

**Автор:**

*Козолуп Євгеній Вікторович  
студент 2 курсу спеціальності  
014 Середня освіта (інформатика)*

**Науковий керівник:**

*Франчук Василь Михайлович  
кандидат педагогічних наук, доцент,  
професор кафедри комп'ютерної інженерії та  
освітніх вимірювань*

## **МЕТОДИЧНІ РЕКОМЕНДАЦІЇ НАВЧАННЯ ПРОГРАМУВАННЯ В ЗАКЛАДАХ ЗАГАЛЬНОЇ СЕРЕДНЬОЇ ОСВІТИ ДЛЯ УЧНІВ 6-7 КЛАСІВ**

Комп'ютер став невіддільною частиною повсякденного існування, на сьогодні вже не достатньо просто вміти користуватися програмними продуктами. Потрібно розуміти як вони працюють та яким чином створюються для того, щоб бути попереду технологій. Тобто акцентувати увагу саме на вивченні тем алгоритмізації та програмування у шкільному курсі інформатики.

Проте учням доволі важко сприймати подібний матеріал, не говорячи вже про безпосередньо розробку та створення алгоритмів та програмних рішень за допомогою спеціальних додатків або мови програмування. Особливо це стосується підлітків 11-14 років.

Саме тому наразі виникло питання, як максимально ефективно і просто навчити дітей середнього підліткового віку основам алгоритмізації та програмування в обмеженому курсі шкільної інформатики.

За навчальною програмою з інформатики для учнів 5-7 класів закладів загальної середньої освіти не менше ніж 40% від загальної кількості навчальних годин, вчитель має витратити на вивчення теми “Алгоритми та програми”.

Зміст навчального матеріалу передбачає вивчення основних понять теми “Алгоритми та програми”, а також базових алгоритмічних структур. Разом з тим з кожним роком основні поняття доповнюються або замінюються іншими, для кращої адаптації навчальної програми під вікові особливості учнів.

Так наприклад, у 5-му класі, учні використовують за основу поняття “виконавця” алгоритму та його властивості. Це зручно, тому що так поняття алгоритму можна легко пов'язати з реальним життям через практичні приклади.

У 6-му класі відбувається перехід від поняття “виконавця” алгоритму до поняття “програмного об'єкту”. Тобто від того *хто*, виконує дії, до того *над чим* ці дії виконуються.

На противагу поняттю програмного об'єкта в 7-му класі вводиться базове поняття програмування – поняття змінної величини. В більшості мов програмування поняття об'єкта і змінної, до речі є тотожними або взаємопов'язаними, але є й такі мови в яких вони принципово відрізняються.

Разом з цим навчальна програма з інформатики не обмежена конкретним програмним забезпеченням, тому вчитель сам в змозі обрати те середовище алгоритмізації та програмування, використання якого, на його думку, максимально допоможе учням засвоїти новий навчальний матеріал.

В результаті проведеного аналізу існуючих наукових напрацювань в цій галузі було отримано такі результати: в більшості робіт розглядається вивчення теми “Алгоритми та програми” у 5-7 класах на прикладах візуальних середовищ опису та виконання алгоритмів такі як Blockly, Code.org, Scratch та інші.

Головною перевагою таких середовищ є простота їх використання. Проте недоліком візуальних середовищ опису та виконання алгоритму є те, що в основі їх роботи покладено поняття “виконавця”, для якого учні мають розробити алгоритм дій.

Через це під час навчання теми “Алгоритми та програми” в 6-7 класі, може виникнути суперечність між поняттям об’єкта та виконавця, якщо пояснювати ці поняття на прикладі одного середовища візуального опису та виконання алгоритмів.

Для кращої адаптації учнів до вивчення основних парадигм програмування в старших класах є доречним поступово впроваджувати вивчення основ програмування на прикладі простих мов програмування, таких як мова Python. Ця мова побудована за принципом об’єктно-орієнтованого програмування. Тобто, всі величини в цій мові по суті є **об’єктами** того чи іншого класу, а всі основні дії зображаються у вигляді простих функцій з певною кількістю аргументів.

Базові алгоритмічні структури розгалуження та циклу створюються за допомогою спеціальних коротких операторів, синтаксично таку конструкцію програмного коду в цій мові відрізняє лише наявні відступи від лівого краю текстового поля, не має жодних додаткових синтаксичних позначень.

Проте єдиним недоліком використання мови програмування Python для вивчення теми “Алгоритми та програми” у 6-му класі в повному обсязі є невідповідність шкільній програмі з математики. Через те, що базові операції виведення та введення значень та багато корисних математичних дій в цій мові є функціями, це може стати досить серйозною проблемою для учнів, через те, що поняттям “функції” вони не володіють.

Проте за даною методикою учителю пропонується обмеживши синтаксис мови від більшої кількості подібних функцій. Тобто адаптувати методи та способи реалізації програмних засобів до навчальної програми з інших дисциплін.

Так, наприклад, учитель повинен обмежити (звужити) деякі алгоритмічні поняття в силу особливостей рівня математичної обізнаності учнів 6-7 класів. Розглядаючи тему “Типи даних” на уроках інформатики, не зосереджувати увагу на роботі з усіма типами та переведенням даних у інші типи даних, а обмежитися лише натуральними та цілими числами. Також, обмежити використання більшості функцій для запиту та опрацювання значень об’єктів.

Наприклад, учень може вручну створити об’єкт в програмному коді, а в подальших випадках самостійно змінювати його значення.

У 7-му класі учнів вже мають освоїти поняття математичної функції, тому вчитель інформатики може поступово адаптувати це поняття до теми “Алгоритми та програми” та в подальшому використовувати всі можливості цієї мови програмування.

На допомогу учителеві в реалізації навчання мови програмування Python в межах цього дослідження було розроблене “Тренувальне середовище програмування”, в якому більша частина коду буде замінена на графічні елементи управління (Рис. 1).

Вікно програми поділене на 3 основні області: область вхідних даних, текстове поле для введення програмного коду та область виведення результату.

В області вхідних даних розташовані поля для введення вхідних даних програми та кнопка запуску виконання алгоритму. Під час цього нові програмні об’єкти класу цілих чисел можна додати, натиснувши на кнопку “+”, що знаходиться під останнім полем.

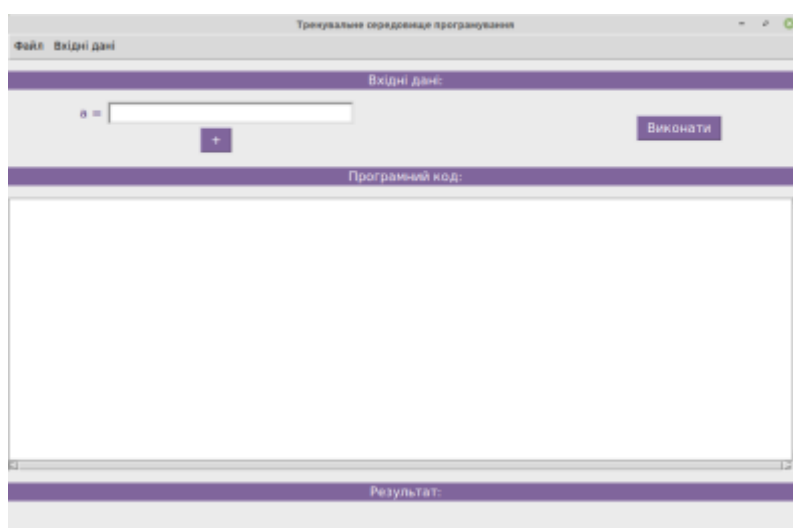


Рис. 1

Рис. 1

В полі програмний код прописується відповідний код, з урахуванням того, як називаються вже створені вище змінні. Для виведення значень потрібно навчитися працювати з єдиною функцією *print()* та вносити в неї в якості аргументу ім'я об'єкта, що є результатом роботи програми. Для запуску алгоритму дій, потрібно натиснути кнопку "Виконати", що розміщена в області вхідних даних.

У майбутньому під час вивчення теми "Алгоритми та програми" вже в 7-му класі, учитель може ввести поняття функції та запропонувати учням використовувати їх для створення повноцінних програмних рішень, разом з цим не змінюючи середовище програмування. У вкладці "Вхідні дані" під час вибору пункту "Поле для введення", замість окремих текстових полів для окремих об'єктів створюється одне загальне поле, зчитати дані з якого можна з допомогою функцій для запиту значень у користувача.

Розроблені методичні рекомендації навчання, спрямовані на полегшення вивчення основ програмування мовою програмування Python для учнів 6-7 класів закладів середньої освіти та впроваджуються в навчальний процес Сумської спеціалізованої школи І-ІІІ ступенів №17, м. Суми у 2019-2020 н.р.

Було проведено опитування для оцінки зацікавленості та мотивації учнів продовжувати вивчати тему "Алгоритми та програми" в курсі інформатики 6-го класу. Опитування пройшли 43 учня з 6 класу Сумської спеціалізованої школи І-ІІІ ступенів №17, м. Суми.

Важливим фактором в оцінюванні ефективності запропонованих методичних рекомендацій є визначення учнями різниці між очікуваним рівнем складності навчання основам програмування реальною мовою програмування в темі "Алгоритми та програми". Загалом до початку вивчення, учні вважали, що загальний рівень складності складатиме 3,28 із 5. В процесі вивчення теми, цей показник зменшився до 2,70 із 5.

Разом з тим, збільшився й рівень зацікавленості учнів до продовження вивчення мови програмування Python за запропонованими методами в 6 класі до 4,07 із 5.

З результатів опитування видно, що запропоновані методичні рекомендації щодо вивчення теми "Алгоритми та програми" з використання об'єктно-орієнтованої мови програмування Python є ефективними під час застосування в 6 класі закладів середньої освіти.

На основі отриманих результатів можна зробити такі висновки:

Через швидкість розвитку сучасних інформаційних технологій, викликає доцільність в максимально ефективному засвоєнні учнями основ алгоритмізації та програмування в закладах загальної середньої освіти. На разі більшість наукових напрацювань орієнтовані на вивчення даної теми в 6-7 класах за допомогою візуальних середовищ опису та виконання алгоритмів. Проте для кращого вивчення даної теми є більш доречним використання реальних навчальних мов програмування та їх адаптація до навчальної програми вищезгаданих класів загальноосвітніх навчальних закладів. Використання запропонованих методичних рекомендацій та розробленого навчального середовища програмування підвищує ефективність вивчення теми "Алгоритми та програми" з використанням реальної мови програмування Python, що є досить популярною та практичною на сьогоднішній день. Набуті навички в процесі вивчення цієї теми зможуть допомогти учням в їх подальшій навчальній та науковій діяльності.

В подальшому планується розширити дослідження в 8-9 класах.

#### Список використаних джерел

1. Козолуп Є.В. Програмування в школі. Мова Python : Навчальний посібник. 8 клас. Суми, 2017. 82 с.
2. Навчальна програма з інформатики для 5-9 класів, 2017 рік. URL: <https://mon.gov.ua/storage/app/media/zagalna%20serednya/programy-5-9-klas/onovlennya-12-2017/8-informatika.docx> (Дата звернення: 10.01.2020)
3. Оніщенко С.М., Франчук В.М. Delphi (Kylix) як засіб навчання основ програмування. Науковий часопис НПУ імені М. П. Драгоманова. Серія №2. Комп'ютерно-орієнтовані

системи навчання: Збірник наукових праць. Київ: НПУ імені М. П. Драгоманова, 2006. №4 (11). с. 126-130.