

УДК 371.315

Дудка О.М. к.п.н., Власій О.О., к.т.н.

ДВНЗ «Прикарпатський національний університет імені Василя Стефаника»

## ОСОБЛИВОСТІ ВИВЧЕННЯ ПРОГРАМУВАННЯ НА SCRATCH

**Дудка О.М., Власій О.О. Особливості вивчення програмування на Scratch.** В статті розглянуто особливості процесу вивчення програмування як навчальної дисципліни; проведено аналіз можливостей візуальної мови програмування Scratch з точки зору сучасних парадигм програмування; запропоновано розробку ігрових Scratch-проектів як мотиваційний засіб при вивченні програмування; проаналізовано наявні дидактичні матеріали для вивчення Scratch; запропоновано використання навчального сайту для зацікавленості програмуванням в Scratch.

**Ключові слова:** мова програмування, Scratch, комп'ютерна гра, Scratch-проект, навчальний сайт

**Дудка О.М., Власій О.О. Особенности изучения программирования на Scratch.** В статье рассмотрены особенности процесса изучения программирования как учебной дисциплины; приведен анализ возможностей визуального языка программирования Scratch с точки зрения современных парадигм программирования; предложена разработка игровых Scratch-проектов как мотивационное средство при изучении программирования; проанализированы имеющиеся дидактические материалы для изучения Scratch; предложено использование учебного сайта для заинтересованности программированием в Scratch.

**Ключові слова:** язык программирования, Scratch, компьютерная игра, Scratch-проект, навчальний сайт

**O. Dudka, O. Vlasii Improving introductory programming with Scratch.** The characteristic of learning programming in secondary school is given. The capabilities of visual programming language Scratch in terms of modern programming paradigms are analyzed. The development of game Scratch-projects as a motivational tool in the study of programming is proposed. Existing teaching resources for learning Scratch are analyzed. Training site about Scratch as interest-based portals is proposed.

**Ключові слова:** programming language, Scratch, computer game, teaching site

**Постановка проблеми.** Вивчення програмування в наш час стало одним із пріоритетних напрямків підготовки майбутніх професіоналів в ІТ-галузі. Цікаво зауважити, що з'явилася значна кількість інтернет-ресурсів, які пропонують вивчення різних мов програмування за різною методикою для різних вікових категорій. Деякі ресурси пропонують вивчати ази програмування, а вірніше ази алгоритмізації, користувачам «які ще не вміють читати» (<https://code.org/>).

Процес вивчення програмування у середній школі є актуальною проблемою сьогодення. Внаслідок великого розмаїття навчальних програм, технічних можливостей шкіл, рівнів підготовки педагогів на даний момент не існує єдиного підходу до узгодженого централізованого вивчення конкретної мови програмування. Думки методистів, ІТ-експертів та програмістів щодо вибору мови програмування, яку необхідно вивчати у середній школі, кардинально розходяться. Одні відстоюють класичну версію навчальної мови Pascal, інші – візуальну реалізацію Lazarus, ще інші – сучасні C++ та Python, десь між пропозицій, звісно, є і HTML [1, 2]. Хочеться зауважити, що про підготовку до класичного програмування, так би мовити, початковий рівень програмування, таких дискусій не ведеться. Тут першість займає, безумовно, Scratch – візуальна мова програмування. Scratch можна розглядати в певній мірі як прототип Flash, хоча Scratch появився значно пізніше. Можливо, саме здатність поєднання графіки, медіа та програмування зі зручним користувацьким інтерфейсом і підштовхнуло до створення чогось аналогічного, але простішого і доступнішого, саме для дітей. Скептики, які вважали Scratch іграшкою, недостойною уваги, все ж змушені були визнати, що за допомогою саме цієї мови і цього середовища дітям набагато простіше даються базові уявлення про програмування. Йде мова саме про розвиток алгоритмічного мислення, яке є необхідним фундаментом будь-якого процесу розробки програмного забезпечення. Scratch дав у руки потужний інструмент створення мультимедійних інтерактивних проектів на основі реалізації складених алгоритмів у вигляді скриптів (так називають програми на мові Scratch). Тому метою дослідження є аналіз можливостей використання Scratch як початкового етапу у вивченні програмування та створення ефективної інформативної системи для надання можливості учням додатково вивчати Scratch, а завданням – популяризація програмування серед учнів, поєднання навчання з ігровими ситуаціями, мотивація школярів до розробки ігрових проектів.

**Аналіз можливостей Scratch.** Scratch створено під керівництвом професора Митчела Резника (Mitchel Resnick) у дослідницькій групі Lifelong Kindergarten research group при Массачусетському технологічному інституті. Попередня відома розробка цієї групи — це

популярна серія LEGO Mindstorms, що є комплектом деталей для складання програмованих роботів. Робота велася у співдружності з дослідниками Університету Лос-Анджелеса (UCLA — University of California, Los Angeles) при фінансовій підтримці Національного наукового фонду і Фонду корпорації Intel. Scratch – вільне програмне забезпечення (ліцензія GNU GPL), до переваг якого варто виділити кросплатформенність, наочність запису алгоритму, багатомовність інтерфейсу (кілька десятків мов) [3, 4]. Scratch знайомить з базовими поняттями алгоритмізації та програмування: типи даних (символьні, числові, логічні, графічні, аудіо), змінні (глобальні й локальні), операції (арифметичні, логічні), вирази (числові, текстові, логічні, порівняння), оператори, функції, алгоритмічні структури (лінійні, розгалужені, циклічні), введення й виведення даних. Scratch можна вільно використовувати у шкільній чи позашкільній освіті, скачавши середовище програмування з офіційного сайту розробників <http://scratch.mit.edu/>.

**Ігрові проекти як мотиваційний засіб вивчення програмування.** Комп'ютерні ігри – з недавніх пір це словосполучення міцно увійшло в наше життя: кожен, хто має комп'ютер, напевно, зміг відчути їх привабливість, мабуть, гра закладена в саму природу людини. Комп'ютерні ігри відносяться до числа актуальних проблем сучасної науки. Ігри допомагають нам змоделювати різні життєві ситуації, проблеми та надають деякі можливі шляхи їх вирішення. Гра містить в собі всі необхідні передумови для природного розвитку особистості і культури суспільства. З вдосконаленням комп'ютерів удосконалювалися й ігри, залучаючи все більше і більше людей. На сьогоднішній день комп'ютерна техніка досягла такого рівня розвитку, що дозволяє програмістам розробляти дуже реалістичні ігри з хорошим графічним і звуковим оформленням. З 2011 року комп'ютерні ігри офіційно визнані в США окремим видом мистецтва. Комп'ютерні ігри зробили настільки істотний вплив на суспільство, що в інформаційних технологіях відзначена стійка тенденція до «гейміфікації» для неігрового прикладного програмного забезпечення. Думка, що ігри роблять лише для дітей, не є вірною. 75% цільової аудиторії ігрового бізнесу – це люди, старші за 35 років. Розробка ігор – це точка перетину мистецтва та високих технологій. В компаніях, що працюють в ігровому бізнесі, лише 20-25% роботи виконують програмісти. Все інше – це праця людей нетехнічної спеціалізації, таких як контент-менеджери, маркетологи, художники чи дизайнери, і їхній внесок стає все вагомим [5, 6].

Багато розробників розглядають індустрію комп'ютерних ігор лише як сектор економіки, в якому можна непогано заробити. Але, насправді, створення ігор – це ще й інструмент культури. Не секрет, що для сучасних дітей часто ігри замінюють дитячі казки, які формують світогляд, для підлітків вони створюють персонажів-кумирів, замінюючи собою класичні книги, і лише для дорослих людей ігри служать здебільшого тільки розвагою. При правильному підході ідеї, закладені в комп'ютерні ігри, дуже добре засвоюються дітьми. Сучасні діти пізнають світ багато в чому через призму комп'ютерних ігор. Ігри ці створюються людьми, часто далекими від культури, педагогіки, психології. А в деяких випадках ігри навмисно створюються з метою пропаганди тих чи інших далеко не позитивних ідей. Культурний потенціал комп'ютерних ігор ще тільки починає розкриватися, але вже зараз зрозуміло, що потенціал цей – величезний [7].

Тому аж ніяк не можна заперечувати значущість комп'ютерних ігор, а навпаки – зважаючи на їх прихований потенціал, використовувати комп'ютерні ігри з метою виховання, навчання та розвитку. На нашу думку, можливості Scratch кардинально змінюють мотивацію вивчення програмування. Якщо при вивченні алгоритмів (наприклад, Logo) в основі мотивації лежить потреба складання алгоритму для виконання головним героєм певного набору дій (хай це Черепашка, Кенгуру чи Садівник), то за можливості реалізації в Scratch мультимедійного проекту за власним сценарієм (від розробки головного героя до програмування його довільних дій) на чільне місце виходить бажання реалізувати власні фантазії, що, безумовно, є потужним інструментом мотивації. Цікаво зауважити, що при цьому завдання формулюються як завдання гри, а не як завдання з вивчення алгоритмів. Потреба володіння знаннями про ті чи інші команди та алгоритми виникає з природного бажання реалізувати задумане і є невидимим потужним інструментом мотивації.

**Розробка гри на прикладі Scratch-проекту.** Візуальна мова програмування Scratch, хоча розглядається як мова програмування для дітей та підлітків, має широкі можливості для реалізації прототипів повноцінних ігрових програм [8]. Scratch дає можливість легко поєднувати графіку, зображення, звук, а саме такі елементи поєднує в собі будь-яка комп'ютерна гра. Процес розробки ігрового проекту в середовищі Scratch включає найважливіші характерні етапи створення

комп'ютерної гри: від ідеї та пошуку жанру до публікації та документування. Наведемо лише кілька аналогій: вмонтований графічний редактор дає можливість розробляти графічні образи героїв та створювати дизайн сцен; в свою чергу, розробка образів спрайтів ознайомлює із основами класичної анімації; послідовне виконання скриптів, організоване на основі передачі повідомлень, дає поняття про механізми передачі параметрів; паралельне виконання скриптів дає уявлення про паралельні процеси; можливість використання датчиків підводить до розуміння інтерактивності; можливість прив'язки змінних та скриптів до спрайтів – до поняття об'єктно-орієнтованого програмування. Як вже зауважувалося вище, Scratch є відмінним інструментом для створення комп'ютерних ігор.

Розглянемо етапи розробки типової комп'ютерної гри:

1. Ідея.
2. Сценарій: герої, їх ролі, правила, за якими вони взаємодіють, і головне - мета гри!
3. Створення чи пошук необхідних зображень та звуків.
4. Написання скриптів для спрайтів та сцен: реалізація самостійних дій спрайтів (паралельні скрипти), реалізація взаємодії спрайтів (послідовні скрипти), якщо гра багатоскладова чи багаторівнева – забезпечення переходу між її складовими чи рівнями.
5. Написання правил та довідок для користувача.
6. Тестування проекту.
7. виправлення помилок.
8. Створення опису гри.
9. Публікація проекту.

Розглянемо ілюстративний приклад процесу розробки комп'ютерної гри на всіх його етапах:

- Ідея: є предмети двох типів, необхідно посортувати, який до якого типу відноситься.
- Можливий сценарій: Рудий Кіт пропонує гравцеві посортувати розкидані по екрану об'єкти.
- Варіанти спрайтів: головний герой (наприклад, Рудий кіт).
- Об'єкти двох типів (наприклад, квадрати і кружки, овочі і фрукти, голосні і приголосні букви, птахи і тварини - кому що до вподоби).
- Контейнери для сортування (наприклад, ящики, кошики, хатинки, книжки і т.п.).
- Вибір ролей (наприклад Рудий кіт – пропонує гравцеві посортувати об'єкти, може допомагати підказками. Об'єкти – хаотично розкидані по екрану; можуть перетягуватися мишкою по екрану; зникають, коли доторкаються до потрібного контейнера. Контейнери – мають визначене місце; видають різні звуки в залежності, чи до них доторкається правильний чи неправильний об'єкт).

Звісно, ролі можна вдосконалювати, ускладнювати сценарій додатковими можливостями, але порада: спочатку краще спробувати реалізувати мінімальний "кістяк" проекту, а потім розробляти додаткові можливості. Діємо за правилом: не все одразу!

Зауважимо, що для зображення взаємозв'язку та взаємодії об'єктів можна використовувати діаграми, які дозволяють наочно зобразити властивості та взаємозв'язки об'єктів і які можна вважати сходинкою до освоєння в майбутньому діаграм потоків та подій (див. рис. 1).

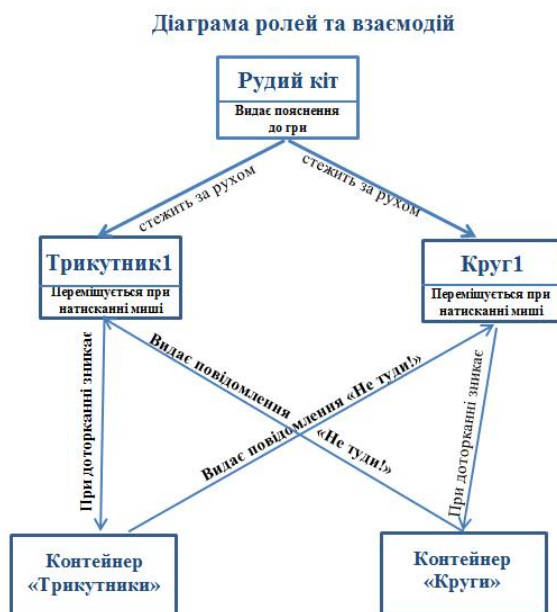


Рис.1 Діаграма ролей та взаємодій об'єктів у ігровому Scratch-проекті

**Навчальний сайт як портал зацікавленості Scratch.** Навчальні програми ХХ рр.. передбачали вивчення Scratch у 5-7 класах, діючі навчальні програми переносять вивчення Scratch у початкову школу. Однак практика показує, що бажання дітей реалізувати свої ідеї на Scratch значно випереджають програму, а найчастіше виходять за її рамки. В першу чергу йде мова про реалізацію комп'ютерних ігор. Не секрет, що залежність від комп'ютерних ігор стала однією із серйозних проблем сучасності. То чому ж таку залежність не використати в позитивному плані? Коли дитина бачить, що вона не просто може грати, а може сама розробити гру і дати завдання своїм друзям, знайомим та навіть і вчителям, то з геймера вона перетворюється на розробника програмного забезпечення, який має практичний досвід роботи з ігровими програмами і при цьому хоче втілити особисті ідеї на практиці.

Повноцінним інструментом для розробки комп'ютерної гри, звичайно, є Scratch. Тому актуальною проблемою є створення ефективної інформативної системи для надання можливості учням додатково вивчати візуальну мову програмування Scratch. Друкованих посібників зі Scratch практично немає (окрім, звісно, підручників, в яких вивчення Scratch по краплинах вивчається в школі, що не дає змоги реалізуватися ідеям творчих учнів). Постійно появляються нові електронні посібники з програмування на Scratch, однак зауважимо, першість тут ведуть англомовні джерела. На сьогодні є багато сайтів для вивчення програмування на Scratch, проте в них є суттєвий недолік – мова інтерфейсу та недостатній рівень методичного викладу матеріалу. На нашу думку, для вивчення Scratch найбільш відповідні електронні джерела, оскільки вивчення Scratch безпосередньо пов'язано із графікою, звуком та анімацією, які набагато цікавіше демонструвати «вживу», аніж на папері.

Для досягнення поставленої мети авторами розроблено сайт «Вивчаємо Scratch разом» [9], який створено за допомогою оновленої версії Google-сайтів (див. рис. 2). На сайті, окрім навчального матеріалу, висвітлюються події, пов'язані із вивченням програмування на Scratch; розміщуються зразки кращих проектів, розроблених учнями; даються цікаві завдання для виконання всім бажаючим; розміщуються приклади реалізації ігрових проектів та створення мультфільмів. На даний момент процес розробки сайту продовжується, зокрема розробляється окрема сторінка сайту, присвячена проектуванню та створенню комп'ютерних ігор. Даний ресурс можна використовувати як для навчання програмуванню Scratch на уроках інформатики, так і для самопідготовки учнів, адже середовище програмування Scratch можна безкоштовно завантажити і вільно використовувати. За допомогою сайту учні мають можливість керувати своїм навчанням, користуватись готовими розробленими сценаріями та опубліковувати власні проекти в міжнародній спільноті в мережі Інтернет (<http://scratch.mit.edu>).

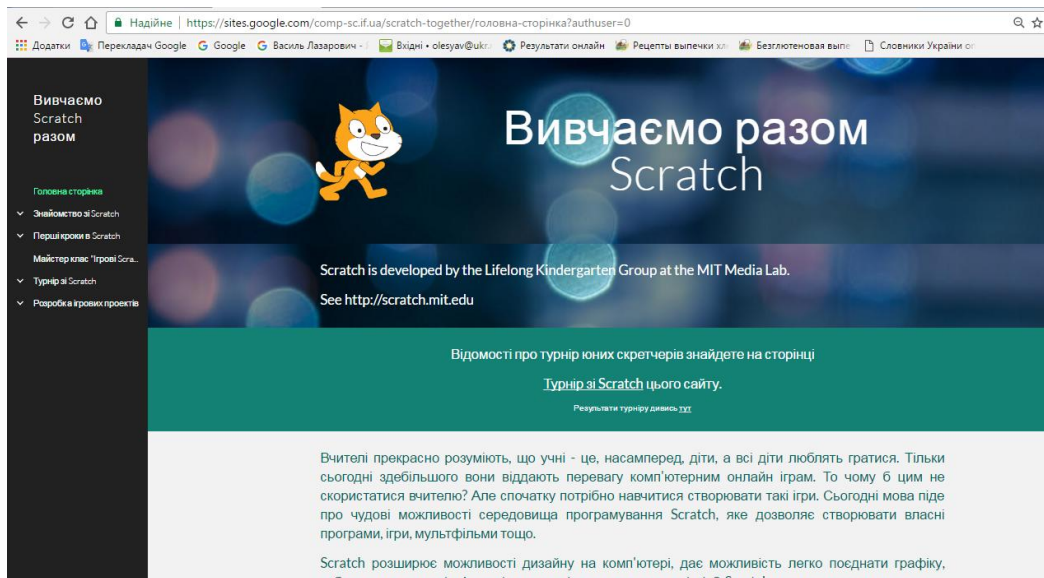
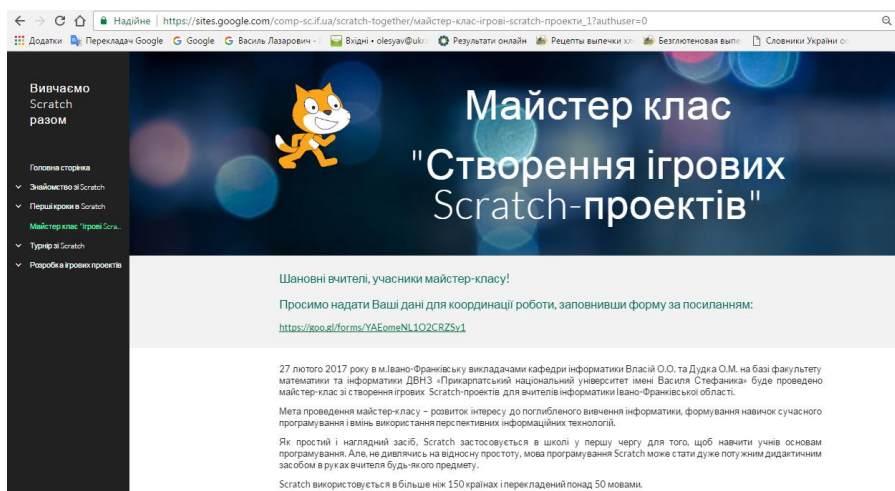


Рис. 2 Навчальний сайт для вивчення Scratch

Сайт має великий потенціал для активного та інтерактивного навчання, розвитку навичок мислення вищого порядку і більшої гнучкості навчального процесу, дозволяє учням використовувати різні способи особистісного та інтелектуального освоєння нових понять в процесі візуалізації абстрактних ідей та інтерактивної взаємодії. Педагогічний потенціал сайтів в першу чергу полягає у тому, що учні в процесі одержання, трансформації знань і подальшому публікуванні своїх робіт вчаться конструювати знання, засновані на відносинах і спілкуванні. Для викладачів опубліковані праці учнів – це можливість зробити висновки про те, як учні трансформують і засвоюють зміст і стратегію. Для бажаючих вчитися подібна публікація – це матеріал для подальшої рефлексії і аналізу, який дозволяє їм ще раз звернутися до своїх робіт і переосмислити їх, збагатити таким чином свій навчальний досвід.

**Апробація.** В якості апробації отриманих результатів на факультеті математики та інформатики Прикарпатського національного університету імені Василя Стефаника було проведено обласний майстер-клас зі створення ігрових Scratch-проектів, матеріали якого були розміщені на цьому сайті (див. рис. 3). Варто зазначити, що кількість учасників перевищила очікування – близько 70 дітей, причому з різними рівнями знання Scratch, виявили бажання дізнатися про Scratch та понад 20 вчителів інформатики підтримали своїх вихованців у їх прагненнях до вдосконалення знань зі Scratch.





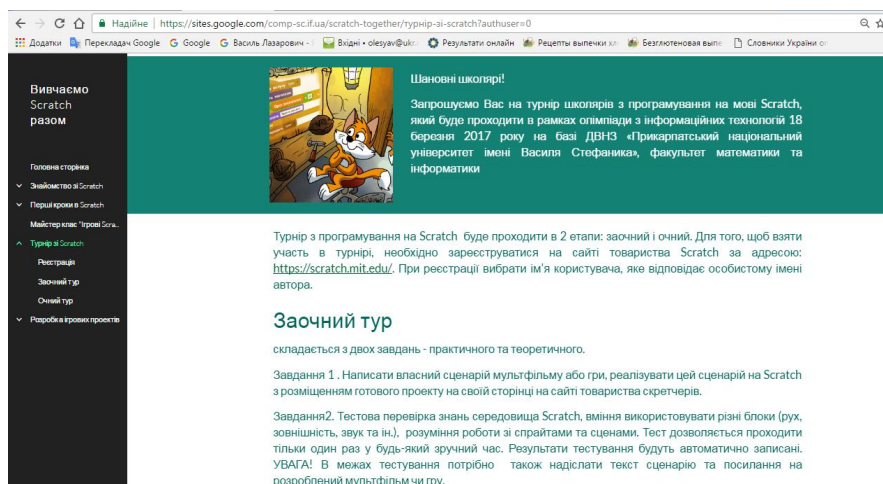


Рис. 3 Використання сайту для організації творчих заходів

З метою раннього залучення молодших школярів до програмування також вперше було проведено обласний турнір юних програмістів на мові Scratch серед учнів 8-12 років, який проходив в 2 етапи: заочний і очний, матеріали якого також розміщені на сайті (див. рис. 3). В заочному турі прийняли участь понад 60 школярів, до очного туру було допущено 40 учасників. Учні проявили значну зацікавленість розробкою ігрових Scratch-проєктів, під час виконання завдань проявляли активність та креативність при реалізації запропонованих ідей.

На даний момент вчителі інформатики Івано-Франківської області використовують розроблений сайт на своїх уроках для навчання програмуванню Scratch, а їх учні – для вдосконалення своїх знань зі Scratch.

Для того, щоб ефективно використовувати сайти у навчальному процесі слід дотримуватися таких рекомендацій: по-перше, викладач повинен ознайомитись з сайтами створеними іншими викладачами з метою формування уявлення про те, як вони можуть використовуватися у навчальному процесі; по-друге, викладач повинен створювати власні сайти, щоб мати уявлення про їх функції і можливості на практиці; по-третє, перед тим як запропонувати такий засіб навчання учням необхідно змоделювати сайт для своїх учнів (розробка правил, обговорення тематики, попередня підготовка матеріалів); по-четверте, створені сайти необхідно популяризувати, щоб відбувався процес обговорення та спілкування.

Розробка педагогами сайтів до певної дисципліни або теми дозволяє здійснювати науково-методичний супровід процесу навчання оперативного, через вчасне інформування та консультування. Сайт розкриває можливості тісного спілкування учителя з учнями. За допомогою різноманітних віртуальних ресурсів учень отримує методичну допомогу в зручний для нього час.

Все це дозволяє вивести сучасне заняття на якісно новий рівень: підвищується статус викладача; розширюються можливості ілюстративного супроводу заняття; використовуються різні форми навчання та види діяльності в межах одного заняття; ефективно організовується контроль знань, вмінь та навичок учнів; полегшується та вдосконалюється розробка творчих проєктів.

**Висновки.** Візуальна мова програмування Scratch має всі можливості для розробки ігрового програмного забезпечення. Простий і доступний інструментарій роботи в середовищі Scratch робить процес створення ігрового проєкту цікавим та зрозумілим і водночас дає можливість в повній мірі відчувати особливості технологій розробки комп'ютерних ігор. В наші дні програмування нерідко називають другою грамотністю. Його вивчають і в школі, і у вузах, хоча, на жаль, далеко не завжди успішно. Важливість освоєння цієї дисципліни не викликає сумніву з урахуванням тенденцій впровадження інформаційних технологій у різні аспекти нашого життя.

1. Яку мову програмування вивчати у школі (матеріали для дискусії) // Комп'ютер у школі та сім'ї. – 2013. – № 7. – С.14-18. – Режим доступу: [http://nbuv.gov.ua/UJRN/komp\\_2013\\_7\\_5](http://nbuv.gov.ua/UJRN/komp_2013_7_5).
2. Яку мову програмування вивчати у школі (матеріали для дискусії) // Комп'ютер у школі та сім'ї. – 2013. – № 8. – С. 9-18. – Режим доступу: [http://nbuv.gov.ua/UJRN/komp\\_2013\\_8\\_3](http://nbuv.gov.ua/UJRN/komp_2013_8_3)

3. Шевченко І. С. Використання навчального середовища SCRATCH при вивченні теми "Алгоритми та їх виконавці" / Шевченко І. С. // FOSS Lviv 2014, 24-27 квітня 2014 року — Л., 2014 — С. 114-117.
4. Величко В.С. Використання технології візуального програмування в університетській освіті засобами вільного програмного забезпечення/ В. С. Величко// Вісник Житомирського державного університету. – Випуск 4 (76). Педагогічні науки. – Режим доступу: <https://visnyk.zu.edu.ua/Articles/76/12.pdf>
5. Игровая индустрия в 2017 году: тренды, ожидания, советы [Електронний ресурс]. – Режим доступу: <http://renatus.com/ru/2017-gaming-industry-trends>
6. Комп'ютерні ігри як мистецтво [Електронний ресурс]. – Режим доступу: <https://goo.gl/MCXE7G>
7. Ігрова індустрія: тренди та прогнози [Електронний ресурс]. – Режим доступу: <http://innovations.com.ua/ua/articles/op-manage/18565/igrova-industriya-trendi-ta-prognozi>
8. Scratch : Programming for All / Mitchel Resnick, John Maloney, Andres Monroy Hernandez, Natalie Rusk, Evelyn Eastmond, Karen Brennan, Amon Millner, Eric Rosenbaum, Jay Silver, Brian Silverman, Yasmin Kafai // Communications of the ACM. – 2009. – Vol. 52, No. 11. – P. 60 – 67.
9. Вивчаємо Scratch разом [Електронний ресурс]. – Режим доступу: <https://sites.google.com/comp-sc.if.ua/scratch-together>