

opportunities of using computer mathematics software in relation to the management of independent work of pre-service teachers in the process of teaching computer science, in particular numerical analysis. The following free software tools of high scientific and methodological popularity, both among domestic and foreign researchers, have been chosen for analysis: Gran1, Gran-2D solvers; Wolfram|Alpha computational knowledge system, SageMathCloud cloud-oriented computer mathematics system. In addition, the paper concerns the use of Python programming language, which provides powerful tools for performing scientific and engineering calculations. Besides, there is presenting of the main areas of application of numerical analysis of the considered software and the methods, which are the basis of execution of some software commands. Features of students' independent work are considered by the example of training of numerical analysis course, that includes strong interdisciplinary communication and promotes creative and research skills of pre-service teachers. The paper also considers an example of solving the problem of function interpolation as one of the most common methods for polynomial approximation in the course of numerical analysis. The proposed software can be used at different stages of management of pre-service teachers' independent work in the process of teaching numerical analysis.

Keywords: students' independent work management, computer mathematics system, numerical analysis, Gran1, Gran-2D, Wolfram | Alpha, SageMathCloud, Python.

DOI 10.31392/NPU-nc.series 2.2020.22(29).26
УДК 373.5.016:004.4

Євгеній Вікторович Козолуп

студент 3-го курсу факультету інформатики

Національний педагогічний університет імені М.П. Драгоманова

ORCID ID: 0000-0002-9098-5602

18fi.ye.kozolup@std.npu.edu.ua

РОЗРОБКА НАВЧАЛЬНО-ТРЕНУВАЛЬНОГО СЕРЕДОВИЩА ДЛЯ ВИВЧЕННЯ ОСНОВ ПРОГРАМУВАННЯ ДЛЯ УЧНІВ 6-7 КЛАСІВ ЗАКЛАДІВ ЗАГАЛЬНОЇ СЕРЕДНЬОЇ ОСВІТИ

Анотація. В статті розглядається важливий розділ навчального курсу інформатики – програмування. Навчання програмування розвиває пам'ять, логічне мислення, креативність, самоорганізованість, стійкість та інтелектуальні здібності. На разі кількість годин, передбачених навчальною програмою з інформатики на вивчення теми “Алгоритми та програми”, а також “Основи об’єктно-орієнтованого програмування” можна вважати більш ніж достатньою (від 30% до 50% в залежності від класу). Тому за системного та виваженого підходу до навчання, після завершення школи кожен учень може мати базові знання з програмування, а також вміти їх застосовувати для розв’язування практичних задач, що виникають під час різних видів його діяльності. Разом з цим в допомозі вчителю та учню можуть стати різноманітні тренувальні середовища розробки, а також навчальні веб-ресурси та сайти. Для таких сучасних та простих мов, як, наприклад Python, їх існує достатня кількість. Проте на жаль, наявні сервіси не зовсім придатні до використання в 6-7 класах під час вивчення теми “Алгоритми та програми”. Саме тому доцільним є розробити просте у використанні, але змістовно наповнене навчально-тренувальне середовище для вивчення основ програмування мовою Python в середній школі. Середовище PyLearn розроблено для спрощення адаптації учнів до правил написання програмного коду, а також його тестування. Використання такого підходу дозволяє максимальну частину навчального часу приділити саме розв’язуванню практичних задач, що надалі дозволить учневі тільки поглиблювати свої знання та в старших класах почати роботу над власним навчальним або науковим проектом з програмування.

В статті розглянуто короткий аналіз популярних програмних засобів для навчання програмування, а також описано принцип роботи з розробленим навчально-тренувальним середовищем PyLearn.

Ключові слова: об’єктно-орієнтоване програмування, мова програмування, середовище розробки програмного забезпечення, навчально-тренувальне середовище.

З кожним роком все більшої популярності на уроках інформатики в закладах загальної середньої освіти набуває використання сучасних об’єктно-орієнтованих мов програмування. Здебільшого це обумовлене тим, що синтаксис таких мов досить простий у вивченні, а самі мови розроблені за об’єктно-орієнтованою парадигмою програмування, що в свою чергу набагато простіше в розумінні того, за яким принципом створюються та взаємопов’язані програмні об’єкти.

Однією з таких мов, що набула неабиякої популярності серед школярів та їх учителів можна вважати мову Python. Окрім того, що створення програм з її використанням відбувається досить просто, за популярністю щодо використання у реальних проектах вона не поступається жодній з відомих мов. Таким чином, за системного вивчення основ програмування в тому числі мовою Python, учні вже в старших класах можуть почати розробку своїх власних навчальних та наукових проєктів, що було передбачено ще кілька десятиліть років тому, в період запровадження інформатики в школи України [1, 2].

Саме тому доцільно починати вивчення основ алгоритмізації та програмування мовою Python з 6 класу середньої школи, але важливо враховувати деякі особливості учнів цієї вікової групи [3]. Важливо розуміти, що ці учні тільки завершують свою адаптацію до старшої школи, та тільки починають вивчати деякі базові дисципліни, що тісно пов'язані з інформатикою та програмуванням. Тому для таких учнів потрібно створити максимально комфортні умови для простого та цікавого вивчення нового, досить складного теоретичного матеріалу. Також потрібно враховувати той факт, що учні 6-го класу не знають більшість вкрай важливих математичних об'єктів як, наприклад, функція чи множина, тому їх бажано уникати під час освітнього процесу на уроках інформатики.

Для реалізації такого підходу можуть стати у пригоді різноманітні тренувальні та навчальні середовища програмування, а також різноманітні веб-ресурси для вивчення основ програмування учнями шкіл. На разі таких середовищ для мови Python існує велика кількість, але не всі вони можуть використовуватися на уроках інформатики в закладах загальної середньої освіти.

Для визначення вимог до навчального програмного забезпечення, що може використовуватися під час вивчення теми алгоритмізації та програмування учнями 6-7 класів, було проаналізовано зміст типової програми з інформатики для учнів 5-9 класів [4], та інструктивно-методичні рекомендації щодо навчання предметів у закладах загальної середньої освіти у 2020/2021 навчальному році [5]. Таким чином було сформовано ряд вимог, яким мають відповідати навчально-тренувальні середовища програмування:

- програмне забезпечення має бути ліцензованим. У випадку, якщо залучення коштів для придбання програмного забезпечення неможливе, рекомендовано використовувати програмні забезпечення з відкритим програмним кодом [6];
- інтерфейс програмного засобу має бути україномовним, зрозумілий та простий для користувачів;
- засіб має підтримуватися на більшості платформ, вільно використовуватися на різних пристроях;
- програмне середовище має бути зручним у користуванні: без збирання непотрібних відомостей про користувача, не потребувати складного процесу авторизації чи реєстрації, не потребувати створення та збереження додаткових файлів та інших даних на комп'ютері користувача;
- використання програмного засобу користувачем не повинно потребувати значної кількості ресурсів персонального комп'ютера, та його має бути комфортно використовувати на пристроях з середніми характеристиками.

Таким чином, для визначення відповідних цим вимогам програмних середовищ було проведено аналіз найпопулярніших навчальних та тренувальних веб-ресурсів. До таких можна віднести: навчальні веб-ресурси SoloLearn, Питонтьютор, E-Olymp.

До важливих спільних характеристик всіх розглянутих засобів можна віднести їх веб-орієнтованість. Жоден із засобів, навіть під час завантаження відповідного додатку на смартфон, не може бути використаним повною мірою. Перевірка програмного коду відбувається на віддаленому сервері, на якому й розміщені навчальні матеріали.

Зручність такого підходу полягає у відсутності необхідності постійного оновлення баз навчального матеріалу користувачем, що неодмінно має відбуватися під час розробки програмних засобів для персональних комп'ютерів. В такому разі процес автоматичної перевірки розв'язків значно пришвидшується, адже не залежить від технічного забезпечення персонального комп'ютера користувача. До цього, зареєструвавшись в сервісі, користувач може з легкістю змінити робочий пристрій, а процес його роботи та навчання буде збережено.

З іншої точки зору використання веб-орієнтованого підходу не дає змогу користувачеві здійснювати своє навчання на пристроях, що не під'єднані до мережі Інтернет. Беручи до уваги той факт, що не всі школи України мають доступ до глобальних служб мережі Інтернет, а подекуди швидкість з'єднання не перебільшує 50-100 Мбіт/сек, чого абсолютно недостатньо для роботи не одного десятку комп'ютерів з усієї школи. Також важливим, а можливо й вирішальним фактором може бути потреба в реєстрації облікового запису користувача для здійснення подальших дій на сайті

Е-Оуптр. Таким чином процес авторизації, може тривати кілька хвилин, враховуючи вік дітей, що своєю чергою вплине на роботу учнів класу на уроці інформатики, час якого є обмежений.

До того ж не всі навчальні сервіси мають вбудовані середовища розробки програмного забезпечення, тому для роботи з цими сайтами потрібно додатково встановити те чи інше програмне забезпечення та здійснювати постійний перехід між програмними вікнами.

Саме тому також було проаналізовано популярні середовища розробки програмного забезпечення з використанням мови Python, що позиціонуються як навчальні. До таких можна віднести: Python IDLE, Thonny, а також веб-ресурс Jdoodle.

В результаті аналізу трьох популярних навчальних середовищ розробки програм з використанням засобів мови Python було визначено, що основними недоліками їх є англійський інтерфейс та використання інтерпретатора Python Shell для запиту та виведення даних. Також всі середовища для персональних комп'ютерів потребують обов'язкового збереження програмних файлів для виконання коду.

До середовищ, які було б доречно використовувати для навчання основ програмування в 6-7 класах можна віднести онлайн-редактор Jdoodle. Проте, важливо зауважити, що для продуктивної роботи з його використанням потрібно мати достатньо хороше Інтернет-з'єднання, а також володіти базовими знаннями англійської мови. Додамо, що інтерфейс сайту переповнений різноманітними рекламними матеріалами, що може заважати користувачеві.

З аналізу зрозуміло, що наявні навчальні та тренувальні засоби не достатньо розроблені з методичних позицій для використання на уроках інформатики в закладах загальної середньої освіти, адже жодне середовище не відповідає всім або більшості вимог, наведеним раніше.

Саме тому доцільним рішенням є спроектувати та розробити єдине навчально-тренувальне середовище для навчання основ програмування мовою Python, що буде складатися з трьох важливих складових (Рис. 1):

- динамічного довідника (посібника) з основ програмування мовою Python;
- практикуму для тренування в розв'язуванні задач з програмування;
- навчального середовища розробки програмного забезпечення.



Рис. 1.

Навчально-тренувальне середовище PyLearn розроблене з використанням засобів мови Python через її кросплатформність, а також для того, аби кожен вчитель за бажанням міг додати зміни в програмний код середовища та налаштувати його для власного застосування або навчання своїх учнів.

Отже, перша складова середовища – динамічний довідник, або динамічний посібник. Вікно посібника можна розділити на 2 частини: панель навігації та робоче поле (Рис. 2). Після обрання потрібної теми, в робочому полі з'являється відповідна сторінка довідника. Наразі йде робота над доопрацюванням довідника та створенням динамічних вправ до кожного розділу посібника.

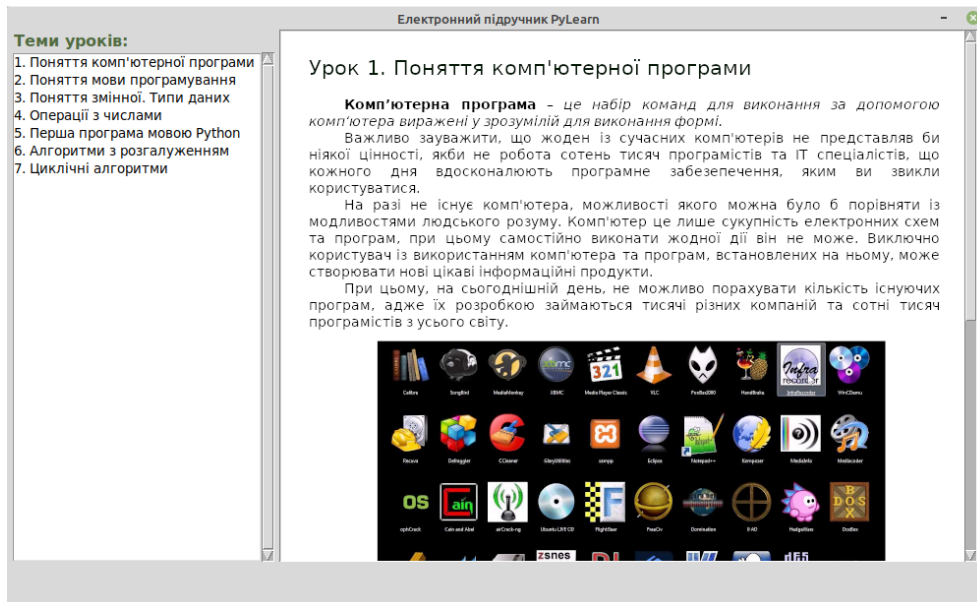


Рис. 2.

Для тренування набутих теоретичних знань на практиці під час розв'язування типових задач з програмування учень може використати програмний практикум PyLearn. Програмний практикум розроблений за прикладом сервісу E-Olymp, з використанням якого учень може розв'язувати різноманітні задачі з основ програмування мовою Python (Рис. 3).

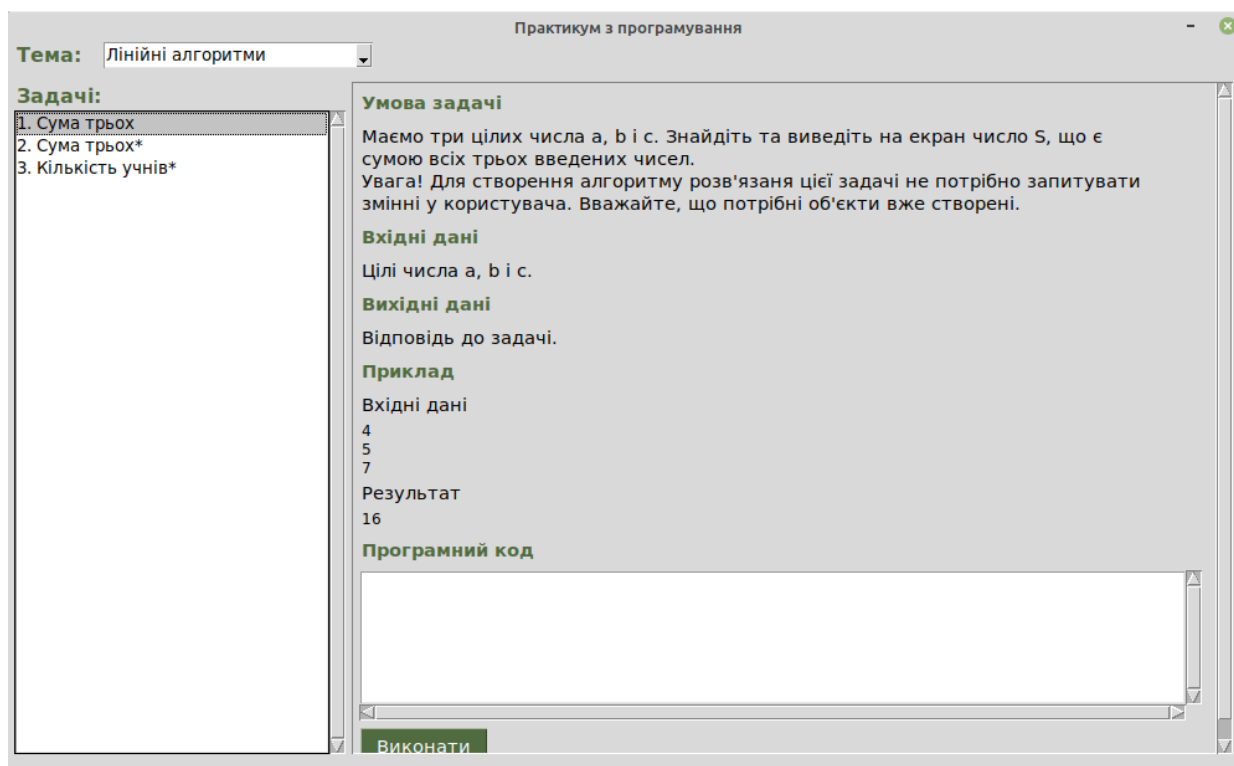


Рис. 3.

Таким чином після обрання відповідної теми та задачі з запропонованого списку учневі потрібно за умовою задачі написати у відповідне текстове поле програму-розв'язок та натиснути кнопку Виконати. В програмному практикумі передбачено автоматичну перевірку розробленого коду за допомогою певного набору тестів, що складаються з вхідних та вихідних даних. В залежності від кількості пройдених тестів користувач отримуватиме відповідні повідомлення на екрані (Рис. 4).

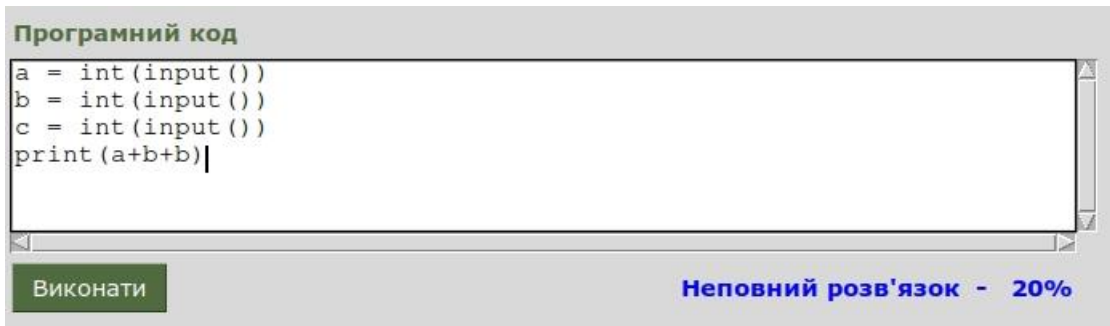


Рис. 4.

Разом з тим в деяких задачах не передбачаються ініціалізація та запит значень змінних, або використання деяких функцій мови програмування Python. Таким чином передбачено процес адаптації учнів 6-го класу до написання програмного коду. Тому користувачам навчально-тренувального середовища буде достатньо розробити тільки математичну модель розв'язування поставленої задачі, ніби відповідні програмні об'єкти вже створено.

На разі йде робота над вдосконаленням бази даних задач з різних тем розділу об'єктно-орієнтованого програмування мовою Python. Важливо зауважити, що поповнити базу завдань новими задачами може будь-який користувач, якщо він правильно створить відповідні файли з умовою та перевірними тестами та додасть їх до теки tasks.

Також для власних програмних розробок з використанням засобів мовою програмування Python було розроблене тренувальне середовище програмування, що є складовою PyLearn. Навчальне середовище було створено в тому числі внаслідок аналізу переваг та недоліків онлайн-редактора Jdoodle. Тобто, вікно середовища поділено на 3 основні частини: поле вхідних даних, поле для введення програмного коду, а також рядок виведення результату (Рис. 5).

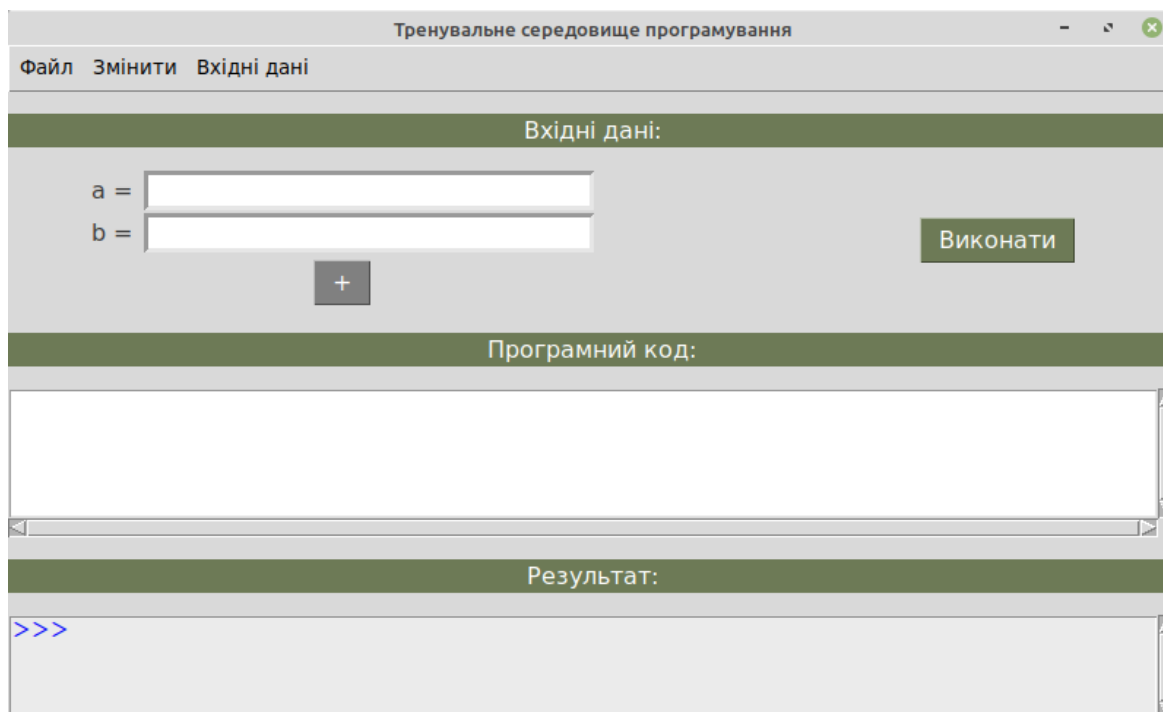


Рис. 5.

Разом з цим, як і у випадку з програмним практикумом, за замовчуванням середовище налаштовано для використання учнями 6-го класу. Таким чином, для створення програмних об'єктів (змінних) учневі не потрібно їх прописувати в програмному коді. Створення змінних відбувається за допомогою графічних елементів управління. А дані, введені через графічні поля, можуть використовуватися в самій програмі під зазначеними іменами. Таким чином учень має освоїти тільки функцію print() для виведення даних на екран.

Вже під час використання середовища в 7-му класі вчитель може запропонувати змінити формат вхідних даних на поле для введення, зчитування даних з якого можливе тільки через спеціальні функції (Рис. 6).

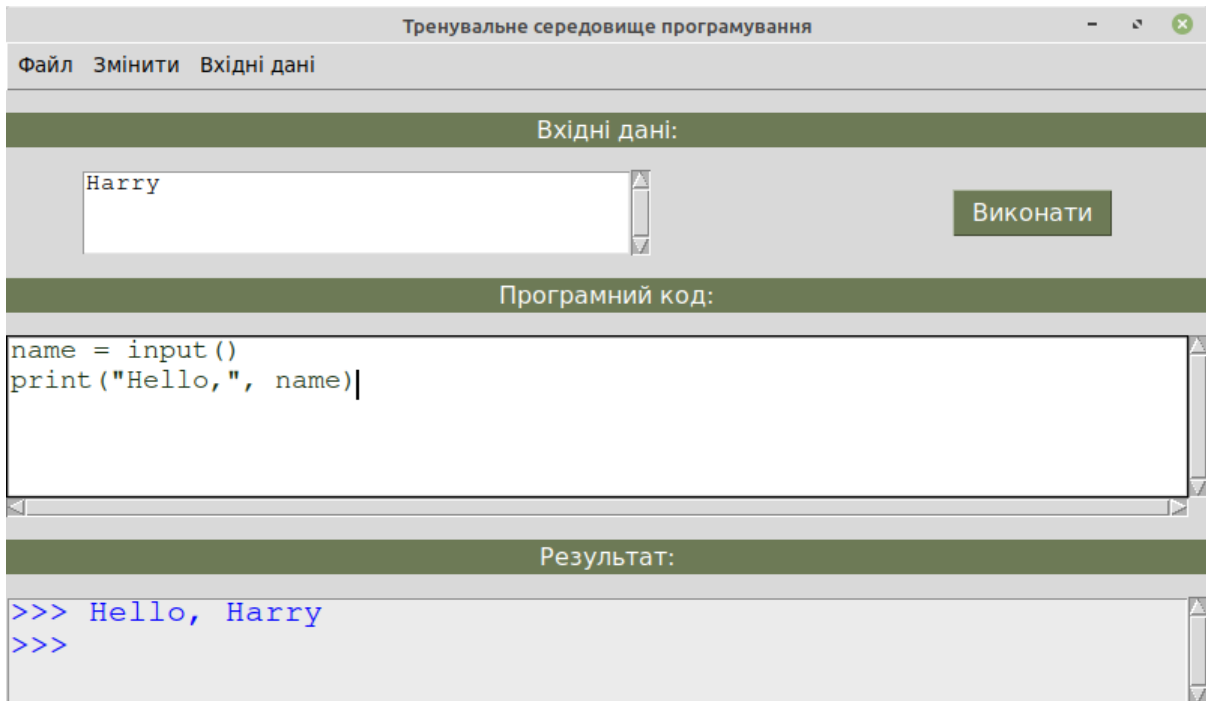


Рис. 6.

Важливо наголосити, що під час виконання програмного коду учневі не потрібно зберігати програмні файли, хоча така можливість передбачена.

На разі тестування ефективності використання розробленого середовища відбувається на уроках інформатики в 6-7 класах Сумської спеціалізованої школи I-III ступенів №17, міста Суми, Сумської області.

Проте вже важливо відмітити, процес вивчення основ програмування мовою Python з використанням цього програмного забезпечення значно легший, учні встигають виконати набагато більше практичних завдань, ніж з використанням стандартного IDLE. Разом з цим принципово збільшився рівень зацікавленості учнів школи, що пройшли відповідне опитування (Рис. 7).

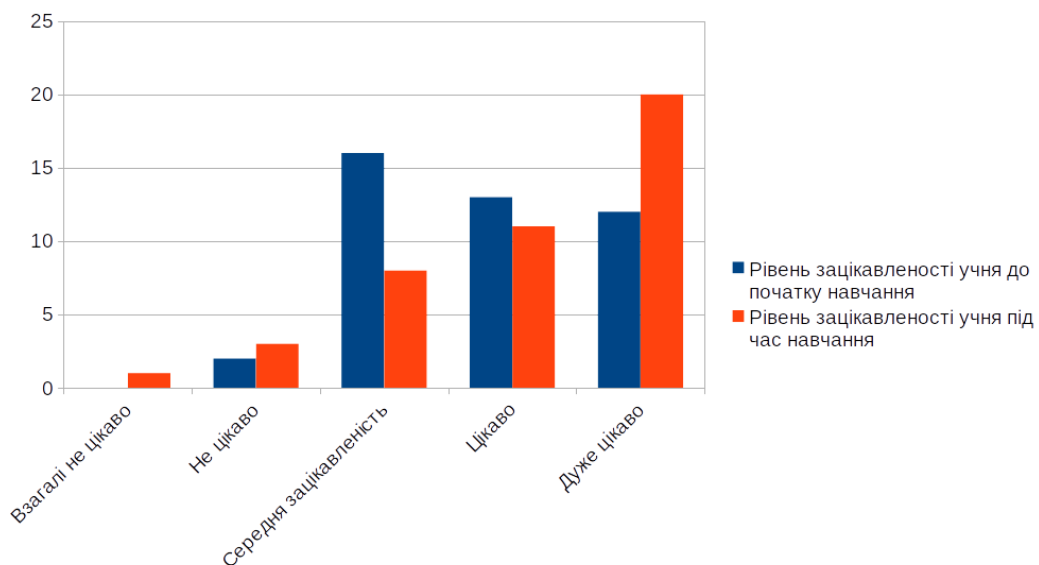


Рис. 7.

На основі отриманих результатів можна зробити такі висновки:

Через сучасні тенденції навчання інформатики в школі доречним є почати навчання основ програмування вже у середній школі. Разом з тим, щоб обійти труднощі у сприйнятті нового, досить складного, навчального матеріалу учнями 6-7 класів, є доречним обирати прості, але практичні мови програмування, а також використовувати різноманітні тренувальні та навчальні середовища програмування. Однією з таких мов може бути Python, а використання розробленого навчально-

тренувального середовища дозволить пришвидшити та полегшити цей процес. Таким чином в 8-му та 9-му класах учні можуть зосередити свою увагу на вивченні вбудованих та сторонніх модулів цієї мови для подальшої розробки різноманітних навчальних та наукових програмних проектів у старшій школі.

Список використаних джерел

- [1] Жалдак М. І. Про деякі методичні аспекти навчання інформатики в школі та педагогічному університеті. Наукові записки Тернопільського національного університету ім. В. Гнатюка. Серія: Педагогіка. 2005. № 6. С. 17-24.
- [2] Жалдак М. І., Рамський Ю. С. Становлення і розвиток методичної системи навчання інформатики в школах і педагогічних університетах України. Науковий часопис НПУ імені М. П. Драгоманова. Серія 2: Комп'ютерно-орієнтовані системи навчання. 2012. № 13. С. 24-41.
- [3] Павелків Р.В. Вікова психологія: підручник. Вид. 2-е, стер. Київ: Кондор, 2015. 469 с.
- [4] 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: 10.01.2021).
- [5] Інструктивно-методичні рекомендації щодо викладання навчальних предметів у закладах загальної середньої освіти у 2020/2021 навчальному році, 2020 рік. URL: <https://mon.gov.ua/storage/app/uploads/public/5f4/cae/d10/5f4caed10f675968632995.pdf> (Дата звернення: 10.01.2021)
- [6] Франчук Н.П. Використання вільного програмного забезпечення під час навчання дисциплін «Методика навчання інформатики» Тези восьмої Міжнародної науково-практичної конференції «FOSS LVIV-2019»: Львів, 18-21 квітня 2019р. Львів: ЛНУ імені Івана Франка, 2019. С. 27-29.
- [7] Медведєв М. Г. Использование интернет - портала E-Olymp для подготовки к олимпиадам по программированию. Актуальні питання сучасної інформатики. 2017. №5. С. 11-13.
- [8] Базурін В.М., Омелечко, Є.А., Ковтун А.В. Порівняльний аналіз середовищ програмування мовою Python. Новітні комп'ютерні технології. 2018. №16. С. 281-292.
- [9] Козолуп Є.В. Програмування в школі. Мова Python: навчальний посібник. 8 клас. Суми, 2017. 82 с.
- [10] Лутц М. Программирование на Python, том I, 4-е издание. / Пер. с англ. СПб.: Символ-Плюс, 2011. 992 с., ил.
- [11] Яковенко А. Основи програмування: методичні вказівки до виконання комп'ютерних практикумів з дисципліни "Основи програмування". Основи програмування мовою Python / А. В. Яковенко. Київ : НТУУ "КПІ ім. І. Сікорського", 2017. 87 с.
- [12] Франчук Н.П. Новітні комп'ютерні технології. Кривий Ріг: Видавничий центр Криворізького національного університету, 2019. Том XVII: спецвипуск «Хмарні технології в освіті». С. 7-11. URL: <https://ccjournals.eu/ojs/index.php/nocote/article/view/937> (дата звернення: 10.12.2020)
- [13] SoloLearn: Learn to Code for Free! URL: <https://www.sololearn.com/> (дата звернення: 10.01.2021)
- [14] Питонтьютор. URL: <https://pythontutor.ru/> (дата звернення: 10.01.2021)
- [15] E-Olymp. URL: <https://www.e-olymp.com/uk/> (дата звернення: 10.01.2021)
- [16] Welcome to Python.org. URL: <https://www.python.org/> (дата звернення: 10.01.2021)
- [17] Thonny, Python IDE for beginners. URL: <https://thonny.org/> (дата звернення: 10.01.2021)
- [18] JDoodle. URL: <https://www.jdoodle.com/> (дата звернення: 10.01.2021)

References

- [1] Zhaldak M. I. (2005) About some methodical aspects of teaching computer science at school and pedagogical university. Scientific notes of Ternopil National University V. Hnatyuk. Series: Pedagogy. **6**. pp. 17-24. (in Ukrainian)
- [2] Zhaldak M. I., Ramsky Yu. S. (2012) Formation and development of the methodical system of teaching computer science in schools and pedagogical universities of Ukraine. Scientific journal of NPU named after M.P. Dragomanov. Series 2: Computer-based learning systems. **13**. pp. 24–41. (in Ukrainian)
- [3] Pavelkiv R. V. (2015) Age psychology: textbook. Pub. 2nd. Kyiv: Condor, 469 p. (in Ukrainian)
- [4] The Curriculum in computer science for grades 5-9, 2017. URL: <https://mon.gov.ua/storage/app/media/zagalna%20serednya/programy-5-9-klas/onovlennya-12-2017/8-informatika.docx> (Access date: 10.01.2021). (in Ukrainian)
- [5] Instructional and methodical recommendations for teaching subjects in general secondary education institutions in the 2020/2021 academic year, 2020. URL: <https://mon.gov.ua/storage/app/uploads/public/5f4/cae/d10/5f4caed10f675968632995.pdf> (Accessed: 10.01.2021) (in Ukrainian)

- [6] Franchuk N. P. (2019) Use of free software during the teaching of disciplines "Methods of teaching computer science" Abstracts of the eighth International scientific-practical conference "FOSS LVIV-2019": Lviv, April 18-21, 2019. Lviv: Ivan Franko Lviv National University, pp. 27-29. (in Ukrainian)
- [7] Medvedev M. G. Using the Internet portal E-Olymp to prepare for programming competitions. Current issues of modern computer science. 2017. **5**. with. 11-13. (in Russian)
- [8] Bazurin V. M., Omelechko E. A., Kovtun A. V. (2018) Comparative analysis of Python programming environments. New computer technology. **16**. p. 281-292. (in Ukrainian)
- [9] Kozolup Ye. V. (2017) Programming at school. Python: tutorial. 8th grade. Sumy, 82 p. (in Ukrainian)
- [10] Lutz M. (2011) Python Programming, Volume **I**, 4th Edition. / Per. with English SPb : Simvol-Plyus, 992 p. (in Russian)
- [11] Yakovenko A. (2017) Fundamentals of programming: guidelines for computer workshops in the discipline "Fundamentals of Programming". Fundamentals of Python programming. Kyiv: NTUU "KPI named after I. Sikorsky", 87 p. (in Ukrainian)
- [12] Franchuk N. P. (2019) New computer technology. Kryvyi Rih: Publishing Center of Kryvyi Rih National University, Volume **XVII**: special ISSUE "Cloud technologies in education". Pp. 7-11. URL: <https://ccjournals.eu/ojs/index.php/nocote/article/view/937> (access date: 10.12.2020) (in Ukrainian)
- [13] SoloLearn: Learn to Code for Free! URL: <https://www.sololearn.com/> (access date: 10.01.2021) (in English)
- [14] Pythontutor. URL: <https://pythontutor.ru/> (access date: 10.01.2021) (in Russian)
- [15] E-Olymp. URL: <https://www.e-olymp.com/uk/> (access date: 10.01.2021) (in English)
- [16] Welcome to Python.org. URL: <https://www.python.org/> (access date: 10.01.2021) (in English)
- [17] Thonny, Python IDE for beginners. URL: <https://thonny.org/> (access date: 10.01.2021) (in English)
- [18] JDoodle. URL: <https://www.jdoodle.com/> (access date: 10.01.2021) (in English)

Kozolup Ye.V.

DEVELOPMENT OF A TRAINING ENVIRONMENT FOR TRAINING THE BASICS OF PROGRAMMING FOR STUDENTS OF 6-7 GRADES OF GENERAL SECONDARY EDUCATION INSTITUTIONS

Abstract. The article considers an important section of the training course of computer science – programming. Programming training develops memory, logical thinking, creativity, self-organization, resilience and intellectual abilities. In this case, the number of hours provided by the curriculum in computer science to study the topic "Algorithms and programs", as well as "Fundamentals of object-oriented programming" can be considered more than sufficient (from 30% to 50% depending on the class). Therefore, with a systematic and balanced approach to training, after school-leaving, each student can have a basic knowledge of programming, he will be able to apply it to solve practical problems arising during various activities. At the same time, a variety of development training environments, as well as educational web resources and sites, can help teachers and students. For such modern and simple languages as Python, there are enough of them. However, unfortunately, the available services are not quite suitable for use in grades 6-7 when studying the topic "Algorithms and programs". That is why it is advisable to develop an easy-to-use but meaningful learning environment for training the basics of Python programming in high school. The PyLearn environment is designed to make it easier for students to adapt to the rules of writing code and testing it. Using this approach allow to be devoted to solving practical problems for the maximum part of study time, which in the future, of course, will allow students to deepen their knowledge and to start working on their own educational or research project on programming in the high school.

The article is focused on a brief analysis of popular software tools for learning programming, as well as the principle of working with the developed learning environment PyLearn in described.

Keywords: object-oriented programming, programming language, software development environment, training development environment.