

шкільних дисциплін, допоможуть при вивченні інших шкільних дисциплін, а також при більш серйозному вивченні програмування в старших класах. Крім того, робота з середовищем Scratch дозволить освоїти навички інформаційної діяльності в глобальній мережі: розміщення своїх проєктів на сайті, обмін ідеями з користувачами інтернет-спільноти, оволодіння культурою спілкування на форумі.

Список використаних джерел

1. Морзе Н., Барна О., Вембер В., Кузьмінська О. Система компетентнісних завдань як засіб формування компетентностей на уроках інформатики. Інформатика та інформаційні технології в навчальних закладах. 2015. № 4. С. 17–27. URL: <https://bit.ly/2R1n9Vm>. (дата звернення 6.03.2020).
2. Морзе Н. В., Барна О. В. Я досліджую світ. Підручник для 3 класу закладів загальної середньої освіти (у 2-х частинах): Ч. 2. К.: УОВЦ «Оріон», 2020. 172 с.
3. Дудка О. М., Власій О. О. Особливості вивчення програмування на Scratch. Комп'ютерно-інтегровані технології: освіта, наука, виробництво. Науковий журнал. 2017. № 26. С. 81-87. URL: <http://ki.lutsk-ntu.com.ua/node/134/section/17>. (дата звернення 3.03.2020).

ПРОПЕДЕВТИКА АЛГОРИТМІЗАЦІЇ ЗАСОБАМИ SCRATCH

Машталір Оріся Володимирівна

магістрантка спеціальності Середня освіта (Інформатика)
Тернопільський національний педагогічний університет імені Володимира Гнатюка
м. Тернопіль, Україна
orusya1902@gmail.com

Лещук Світлана Олексіївна

кандидат педагогічних наук, доцент кафедри інформатики та методики її навчання
Тернопільський національний педагогічний університет імені Володимира Гнатюка
м. Тернопіль, Україна
leshchuk_so@fizmat.tnpu.edu.ua

Дільна Наталія Зіновіївна

кандидат фізико-математичних наук, старший науковий співробітник
Інститут математики Словацької академії наук
м. Братислава, Словацька республіка
nataliya.dilna@mat.savba.sk

У сучасному інформаційному суспільстві професія програміст затребувана та високооплачувана. Враховуючи специфіку фаху, вагомим фактором є самоосвіта, адже інформаційні технології постійно вдосконалюються, періодично з'являються новітні технологічні рішення, зростає значення досягнень у програмуванні. Тому важливо надати ґрунтовну базу школярам у напрямку вивчення алгоритмізації та програмування практично починаючи з початкової школи. Великих перспектив в цьому плані можна досягти, використовуючи Scratch [1; 4].

Можливості програмування на Scratch висвітлюють у своїх напрацюваннях багато вчених. Зокрема, Т. Є. Сорокіна, Є. Д. Патаракін [6; 7] розкривають різні напрямки роботи у середовищі Scratch: деякі можливості застосування графічного редактора, проектування, моделювання, створення інтерактивних навчальних матеріалів, реалізація самостійного навчання та робота у співпраці. Аналіз

публікацій в іноземних виданнях свідчить про накопичення значного досвіду щодо впровадження середовища програмування Scratch у навчальний процес [2; 3; 5].

Програмування, наразі, є однією з найцікавіших справ, а програмістів, навіть, називають «повелителями» комп'ютерів. У 2003 році група дослідників з Массачусетського технологічного університету (керівник Мітчел Резнік) створює загальнодоступну мову програмування – Scratch. Є думка, що це слово іде від «подряпина» (*англ. to scratch – дряпнути*), інші вважають, що назва Scratch походить від американського виразу «to start from scratch» – «розпочати з нуля (з основ)». Саме у Scratch є можливість навчитися дітям основам програмування. Це відкриває їм шлях до таких професій, як програміст, схемотехнік, конструктор, технічний дизайнер.

Професор Мітч Резнік пояснює, що вміння програмувати – це набагато більше, ніж просто технічні навички. Написання програмних кодів розвиває аналітичне мислення, вчить мислити логічно, розвиває уяву, креативність. У початковій школі програмування вчить дітей мислити швидко і чітко: вони розуміють, як влаштований світ, встановлюють логічний ланцюжок подій і можуть передбачити, що буде далі. Чим раніше починається навчання, тим винахідливішою і креативнішою стає людина [8].

Scratch є середовищем об'єктно-орієнтованого візуального програмування, яке надає можливості створювати комп'ютерні анімації, мультимедійні презентації, інтерактивні матеріали у вигляді історій та ігор, моделі та ін. Створюючи проекти Scratch, ехус опановують навички 21 сторіччя, які їм будуть необхідні для успішної самореалізації та майбутнього успіху: творче мислення; прозоре спілкування; системний аналіз; використання технологій; ефективна взаємодія; проектування; постійне навчання.

Основні особливості Scratch:

1. *Блочне програмування*: для створення програм потрібно просто сумістити графічні блоки в стеках. Блоки зроблені так, щоб їх можна було збирати лише в синтаксично вірні конструкції, що виключає помилки. Різні типи даних мають різні форми, що підкреслює їх несумісність. Користувач може вносити зміни в стеки навіть тоді, коли програма вже запущена, що дає змогу більше експериментувати з основними ідеями знову і знову.

2. *Маніпуляції даними*: використовуючи Scratch можна створювати програми, котрі змішують графіку, анімацію, музику.

3. *Робота в команді, обмін інформацією*: на сайті проекту Scratch можна подивитись проекти інших людей, використовувати і змінювати їх картинки та скрипти.

На рис. 1 подана схема, на якій перераховані передумови вибору Scratch в навчальному процесі:



Рис. 1. Вибір Scratch в навчальному процесі

Сама ідеологія Scratch дає змогу використати при навчанні сучасні методики і технології навчання, такі як проблемний підхід і метод проектів. Та основним моментом є те, що відбувається пропедевтика основних понять програмування. Виходячи з досвіду роботи в Бучацькій ЗОШ №3 I–III ст., є підстава стверджувати про необхідність циклічного вивчення дітьми матеріалу (див. табл. 1) і ефективність застосування Scratch.

Таблиця 1

Вивчення алгоритмів засобами Scratch

Розділ	Основні поняття	Навики та вміння
3 клас		
Алгоритми виконавці	і - команда; - виконавець; - алгоритм; - система команд виконавця; - висловлювання; - істинне і хибне висловлювання.	- вміти формувати команди для виконавця; - складати алгоритми за зразком; - розрізняти алгоритмічні структури; - розрізняти істинні та хибні висловлювання, формувати висловлювання з логічним слідуванням.
4 клас		
Алгоритми розгалуженням повторенням	з і - розгалуження; - алгоритм з розгалуженням; - цикл;	- складати, виконувати, змінювати алгоритм з розгалуженням; - розуміти відмінності між повним та неповним розгалуженням.
5 клас		
Алгоритми програми	та - програма; - середовище виконання алгоритму; - лінійний алгоритм; - блок-схема; - циклічний алгоритм; - цикл із лічильником; - цикл з умовою.	- складати прості алгоритми для виконавця в комп'ютерному середовищі та алгоритмів з повсякденного життя; - використовувати циклічні алгоритми в середовищі Scratch; - будувати лінійні алгоритми та реалізувати їх в середовищі програмування.
6 клас		
Алгоритми програми	та - об'єкт; - подія; - вкладені цикли; - вкладена алгоритмічна структура	- наводити приклади виконавців та систем команд виконавців алгоритмів; - вміти записувати алгоритми у вигляді послідовності команд виконавця; - вміти наводити приклади

		алгоритмів різних структур: із об'єктами та подіями; - діяти за інструкцією, планувати свою діяльність, аналізувати і робити висновки.
7 клас		
Алгоритми повторенням розгалуженням	з і - висловлювання; - умовне висловлювання; - величини; - змінні.	- встановлювати істинні, хибні та умовні висловлювання; - використовувати змінні при створенні проектів в Scratch; - розуміти базові алгоритми роботи із змінними: обмін значеннями, визначення найбільшого і найменшого з двох значень.

Для того, щоб діти ефективно засвоювали знання в новій галузі, необхідно наявність мотивації. А мотивація навчальної діяльності виникає, якщо вони відчують особисту зацікавленість у здобутті знань для досягнення своєї мети. Розглянуте програмне забезпечення дає змогу створювати алгоритми, програми, проекти, які цікаві безпосередньо віковій категорії молодших школярів. А також сприяють розвитку у дітей алгоритмічного мислення. При проведенні регулярних розвиваючих занять, систематично організованих цікавих завдань створюються сприятливі умови для формування такої цінної якості, як алгоритмічне мислення, як самостійність, що виявляється в активному та ініціативному пошуку рішень задач, в глибокому та всебічному аналізі їх умов, в критичному обговоренні та обґрунтуванні шляхів вирішення, в попередньому плануванні та передбаченні різних варіантів здійснення розв'язку.

Отже, використання можливостей Scratch дає можливість ефективно проводити пропедевтику програмування. Розглянуте середовище займає одне з перших місць програмних середовищ візуального програмування для школярів. Scratch – це не тільки мова програмування, а й вдале програмне середовище для проектної діяльності.

Список використаних джерел

1. Василенко Я. П., Кирстюк І. П. Про педагогічні та дидактичні особливості середовища Scratch як інструменту навчання основам алгоритмізації та програмування. *Матеріали III Міжнародної науково-практичної інтернет-конференції «Сучасні інформаційні технології та інноваційні методики навчання: досвід, тенденції, перспективи»*. Тернопіль, ТНПУ, 2019. С. 21–24.
2. Денисова Л. В. Проектная деятельность школьника в среде программирования Scratch Оренбург. 2009. URL: <https://sites.google.com/site/orensratch/nasirazrabotki>. (дата звернення 7.02.2020).
- Денисова Л.В. Среда Scratch в практике учителя начальной школы *Начальная школа*. 2012. № 5. С. 31–35.
3. Лещук С. О. Scratch: ази формування програмістського мислення. FOSS Lviv – 2018: матеріали міжнар. наук.-практ. конф., м. Львів, 26-29.04 2018 р.
4. Патаракин Е.Д. Педагогический дизайн социальной сети Scratch. *Образовательные технологии и общество (Educational Technology & Society)*. 2013. С. 505–528.
5. Патаракин Е.Д. Школа Scratch *Шкільні технології*. 2010. С. 132–135.
6. Сорокина Т.Е. Визуальная среда Scratch как средство мотивации учащихся основной школы к изучению программирования *Информатика и образование*. 2015. № 5 (264). С. 30–34.

7. Maloney J., Burd L., Kafai Y., Rusk N., Silverman, B., and Resnick, M.. "Scratch: A Sneak Preview, Proceedings of IEEE Conference on Creating, Connecting, and Collaborating through Computing (Kyoto, Japan), 2004, p. 84.

ОСОБЛИВОСТІ ВИВЧЕННЯ PYTHON В РІЗНИХ ФОРМАХ НАВЧАННЯ

Струк Оксана Олегівна

кандидат фізико-математичних наук, доцент, кафедра інформатики та методики її навчання
Тернопільський національний педагогічний університет імені Володимира Гнатюка
м. Тернопіль, Україна
oksana.struk@gmail.com

Струк Олександр Сергійович

студент, факультет інформатики, математики, фізики
Люблінський університет Марії Кюрі Скадовської
м. Люблін, Польща
sasha.struk1@gmail.com

Стрімкий розвиток сприяє модернізації сучасної системи освіти. Сучасна дистанційна освіта – це розгалужена система передачі знань на відстані за допомогою різних засобів і технологій, яка сприяє отриманню студентами необхідної інформації для використання у практичній діяльності [1]. Дистанційне навчання – це така форма організації навчального процесу та педагогічна технологія, основою якої є керована самостійна робота студентів та широке застосування у навчанні сучасних інформаційно-комунікаційних технологій [3].

В Україні проблемі дистанційної освіти присвячено роботи В. Бикова, В. Габрусєва, О. Глазунової, Н. Думанського, О. Захар, П. Камінської, В. Кухаренка, Н. Морзе, К. Обухової, В. Олійника, Є. Полат, О. Рибалко Н. Сиротенко, О. Спіріна, Ю. Триуса.

Платформа масових відкритих онлайн-курсів Prometheus – перший та найбільший проект безкоштовної освіти в Україні. На цій платформі розташовано курс «Основи програмування» В ньому подаються базові елементи мови Python.

Перше заняття ознайомлює нас з історією розвитку обчислювальної техніки, оглядом теми «Системи числення», описом будови персонального комп'ютера, та встановленням мови Python на комп'ютері.

В рамках другого заняття розглядаються основні типи алгоритмів, інтерпретатори та компілятори, арифметичні операції в Python. Крім того, тут подано навчальну гру «Лабіринт» Blockly, за допомогою якої можна навчитись складати програми.

Третє заняття «Готуємо пітона» розповідає нам про структуру програми. Серед питань, описаних на занятті, є питання тестування програми, що є дуже корисним.

В четвертому занятті «Алгоритмічні структури у мові Python» висвітлюються питання розгалуження if..else, логічний тип даних, списки, цикл обходу послідовності for, універсальний цикл while.