку після рецензування	2026-09-21
Дата публікації/оприлюднення	2026-09-21

ПРОГРАМУВАННЯ У SCRATCH ЯК ЗАСІБ ФОРМУВАННЯ ІНФОРМАЦІЙНО-ЦИФРОВОЇ КОМПЕТЕНТНОСТІ МОЛОДШИХ ШКОЛЯРІВ

Василиків Іван Богданович

кандидат педагогічних наук, доцент,

доцент кафедри фундаментальних дисциплін початкової освіти,

Дрогобицький державний педагогічний університет імені Івана Франка,

Дрогобич, Україна

<https://orcid.org/0000-0002-9220-1736>

Анотація. У статті теоретично обґрунтовано можливості використання візуального середовища програмування Scratch як засобу формування інформаційно-цифрової компетентності молодших школярів. Актуальність дослідження зумовлена потребою переходу від епізодичного використання цифрових інструментів до системної організації діяльності, у якій дитина не лише споживає готовий цифровий контент, а створює власні інформаційні продукти, алгоритми, інтерактивні історії, анімації та навчальні проекти. Метою статті є визначення дидактичного потенціалу Scratch і побудова моделі формування інформаційно-цифрової компетентності учнів 2–4 класів у процесі навчання програмування. Методологічну основу становлять компетентнісний, діяльнісний, конструктивістський і проектний підходи. Використано аналіз нормативних документів і наукових джерел, систематизацію, порівняння, педагогічне моделювання та структурно-функціональний аналіз. Уточнено структуру інформаційно-цифрової компетентності в контексті роботи у Scratch та виділено п'ять взаємопов'язаних компонентів: інформаційно-пошуковий, алгоритмічний, технологічно-операційний, творчо-проектувальний і рефлексивно-безпековий. Обґрунтовано, що педагогічний потенціал Scratch полягає не лише в засвоєнні елементарних основ програмування, а й у розвитку здатності визначати мету цифрової діяльності, планувати послідовність дій, добирати та створювати цифрові об'єкти, конструювати й налагоджувати алгоритми, аналізувати помилки, перевіряти результат, презентувати цифровий продукт і дотримуватися правил безпечної та відповідальної поведінки у цифровому середовищі. Запропоновано послідовність навчальної діяльності «зрозуміти завдання – спланувати – створити алгоритм – реалізувати у Scratch – перевірити – удосконалити – представити результат». Розроблено систему типових навчальних завдань і критерії оцінювання цифрових проектів молодших школярів. Наукова новизна полягає у представленні програмування у Scratch як інтегрованого засобу формування інформаційно-цифрової компетентності, у якому оволодіння програмними конструкціями підпорядковується створенню осмисленого цифрового продукту. Практичне значення результатів полягає в можливості використання запропонованої моделі на уроках інформатики, інтегрованого курсу «Я досліджую світ», під час проектної та позакласної діяльності.

Ключові слова: Scratch, програмування, початкова школа, інформаційно-цифрова компетентність, цифрова компетентність, алгоритмічне мислення, інформаційні технології, молодші школярі, проектне навчання, цифровий освітній продукт.

PROGRAMMING IN SCRATCH AS A MEANS OF DEVELOPING PRIMARY SCHOOL PUPILS' INFORMATION AND DIGITAL COMPETENCE**Ivan Bohdanovych Vasylykiv**

Candidate of Pedagogical Sciences, Associate Professor,
Associate Professor at the Department of Fundamental Disciplines of Primary Education,
Drohobych Ivan Franko State Pedagogical University,
Drohobych, Ukraine

<https://orcid.org/0000-0002-9220-1736>

Abstract. The article theoretically substantiates the potential of the Scratch visual programming environment as a means of developing information and digital competence among primary school pupils. The relevance of the study is determined by the need to move from episodic use of digital tools to a systematic organisation of learning activities in which children not only consume ready-made digital content but also create their own information products, algorithms, interactive stories, animations, and educational projects. The aim of the article is to identify the didactic potential of Scratch and to construct a model for developing information and digital competence of pupils in grades 2–4 through programming activities. The methodological basis combines competence-based, activity-based, constructivist, and project-based approaches. The methods include analysis of regulatory documents and scientific publications, systematisation, comparison, pedagogical modelling, and structural-functional analysis. Five interconnected components of information and digital competence are distinguished: information-search, algorithmic, technological-operational, creative-project, and reflective-safety. The pedagogical potential of Scratch is shown to include not only mastering elementary programming but also developing the ability to define the purpose of digital activity, plan sequences of actions, select and create digital objects, construct and debug algorithms, analyse errors, test results, present a digital product, and follow principles of safe and responsible behaviour in digital environments. A learning sequence is proposed: “understand the task – plan – construct an algorithm – implement it in Scratch – test – improve – present the result”. A system of typical learning tasks and criteria for assessing pupils’ digital projects is developed. The scientific novelty lies in considering Scratch programming as an integrated means of developing information and digital competence, in which the mastery of programming constructs is subordinated to the creation of a meaningful digital product. The practical significance lies in the possibility of applying the proposed model in Informatics lessons, the integrated course “I Explore the World”, project-based learning, and extracurricular activities.

Keywords: Scratch, programming, primary school, information and digital competence, digital competence, algorithmic thinking, information technologies, primary school pupils, project-based learning, digital educational product.

Вступ**Актуальність проблеми.**

Цифрова трансформація освіти змінює не лише перелік технічних засобів, які використовує вчитель, а й характер навчальної діяльності дитини. Для початкової школи принципово важливо, щоб цифрові технології не зводилися до демонстрації готових матеріалів або механічного виконання інтерактивних вправ. Учень має поступово переходити від споживання інформації до її осмисленого опрацювання, моделювання, створення, перевірки та представлення. Саме такий перехід формує основу інформаційно-цифрової компетентності як здатності ефективно, безпечно і відповідально діяти у цифровому середовищі.

У типових освітніх програмах для 1–4 класів Нової української школи інформатична освітня галузь орієнтує навчання на формування в учнів здатності працювати з інформацією, цифровими пристроями, алгоритмами та інформаційними

моделями, створювати цифрові продукти й дотримуватися правил безпечної діяльності в цифровому середовищі [18; 19]. Отже, педагогічне завдання полягає не лише в ознайомленні дитини з інтерфейсом програмного засобу, а у формуванні здатності розв'язувати навчальні та життєві задачі із застосуванням цифрових технологій.

Одним із середовищ, яке дозволяє поєднати алгоритмічну, творчу, інформаційну та комунікативну діяльність, є Scratch. Візуальний принцип побудови програм із графічних блоків зменшує навантаження, пов'язане з синтаксисом текстових мов, і спрямовує увагу молодшого школяра на логіку дій, послідовність команд, причинно-наслідкові зв'язки та перевірку результату. Дитина може одразу спостерігати наслідок зміни алгоритму, експериментувати з параметрами, знаходити помилки та вдосконалювати власний проєкт.

Водночас сам факт використання Scratch ще не гарантує формування цифрової компетентності. Якщо діяльність обмежується відтворенням готового скрипту за зразком, учень набуває лише окремих операційних навичок. Компетентнісний результат виникає тоді, коли програмування включене в повний цикл розв'язування проблеми: усвідомлення мети, планування, добір інформаційних ресурсів, конструювання алгоритму, реалізацію, тестування, виправлення, пояснення та презентацію створеного продукту. Тому актуальним є визначення педагогічної моделі, у якій Scratch виступає не самоціллю, а інструментом формування цілісної інформаційно-цифрової компетентності молодшого школяра.

Аналіз останніх досліджень і публікацій. У працях українських науковців цифрова компетентність розглядається як інтегрована здатність до свідомого, доцільного й безпечного використання цифрових технологій у навчанні та професійній діяльності. У. Когут, О. Сікора та Т. Вдовичин акцентують на безперервному розвитку цифрової компетентності й побудові індивідуальної освітньої траєкторії [5]. О. Ляшенко, О. Спірін, С. Литвинова, О. Пінчук, О. Овчарук та А. Сухіх пов'язують цифровізацію освітнього середовища з формуванням цифрової компетентності, безпеки та відповідального використання технологій [6]. Т. Вдовичин, О. Сікора, Т. Кобильник і Н. Винницька розглядають цифрові технології як складову адаптивного освітнього середовища сучасної школи [7].

У контексті початкової освіти вагомими є дослідження готовності педагогів і майбутніх учителів до використання цифрових засобів. І. Хижняк, К. Власенко, І. Вікторенко та В. Величко обґрунтовують значення цифрових навчальних інструментів у професійній діяльності фахівців початкової освіти [8]. О. Саган, М. Гаран та О. Ліба визначають методико-інформатичну компетентність учителя початкових класів як важливу умову реалізації інформатичної освітньої галузі [9]. Л. Нежива, С. Паламар і Н. Семеній показують можливості цифрових інструментів у проєктній діяльності майбутніх педагогів [10].

Питання формування алгоритмічного мислення та навчання програмування в початковій школі представлено у працях українських дослідників. О. Шкуренко та А. Смик узагальнюють методичні особливості вивчення теми «Алгоритми і виконавці» та підкреслюють значення інтерактивних цифрових ресурсів [1]. О. Яценко і Л. Чумак обґрунтовують критерії добору середовищ навчання програмування та серед візуальних блокових середовищ виокремлюють Scratch як доцільний інструмент початкового навчання [2]. Т. Пушкарьова та О. Рибалко розглядають пропедевтику програмування у початковій школі та наголошують на поєднанні вивчення алгоритмів із практичною роботою у Scratch [3].

Нові українські публікації поглиблюють методичний аспект використання блокового програмування. І. Ліпчевська аналізує Scratch, Code.org, Blockly Games та інші блокові середовища з позицій розвитку алгоритмічного мислення й дослідницьких

умінь молодших школярів [4]. О. Кривонос, Н. Бірук, А. Торгонська та О. Яценко досліджують діагностику цифрової компетентності здобувачів освіти, що актуалізує питання чітких критеріїв оцінювання цифрових результатів [11]. І. Борисенко та співавтори аналізують упровадження інформаційно-комунікаційних технологій у зміст початкової освіти [12].

Важливими для обґрунтування інформаційно-цифрової складової є також праці українських авторів, присвячені цифровим інструментам і цифровізації освіти. Т. Качак характеризує цифрові інструменти в підготовці майбутніх учителів початкової школи [13], Р. Гуревич, Л. Коношевський і Н. Опушко аналізують проблеми й перспективи цифровізації освіти [14], а В. Биков, С. Литвинова та О. Мельник досліджують ефективність використання електронних освітніх ресурсів у початковій школі [15]. У попередній праці І. Василюк та Р. Романчука розглянуто особливості формування алгоритмічного мислення молодших школярів на уроках інформатики [16]. Дослідження Н. Бахмат та співавторів доповнює теоретичне поле сучасними підходами до розвитку цифрової компетентності педагогів [17].

Виділення невирішеної частини проблеми. У наукових публікаціях достатньо широко представлені окремі аспекти використання Scratch: розвиток алгоритмічного й обчислювального мислення, мотивації, творчості, навичок програмування. Проте менш розробленою залишається модель, яка безпосередньо пов'язує типи діяльності учня у Scratch з компонентами інформаційно-цифрової компетентності та дозволяє вчителю проєктувати не окремі ефектні вправи, а послідовність компетентнісно орієнтованих завдань. Також потребують уточнення критерії, за якими можна побачити, що учень не лише виконав програму, а навчився планувати цифрову діяльність, перевіряти інформаційний продукт, пояснювати свої рішення, знаходити помилки й відповідально працювати з цифровим контентом.

Мета статті. Метою статті є теоретичне обґрунтування дидактичного потенціалу програмування у Scratch та розроблення структурно-діяльнісної моделі формування інформаційно-цифрової компетентності молодших школярів. Для досягнення мети поставлено такі завдання: уточнити складники інформаційно-цифрової компетентності у контексті початкового програмування; визначити педагогічні функції Scratch; побудувати послідовність навчальної діяльності; розробити типологію компетентнісних завдань і критерії оцінювання цифрових проєктів учнів 2–4 класів.

Наукова новизна. Наукова новизна дослідження полягає у представленні Scratch не лише як середовища формування елементарних умінь програмування, а як інтегрованого засобу розвитку інформаційно-цифрової компетентності. Запропоновано п'ятикомпонентну структуру компетентності та цикл навчальної дії «зрозуміти – спланувати – алгоритмізувати – реалізувати – перевірити – удосконалити – представити», у якому кожний етап має спостережуваний учнівський продукт та критерії результативності.

Практичне значення. Запропоновану модель можна використовувати під час планування уроків інформатики у 2–4 класах, інтегрованих STEM-завдань, проєктів курсу «Я досліджую світ», факультативних і гурткових занять. Таблиці та критерії можуть бути основою для конструювання навчальних карток, чек-листів самооцінювання, рубрик оцінювання учнівських цифрових проєктів і методичних рекомендацій для вчителів початкових класів.

Методологія

Методи дослідження.

Дослідження має теоретико-проєктувальний характер. Компетентнісний підхід визначає кінцевий результат як здатність учня самостійно або з віково доцільною підтримкою розв'язувати навчальні завдання із застосуванням цифрових технологій. Діяльнісний підхід орієнтує на засвоєння понять через виконання реальних дій:

створення, перевірку, виправлення, пояснення та презентацію цифрового продукту. Конструктивістський підхід передбачає побудову знання самим учнем у процесі експериментування з блоками, спостереження результату та рефлексії. Проектний підхід забезпечує цілісність діяльності від задуму до завершеного продукту.

Методи дослідження включали аналіз типових освітніх програм початкової освіти України [18; 19], праць українських учених із цифрової компетентності та цифровізації освіти [5–8; 11; 14; 17], досліджень цифрової підготовки вчителя початкової школи [9; 10; 13], а також публікацій із методики алгоритмізації, початкового програмування та використання Scratch [1–4; 12; 15; 16]. Застосовано систематизацію для виокремлення компонентів компетентності, структурно-функціональний аналіз для встановлення зв'язку між компонентами й видами діяльності, а також педагогічне моделювання для побудови циклу навчальної роботи.

Процедура моделювання складалася з чотирьох кроків. Спочатку було зіставлено результати інформатичної освітньої галузі з типовими діями учня у Scratch. Далі ці дії згруповано за функціями: пошук і добір інформації, алгоритмізація, технічне виконання, створення цифрового продукту, оцінювання та безпека. На третьому кроці визначено послідовність навчальної діяльності, яка забезпечує перехід від виконання за зразком до самостійного проектування. На завершальному етапі сформульовано критерії, за якими вчитель може оцінити не лише працездатність програми, а й якість цифрової діяльності учня.

Результати

Інформаційно-цифрову компетентність молодшого школяра у межах цього дослідження розглядаємо як інтегровану здатність усвідомлено використовувати інформацію, цифрові пристрої та програмні середовища для розв'язування навчальних і творчих завдань, створювати й перевіряти цифрові продукти, пояснювати прийняті рішення та діяти безпечно і відповідально. У контексті Scratch ця компетентність виявляється не в кількості засвоєних блоків, а в якості дії учня з цифровим завданням.

На основі аналізу визначено п'ять взаємопов'язаних компонентів інформаційно-цифрової компетентності: інформаційно-пошуковий, алгоритмічний, технологічно-операційний, творчо-проектувальний та рефлексивно-безпековий. Їх зміст і прояви у Scratch подано в таблиці 1.

Таблиця 1. Компоненти інформаційно-цифрової компетентності та їх прояв у діяльності зі Scratch

Компонент	Зміст	Діяльність у Scratch	Спостережуваний результат
Інформаційно-пошуковий	Розуміння інформаційного завдання, добір і впорядкування даних	Вибір сюжету, фону, спрайтів, текстів; визначення потрібної інформації	Учень пояснює, які об'єкти й дані потрібні для реалізації задуму
Алгоритмічний	Планування послідовності дій, використання умов, повторень і подій	Побудова скриптів, циклів, розгалужень, взаємодії об'єктів	Учень прогнозує результат алгоритму та пояснює логіку його роботи
Технологічно-операційний	Практичне володіння цифровим середовищем	Створення спрайтів і сцен, налаштування властивостей, запуск і збереження	Учень самостійно виконує базові операції та обирає потрібні інструменти

		проєкту	
Творчо-проєктувальний	Перетворення задуму на цілісний цифровий продукт	Створення анімацій, історій, ігор, вікторин, моделей	Цифровий продукт відповідає задуму, має структуру й авторські рішення
Рефлексивно-безпековий	Перевірка, оцінювання, виправлення та відповідальна поведінка	Тестування, налагодження, самооцінювання, коректне використання чужих матеріалів	Учень знаходить помилки, обґрунтовує зміни та дотримується правил цифрової безпеки

Джерело: розроблено автором.

Інформаційно-пошуковий компонент формується тоді, коли робота починається не з механічного перетягування блоків, а з постановки проблеми. Наприклад, перед створенням інтерактивної історії про правила дорожнього руху учні визначають, які персонажі потрібні, які ситуації мають бути показані, яку інформацію необхідно перевірити, у якій послідовності розгортатиметься сюжет. Учитель може запропонувати список матеріалів із надлишковими елементами, щоб дитина аргументовано обрала лише потрібне. Таким чином відбувається перехід від роботи з інтерфейсом до інформаційного планування.

Алгоритмічний компонент є центральним у програмуванні, однак його формування доцільно пов'язувати з осмисленою задачею. Команди руху, цикли, умови та події мають виступати не окремими темами, а засобами реалізації задуму. Якщо учень створює модель світлофора, цикл потрібний для повторюваної зміни сигналів; якщо розробляє вікторину – умова використовується для перевірки відповіді; якщо створює анімаційну історію – події забезпечують взаємодію персонажів. У такому контексті програмна конструкція набуває функціонального значення.

Технологічно-операційний компонент охоплює володіння інструментами Scratch, але його не слід ототожнювати з компетентністю загалом. Операційна впевненість потрібна для того, щоб технічні труднощі не блокували задум. Тому на початкових етапах доцільні короткі тренувальні вправи: змінити фон, додати спрайт, налаштувати координати, записати звук, використати костюм, зберегти проєкт. Поступово ці дії інтегруються у складніші завдання, де учень сам обирає інструмент відповідно до мети.

Творчо-проєктувальний компонент перетворює програмування на продуктивну діяльність. Важливо, щоб завдання допускало варіативність: різні сюжети, персонажів, способи взаємодії, оформлення та рівні складності. Проєкт не обов'язково має бути великим. Для молодших школярів достатньо одного завершеного функціонального задуму: короткої анімації, діалогу, моделі явища, математичного тренажера або гри з простими правилами. Важливо, щоб учень міг пояснити, що саме створено, для кого і з якою метою.

Рефлексивно-безпековий компонент охоплює тестування, налагодження, оцінювання і відповідальне використання цифрового контенту. У Scratch помилка є педагогічно цінним ресурсом: учень має можливість швидко змінити блок, запустити програму повторно й одразу побачити результат. Учитель має формувати ставлення до налагодження не як до виправлення «невдачі», а як до нормальної частини цифрового проєктування. Паралельно необхідно звертати увагу на авторство, коректне використання зображень і звуків, нерозголошення особистих даних і правила безпечної взаємодії в мережових середовищах.

Ефективність формування компетентності залежить від логіки організації діяльності. Запропонований цикл включає сім взаємопов'язаних кроків і може бути використаний як структура окремого уроку або невеликого проєкту.

Таблиця 2. Цикл навчальної діяльності учня під час створення проєкту у Scratch

Етап	Ключове запитання	Дія учня	Результат
1. Зрозуміти завдання	Що потрібно створити і навіщо?	Формулює мету, визначає користувача та очікуваний результат	Короткий опис задуму
2. Спланувати	Які об'єкти й дані потрібні?	Добирає персонажів, сцени, інформацію, визначає основні кроки	План або ескіз
3. Алгоритмізувати	У якій послідовності все має відбуватися?	Будує словесний, графічний або блоковий алгоритм	Чернетка алгоритму
4. Реалізувати	Які блоки Scratch забезпечать потрібну дію?	Створює скрипти, налаштовує об'єкти й події	Робочий прототип
5. Перевірити	Чи працює проєкт так, як задумано?	Запускає, тестує різні ситуації, знаходить помилки	Перелік виявлених проблем
6. Удосконалити	Що потрібно змінити або спростити?	Виправляє помилки, покращує алгоритм і оформлення	Виправлена версія
7. Представити	Як пояснити власне рішення іншому?	Демонструє продукт, пояснює алгоритм, оцінює результат	Презентація та самооцінка

Джерело: авторська розробка.

Перший і другий етапи принципово важливі для компетентної спрямованості. Якщо учень одразу отримує готовий набір блоків, він не виконує інформаційного аналізу й планування. Тому навіть проста вправа має починатися з короткого проблемного запитання. Наприклад: «Як створити персонажа, який проведе екскурсію визначними місцями нашого міста?» або «Як зробити математичну гру, що перевіряє таблицю множення?». Таке формулювання задає функцію майбутнього цифрового продукту.

На етапі алгоритмізації доцільно поєднувати комп'ютерні та безкомп'ютерні форми. Перед створенням скрипту учні можуть розкласти картки з командами, зіграти роль виконавця, намалювати блок-схему або записати послідовність дій словами. Перехід між представленнями допомагає зрозуміти, що алгоритм існує незалежно від конкретного інтерфейсу Scratch. Це важливо для перенесення способу дії в інші цифрові середовища.

Під час реалізації вчитель має диференціювати підтримку. Частині учнів достатньо сформулювати мету, іншим потрібна картка з опорними блоками або частково готовий скрипт. Важливо, щоб допомога не перетворювала завдання на

копіювання. Корисним прийомом є «мінімальна підказка»: учень отримує лише той ресурс, який дозволяє подолати конкретну перешкоду, після чого продовжує самостійно.

Етап перевірки доцільно організовувати через тестові сценарії. Учень не просто натискає зелений прапорець, а перевіряє кілька ситуацій: що буде після повторного запуску, як поводить себе персонаж на межі сцени, чи правильно спрацьовує умова, чи завершується цикл, що відбувається при неправильній відповіді. Така робота наближує навчальну діяльність до реального процесу розроблення цифрового продукту.

Представлення результату перетворює цифровий продукт на засіб комунікації. Учень коротко пояснює задум, демонструє роботу програми, називає використані алгоритмічні конструкції, показує одну проблему, яку довелося виправити, і формулює, що хотів би покращити. Саме така рефлексія дозволяє оцінити глибину розуміння, яку неможливо побачити лише з готового проєкту.

Для практичного використання моделі важливо визначити типи навчальних завдань, які поступово збільшують самостійність молодшого школяра. Запропоновано п'ять рівнів – від відтворення до авторського проєктування.

Таблиця 3. Типологія завдань зі Scratch за рівнем самостійності учня

Рівень	Характер завдання	Приклад	Компетентнісний акцент
1. Відтворення	Виконання короткої інструкції з обов'язковим поясненням результату	Змусити спрайт пройти заданий маршрут	Операційна впевненість, зв'язок «команда – дія»
2. Доповнення	Зміна або завершення частково готового проєкту	Додати реакцію персонажа на дотик до об'єкта	Розуміння подій та умов
3. Налаштування	Пошук і виправлення помилки у скрипті	Виправити цикл, через який герой не зупиняється	Аналіз, тестування, причинно-наслідкові зв'язки
4. Модифікація	Перенесення відомого способу в нову ситуацію	Перетворити анімацію на навчальну вікторину	Перенесення, добір інструментів, творчість
5. Проєктування	Самостійне створення цифрового продукту за проблемним завданням	Створити інтерактивну модель або гру для однокласників	Планування, алгоритмізація, творчість, рефлексія

Джерело: авторська розробка.

Послідовне використання цих рівнів запобігає двом крайностям: надмірному інструктуванню та передчасній повній самостійності. На початку учень опановує базові операції у структурованій ситуації, але вже повинен пояснювати зв'язок між командою і результатом. Далі зростає частка вибору: спочатку вибір окремого блока або параметра, потім способу виправлення, далі – структури проєкту та власного задуму.

Особливе місце має завдання на налаштування. Воно формує здатність аналізувати готовий цифровий продукт, локалізувати проблему та перевіряти гіпотезу.

Наприклад, учням пропонується скрипт, у якому персонаж має рухатися до краю сцени й повертатися, але через помилкову кількість повторень зупиняється раніше. Діти прогнозують, який блок може бути причиною, змінюють один параметр і перевіряють результат. Такі завдання формують культуру доказового виправлення, а не випадкового «перетягування блоків».

Проекти у Scratch доцільно інтегрувати з іншими освітніми галузями. У математичній галузі це можуть бути тренажери обчислень, моделі геометричних побудов або анімації задач. У мовно-літературній – інтерактивні діалоги, продовження казки, озвучені історії. У природничій – моделі кругообігу води, зміни пір року, руху планет, життєвих циклів рослин. У громадянській та історичній – інтерактивні екскурсії й вікторини. Така інтеграція посилює інформаційно-пошуковий компонент, оскільки учень має працювати зі змістом предмета, а не лише з програмними конструкціями.

У початковій школі доцільно оцінювати не технічну складність проекту, а відповідність задуму, логічність алгоритму, самостійність, перевірку результату та здатність пояснити прийняті рішення. Висока кількість блоків не повинна автоматично означати високий рівень. Часто короткий, чітко структурований скрипт є кращим свідченням розуміння, ніж великий проект, створений шляхом копіювання.

Таблиця 4. Критерії оцінювання цифрового проекту молодшого школяра у Scratch

Критерій	Початковий рівень	Достатній рівень	Високий рівень
Відповідність задуму	Продукт частково відповідає завданню, мета пояснюється нечітко	Основні функції відповідають меті	Усі елементи підпорядковані чіткому задуму, учень аргументує рішення
Алгоритмічна коректність	Скрипт містить помилки або працює лише в одному випадку	Алгоритм працює коректно у типових ситуаціях	Учень прогнозує різні ситуації, оптимізує або пояснює структуру алгоритму
Самостійність і добір інструментів	Переважає робота за детальною інструкцією	Учень самостійно виконує основні дії, інколи потребує підказки	Самостійно обирає засоби Scratch та пояснює їх доцільність
Тестування і налагодження	Перевірка епізодична, помилки виправляються переважно з допомогою	Учень знаходить і виправляє типові помилки	Системно тестує проєкт, пояснює причину помилки та спосіб виправлення
Цифрова відповідальність	Потребує нагадування про безпечне використання даних і ресурсів	Дотримується основних правил безпеки та коректного використання матеріалів	Усвідомлено пояснює правила, зазначає походження матеріалів і захищає особисті дані
Презентація і рефлексія	Показує продукт без пояснення процесу	Пояснює задум і основні кроки	Аналізує сильні сторони, труднощі, зміни та можливості

		вдосконалення
--	--	---------------

Джерело: розроблено автором.

Обговорення

Запропонована модель демонструє, що формування інформаційно-цифрової компетентності засобами Scratch потребує зміни акценту з «вивчення блоків» на «розв'язування задачі цифровими засобами». Програмні конструкції залишаються важливими, але набувають педагогічного сенсу лише тоді, коли учень використовує їх для досягнення зрозумілої мети. Такий підхід узгоджується з українськими дослідженнями методики вивчення алгоритмів і програмування в початковій школі [1–4; 16].

Українські дослідники підкреслюють, що результативність цифрового навчання залежить не від самого факту використання технології, а від педагогічного сценарію, змісту завдання, рівня самостійності учня та організації проєктної діяльності [8–10; 15]. Запропонована модель конкретизує цю позицію через спостережувані продукти кожного етапу: план, алгоритм, прототип, перелік помилок, виправлену версію і коротку рефлексію. Завдяки цьому вчитель може бачити не тільки фінальний результат, але й хід мислення дитини.

Водночас інформаційно-цифрова компетентність не може бути зведена лише до засвоєння циклів, умов або інших програмних конструкцій. Українські науковці пов'язують її також із роботою з інформацією, створенням цифрового контенту, безпечним і відповідальним використанням технологій, комунікацією та здатністю розв'язувати навчальні проблеми [5–7; 11; 14; 17]. Тому в запропонованій структурі алгоритмічний компонент є центральним, але взаємодіє з інформаційно-пошуковим, технологічно-операційним, творчо-проектувальним і рефлексивно-безпековим компонентами.

Практично важливим є дозоване педагогічне підтримування учня під час переходу від готового зразка до самостійного програмування. Методичні напрацювання О. Шкуренко й А. Смик, Т. Пушкарьової й О. Рибалко та І. Ліпчевської засвідчують доцільність поступового ускладнення алгоритмічних завдань, поєднання пояснення, практичної роботи та цифрових середовищ [1; 3; 4]. Тому підказки, частково готові фрагменти, запитання для самоперевірки й взаємооцінювання доцільно використовувати як тимчасову опору, поступово зменшуючи її в міру зростання самостійності учня.

Вікові особливості молодших школярів вимагають обмеження розміру проєкту й чіткого бачення результату. Проєкт на один урок може містити один новий алгоритмічний елемент, але завершуватися функціональним продуктом. Наприклад, замість абстрактного опрацювання циклу учні створюють анімацію «Кіт робить зарядку», де одна послідовність рухів повторюється декілька разів. У наступному завданні вони змінюють кількість повторень і прогнозують результат. Так поєднуються наочність, дія та алгоритмічне узагальнення.

Організація парної роботи також має компетентнісний потенціал. Один учень може виконувати роль «програміста», другий – «тестувальника», після чого ролі змінюються. Тестувальник не просто вказує, що «не працює», а формулює, за якої дії виникла проблема. Така модель розвиває мовлення, аргументацію, взаємоперевірку і культуру конструктивного зворотного зв'язку. Водночас вона дозволяє учням побачити, що створення цифрового продукту є соціальною, а не лише індивідуальною діяльністю.

Окремої уваги потребує цифрова безпека. Навіть якщо Scratch використовується офлайн або в закритому шкільному середовищі, завдання можуть формувати базові звички: не вказувати персональні дані у відкритому проєкті, використовувати коректні

назви файлів, поважати авторство чужих зображень, звуків і програмних фрагментів, не публікувати матеріали без погодження. Такі правила доцільно включати безпосередньо в рубрику оцінювання, щоб безпека сприймалася як частина якісної цифрової роботи.

Запропонована модель має теоретико-проектувальний статус і не подається як емпірично доведена технологія. Її подальша перевірка потребує педагогічного експерименту з визначенням початкового та підсумкового рівнів інформаційно-цифрової компетентності, аналізом учнівських проєктів і спостереженням за самостійністю дітей під час планування, програмування, тестування та презентації результатів. Перспективним є також порівняння різних форматів підтримки – покрокових інструкцій, карток-підказок, парного програмування та відкритих проєктних завдань.

Висновки

Програмування у Scratch може бути ефективним засобом формування інформаційно-цифрової компетентності молодших школярів за умови, що навчання не обмежується відтворенням готових скриптів. Компетентнісний потенціал середовища реалізується через включення учня у повний цикл цифрової діяльності: розуміння проблеми, інформаційне планування, алгоритмізацію, програмну реалізацію, тестування, налагодження, презентацію та рефлексію.

У структурі інформаційно-цифрової компетентності доцільно виокремлювати п'ять взаємопов'язаних компонентів: інформаційно-пошуковий, алгоритмічний, технологічно-операційний, творчо-проектувальний та рефлексивно-безпековий. Scratch створює можливості для одночасного розвитку цих компонентів, оскільки поєднує роботу з інформаційними об'єктами, алгоритмами, мультимедійним контентом, творчим проєктуванням і перевіркою результату.

Розроблена послідовність «зрозуміти завдання – спланувати – створити алгоритм – реалізувати у Scratch – перевірити – удосконалити – представити результат» може бути використана як універсальна схема організації уроку або короткого проєкту. Запропонована типологія завдань забезпечує поступовий перехід від виконання за зразком до самостійного цифрового проєктування, а критерії оцінювання дозволяють враховувати не лише технічну працездатність програми, а й логіку, самостійність, тестування, цифрову відповідальність і здатність учня пояснити власне рішення.

Практичне значення моделі полягає у можливості її застосування на уроках інформатики, у міжпредметних цифрових проєктах, інтегрованому навчанні та позакласній роботі. Подальші дослідження доцільно спрямувати на емпіричну перевірку запропонованої моделі, визначення вікової динаміки її компонентів та розроблення діагностичного інструментарію для оцінювання інформаційно-цифрової компетентності учнів 2–4 класів.

Список використаних джерел

1. Шкуренко О. В., Смик А. О. Методичні особливості вивчення теми «Алгоритми і виконавці» у початковій школі. Молодий вчений. 2024. № 7 (131). DOI: 10.32839/2304-5809/2024-7-131-3.
2. Яценко О. І., Чумак Л. М. Критерії добору середовища навчання програмування для формування ІКТ-компетентності майбутніх учителів початкової школи. Інформаційні технології і засоби навчання. 2020. Т. 78, № 4. С. 219–236. DOI: 10.33407/itlt.v78i4.2912.
3. Пушкарьова Т. О., Рибалко О. О. Пропедевтика вивчення програмування у початковій школі за підручниками інформатики освітньої технології «Росток». Проблеми сучасного підручника. 2021. Вип. 27. С. 223–232. DOI: 10.32405/2411-1309-2021-27-223-232.

4. Ліпчевська І. Л. Методичні аспекти використання блокових середовищ програмування в навчанні інформатики у початковій школі. Актуальні проблеми та стратегії розвитку початкової освіти : матеріали VI Всеукраїнської студентської науково-практичної конференції. Рівне : РДГУ, 2026. С. 13–15.
5. Когут У. П., Сікора О. В., Вдовичин Т. Я. Формування індивідуальної освітньої траєкторії вчителя з розвитку цифрової компетентності. Інформаційні технології і засоби навчання. 2022. Т. 91, № 5. С. 186–204. DOI: 10.33407/itlt.v91i5.5006.
6. Ляшенко О. І., Спірін О. М., Литвинова С. Г., Пінчук О. П., Овчарук О. В., Сухіх А. С. Концептуальні засади цифровізації освітнього середовища закладів загальної середньої освіти. Інформаційні технології і засоби навчання. 2024. Т. 102, № 4. С. 1–25. DOI: 10.33407/itlt.v102i4.5829.
7. Вдовичин Т. Я., Сікора О. В., Кобильник Т. П., Винницька Н. В. Формування адаптивного цифрового середовища в закладах загальної середньої освіти. Інформаційні технології і засоби навчання. 2024. Т. 103, № 5. С. 55–77. DOI: 10.33407/itlt.v103i5.5656.
8. Хижняк І. А., Власенко К. В., Вікторенко І. Л., Величко В. Є. Готовність майбутніх фахівців початкової освіти до використання цифрових засобів навчання у професійній діяльності. Інформаційні технології і засоби навчання. 2021. Т. 86, № 6. С. 106–122. DOI: 10.33407/itlt.v86i6.3666.
9. Саган О. В., Гаран М. С., Ліба О. М. Формування методико-інформатичної компетентності вчителів початкових класів. Інформаційні технології і засоби навчання. 2018. Т. 65, № 3. С. 304–315. DOI: 10.33407/itlt.v65i3.1942.
10. Нежива Л. Л., Паламар С. П., Семеній Н. В. Використання цифрових інструментів у проєктній діяльності студентів педагогічних спеціальностей закладів вищої освіти. Інформаційні технології і засоби навчання. 2024. Т. 101, № 3. С. 71–85. DOI: 10.33407/itlt.v101i3.5398.
11. Кривонос О. М., Бірук Н. П., Торгонська А. О., Яценко О. І. Діагностика сформованості цифрової компетентності старшокласників. Інформаційні технології і засоби навчання. 2023. Т. 97, № 5. С. 94–124. DOI: 10.33407/itlt.v97i5.5456.
12. Борисенко І. В., Биконя О. П., Рембач О. О., Шумна Л. П., Олійник О. І., Аніщенко В. О. Досвід упровадження ІКТ у зміст початкової освіти в Україні та Великій Британії. Інформаційні технології і засоби навчання. 2020. Т. 75, № 1.
13. Качак Т. Б. Цифрові інструменти літературної освіти майбутніх учителів початкової школи в умовах дистанційного навчання. Інформаційні технології і засоби навчання. 2021. Т. 86, № 6. С. 144–169. DOI: 10.33407/itlt.v86i6.4079.
14. Гуревич Р. С., Коношевський Л. Л., Опушко Н. Р. Цифровізація освіти в сучасному суспільстві: проблеми, досвід, перспективи. Освітологічний дискурс. 2022. № 3–4 (38–39). С. 22–46. DOI: 10.28925/2312-5829.2022.342.
15. Биков В. Ю., Литвинова С. Г., Мельник О. М. Ефективність навчання з використанням електронних освітніх ігрових ресурсів у початковій школі. Інформаційні технології і засоби навчання. 2017. Т. 62, № 6.
16. Василиків І. Б., Романчук Р. М. Особливості формування алгоритмічного мислення молодших школярів на уроках інформатики. Молодь і ринок. 2021. № 2 (188). С. 90–95. DOI: 10.24919/2308-4634.2021.230493.
17. Бахмат Н. В., Сторчова Т. В., Моцик Р. В., Мелекесцева Н. В., Братиця Г. Г. Сучасні тенденції розвитку цифрової компетентності майбутніх учителів: європейський досвід. Академічні візії. 2023. Вип. 15. DOI: 10.5281/zenodo.7593637.
18. Типова освітня програма для 1–2 класів закладів загальної середньої освіти, розроблена під керівництвом О. Я. Савченко : затверджена наказом Міністерства освіти і науки України. 2022. URL: <https://mon.gov.ua/osvita-2/zagalna-serednya-osvita/osvitni-programi/navchalni-programi-dlya-1-4-klasiv>.

19. Типова освітня програма для 3–4 класів закладів загальної середньої освіти, розроблена під керівництвом Р. Б. Шияна : затверджена наказом Міністерства освіти і науки України. 2022. URL: <https://mon.gov.ua/osvita-2/zagalna-serednya-osvita/osvitni-programi/navchalni-programi-dlya-1-4-klasiv>.

References

Shkurenko, O. V., & Smyk, A. O. (2024). Metodichni osoblyvosti vyvchennia temy «Alhorytmy i vykonavtsi» u pochatkovii shkoli [Methodological features of studying “Algorithms and executors” in primary school]. *Molodyi vchenyi*, 7(131). <https://doi.org/10.32839/2304-5809/2024-7-131-3>

Yatsenko, O. I., & Chumak, L. M. (2020). Kryterii doboru seredovyscha navchannia prohramuvannia dlia formuvannia IKT-kompetentnosti maibutnikh uchyteliv pochatkovoi shkoly [Criteria for selecting a programming learning environment]. *Information Technologies and Learning Tools*, 78(4), 219–236. <https://doi.org/10.33407/itlt.v78i4.2912>

Pushkarova, T. O., & Rybalko, O. O. (2021). Propedevtyka vyvchennia prohramuvannia u pochatkovii shkoli za pidruchnykamy informatyky osvitnoi tekhnolohii «Rostok» [Propaedeutics of programming in primary school]. *Problemy suchasnoho pidruchnyka*, 27, 223–232. <https://doi.org/10.32405/2411-1309-2021-27-223-232>

Lipchevska, I. L. (2026). Metodichni aspekty vykorystannia blokovykh seredovysch prohramuvannia v navchanni informatyky u pochatkovii shkoli [Methodological aspects of using block-based programming environments in primary school computer science education]. *Aktualni problemy ta stratehii rozvytku pochatkovoi osvity*, 13–15.

Kohut, U. P., Sikora, O. V., & Vdovychyn, T. Ya. (2022). Formuvannia indyvidualnoi osvitnoi traiektorii vchytelia z rozvytku tsyfrovoy kompetentnosti [Formation of the teacher’s individual educational trajectory for digital competence development]. *Information Technologies and Learning Tools*, 91(5), 186–204. <https://doi.org/10.33407/itlt.v91i5.5006>

Liashenko, O. I., Spirin, O. M., Lytvynova, S. H., Pinchuk, O. P., Ovcharuk, O. V., & Sukhikh, A. S. (2024). Kontseptualni zasady tsyfrovizatsii osvitnoho seredovyscha zakladiv zahalnoi serednoi osvity [Conceptual fundamentals for digitalisation of the educational environment in general secondary education institutions]. *Information Technologies and Learning Tools*, 102(4), 1–25. <https://doi.org/10.33407/itlt.v102i4.5829>

Vdovychyn, T. Ya., Sikora, O. V., Kobylnyk, T. P., & Vynnytska, N. V. (2024). Formuvannia adaptivnoho tsyfrovoho seredovyscha v zakladakh zahalnoi serednoi osvity [Formation of an adaptive digital environment in general secondary education institutions]. *Information Technologies and Learning Tools*, 103(5), 55–77. <https://doi.org/10.33407/itlt.v103i5.5656>

Khyzhniak, I. A., Vlasenko, K. V., Viktorenko, I. L., & Velychko, V. Ye. (2021). Hotovnist maibutnikh fakhivtsiv pochatkovoi osvity do vykorystannia tsyfrovyykh zasobiv navchannia u profesiinii diialnosti [Readiness of future primary education specialists to use digital learning tools]. *Information Technologies and Learning Tools*, 86(6), 106–122. <https://doi.org/10.33407/itlt.v86i6.3666>

Sagan, O. V., Haran, M. S., & Liba, O. M. (2018). Formuvannia metodyko-informatychnoi kompetentnosti vchyteliv pochatkovyykh klasiv [Formation of the methodical-informatic competence of primary school teachers]. *Information Technologies and Learning Tools*, 65(3), 304–315. <https://doi.org/10.33407/itlt.v65i3.1942>

Nezhyva, L. L., Palamar, S. P., & Semenii, N. V. (2024). Vykorystannia tsyfrovyykh instrumentiv u proiektnii diialnosti studentiv pedahohichnykh spetsialnostei zakladiv vyshchoi osvity [The use of digital tools in project activities]. *Information Technologies and Learning Tools*, 101(3), 71–85. <https://doi.org/10.33407/itlt.v101i3.5398>

Kryvonos, O. M., Biruk, N. P., Torgonska, A. O., & Yatsenko, O. I. (2023). Diahnostyka sformovanosti tsyfrovoy kompetentnosti starshoklasnykiv [Diagnostics of high school students’ digital competence]. *Information Technologies and Learning Tools*, 97(5), 94–124. <https://doi.org/10.33407/itlt.v97i5.5456>

Borysenko, I. V., Bykonja, O. P., Rembach, O. O., Shumna, L. P., Oliinyk, O. I., & Anishchenko, V. O. (2020). Dosvid uprovadzhennia IKT u zmist pochatkovoї osvity v Ukraini ta Velykii Brytanii [Experience of ICT implementation in primary curriculum in Ukraine and the United Kingdom]. *Information Technologies and Learning Tools*, 75(1).

Kachak, T. B. (2021). Tsyfrovi instrumenty literaturnoi osvity maibutnikh uchyteliv pochatkovoї shkoly v umovakh dystantsiinoho navchannia [Digital tools of literary education of future primary school teachers in distance learning]. *Information Technologies and Learning Tools*, 86(6), 144–169. <https://doi.org/10.33407/itlt.v86i6.4079>

Hurevych, R. S., Konoshevskiy, L. L., & Opushko, N. R. (2022). Tsyfrovizatsiia osvity v suchasnomu suspilstvi: problemy, dosvid, perspektyvy [Digitalization of education in modern society: problems, experience, prospects]. *Osvitohichnyi dyskurs*, 3–4(38–39), 22–46. <https://doi.org/10.28925/2312-5829.2022.342>

Bykov, V. Yu., Lytvynova, S. H., & Melnyk, O. M. (2017). Efektyvnist navchannia z vykorystanniam elektronnykh osvitnikh ihrovykh resursiv u pochatkovii shkoli [Effectiveness of learning with electronic educational game resources in primary school]. *Information Technologies and Learning Tools*, 62(6).

Vasylykiv, I. B., & Romanchuk, R. M. (2021). Osoblyvosti formuvannia alhorytmichnoho myslennia molodshykh shkoliariv na urokakh informatyky [Features of developing primary school pupils' algorithmic thinking in informatics lessons]. *Molod i rynok*, 2(188), 90–95. <https://doi.org/10.24919/2308-4634.2021.230493>

Bakhmat, N. V., Storchova, T. V., Motsyk, R. V., Mieliekiesceva, N. V., & Bratytsia, H. H. (2023). Suchasni tendentsii rozvytku tsyfrovoi kompetentnosti maibutnikh uchyteliv: yevropeyskyi dosvid [Current trends in the development of future teachers' digital competence]. *Akademichni vizii*, 15. <https://doi.org/10.5281/zenodo.7593637>

Ministry of Education and Science of Ukraine. (2022). *Typova osvitnia prohrama dlia 1–2 klasiv zakladiv zahalnoi serednoi osvity, rozroblena pid kerivnytstvom O. Ya. Savchenko* [Typical educational programme for grades 1–2].

Ministry of Education and Science of Ukraine. (2022). *Typova osvitnia prohrama dlia 3–4 klasiv zakladiv zahalnoi serednoi osvity, rozroblena pid kerivnytstvom R. B. Shyiana* [Typical educational programme for grades 3–4].