

Шифр: Usus est optimus magister

**ДИСТАНЦІЙНІ ТЕХНОЛОГІЇ НАВЧАННЯ
ОСНОВ ПРОГРАМУВАННЯ
НА ЗАСАДАХ ІНТЕГРОВАНОГО ПІДХОДУ ЩОДО
РОЗВИТКУ КЛЮЧОВИХ КОМПЕТЕНТНОСТЕЙ**

ЗМІСТ

ВСТУП	3
РОЗДІЛ 1 ТЕОРЕТИЧНІ ОСНОВИ ОРГАНІЗАЦІЇ ДИСТАНЦІЙНОЇ ПІДТРИМКИ НАВЧАННЯ ІНФОРМАТИКИ НА ЗАСАДАХ ІНТЕГРОВАНОГО ПІДХОДУ ЩОДО РОЗВИТКУ КЛЮЧОВИХ КОМПЕТЕНТНОСТЕЙ.....	5
1.1 Ключові компетентності: основні визначення, особливості інтегрованого розвитку на уроках інформатики	5
1.2 Реалізація дистанційної підтримки навчання інформатики	9
РОЗДІЛ 2 ЕЛЕКТРОННИЙ НАВЧАЛЬНИЙ MOODLE-КУРС «ОСНОВИ ПРОГРАМУВАННЯ (ІНТЕГРОВАНІЙ ПІДХІД)»: ЗАГАЛЬНА ХАРАКТЕРИСТИКА ТА МЕТОДИЧНІ РЕКОМЕНДАЦІЇ ЩОДО ВИКОРИСТАННЯ	12
2.1 Зміст та інструментальні засоби навчання основ програмування у шкільній інформатиці	12
2.2 Загальна характеристика та методичні рекомендації до використання електронного навчального Moodle-курсу «Основи програмування (інтегрований підхід)»	15
2.3. Методичні рекомендації щодо роботи з курсом «Основи програмування (розвиток багатомовної компетентності)»	18
ВИСНОВКИ.....	21
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	22
ДОДАТКИ.....	25

ВСТУП

Актуальність теми. Модель будь-якої системи освіти має фіксувати відповіді на ключові питання – «чому навчати?» (зміст навчання), «за допомогою чого навчати?» (засоби навчання) та «як навчати?» (методи навчання). Згідно Концепції Нової української школи [12], новий зміст навчання має забезпечувати наскрізне (інтегроване) формування та удосконалення ключових компетентностей освіти впродовж життя, сучасне освітнє середовище (просторово-предметне оточення, програми та дидактичні засоби) має сприяти ефективному навчанню не лише в приміщенні закладу освіти, а діяльнісно-інтерактивні методи навчання мають набути першочергового значення. Окрім того, сучасне технологічно-цифрове суспільство потребує з одного боку – впевнених користувачів наявних програмних засобів та Інтернет-сервісів, з іншого – проектування та реалізації нових засобів автоматизації різних предметних галузей. За таких умов інформатика та інформаційно-комунікаційні технології не втрачають зайнятих з 2011 року визначальних позицій (2011 рік – рік прийняття нового Державного стандарту базової та повної середньої освіти, згідно якого обов'язкове вивчення інформатики починається з другого класу), не дивлячись на те, що результати її вивчення не підлягають зовнішньому незалежному оцінюванню.

Мета дослідження – дослідити особливості організації дистанційної підтримки навчання учнів інформатики та розробити електронний навчальний курс з основ програмування мовою Python, реалізуючи інтегрований підхід щодо розвитку інформаційно-цифрової та багатомовної компетентностей.

Завдання дослідження:

1. Узагальнити та систематизувати теоретичні відомості щодо інтегрованого розвитку ключових компетентностей на уроках інформатики.
2. Узагальнити та систематизувати теоретичні відомості щодо організації дистанційної підтримки навчання учнів інформатики.
3. Розробити електронний навчальний Moodle-курс з основ програмуван-

ня мовою Python на засадах інтегрованого підходу та сформулювати методичні рекомендації щодо роботи з ним.

Об’єкт дослідження – навчання інформатики учнів основної та старшої шкіл.

Предмет дослідження – методика та засоби реалізації дистанційної підтримки навчання учнів основ програмування мовою Python на засадах інтегрованого підходу щодо розвитку інформаційно-цифрової та багатомовної компетентностей.

Методи дослідження: *теоретичні* – аналіз, узагальнення, систематизація наукових та науково-методичних джерел з проблеми дослідження; *емпіричні* – проектування та розробка електронних навчальних курсів.

Практичне значення одержаних результатів полягає в тому, що автором розроблено електронний навчальний Moodle-курс «Основи програмування (інтегрований підхід)» та упроваджено в освітній процес КДПУ:

– в межах забезпечення неформальної та інформальної освіти учнів 7-11-х класів закладів загальної середньої освіти міста;

– при підготовці майбутніх учителів інформатики за основною та додатковою спеціальностями (окремі змістові модулі дисциплін «Програмування мовою Python», «Методика навчання інформатики» та «Сучасний урок інформатики»).

Апробація результатів дослідження

Shokaliuk S. V. Technologies of distance learning for programming basics on the principles of integrated development of key competences / Svitlana V. Shokaliuk, Yelyzaveta Yu. Bohunenko, Iryna V. Lovianova and Mariya P. Shyshkina // Cloud Technologies in Education : Proceedings of the 7th Workshop on Cloud Technologies in Education (CTE 2019). Kryvyi Rih, Ukraine, December 20, 2019 / Edited by : Arnold E. Kiv, Mariya P. Shyshkina. – P. 548-562. – (CEUR Workshop Proceedings (CEUR-WS.org), Vol. 2643. – Access mode : <http://ceur-ws.org/Vol-2643/paper32.pdf> (збірник індексується у Scopus)

РОЗДІЛ 1

ТЕОРЕТИЧНІ ОСНОВИ ОРГАНІЗАЦІЇ ДИСТАНЦІЙНОЇ ПІДТРИМКИ НАВЧАННЯ ІНФОРМАТИКИ НА ЗАСАДАХ ІНТЕГРОВАНОГО ПІДХОДУ ЩОДО РОЗВИТКУ КЛЮЧОВИХ КОМПЕТЕНТНОСТЕЙ

1.1 Ключові компетентності: основні визначення, особливості інтегрованого розвитку на уроках інформатики

У Законі України «Про освіту» *компетентність* визначається як «динамічна комбінація знань, умінь, навичок, способів мислення, поглядів, цінностей, інших особистих якостей, що визначає здатність особи успішно соціалізуватися, провадити професійну та/або подальшу навчальну діяльність» [7].

Розрізняють предметні (галузеві), міжпредметні та ключові компетентності.

Під *предметною компетентністю* розуміється набутий у процесі навчання досвід специфічної для певного предмета діяльності, пов'язаної із засвоєнням, розумінням і застосуванням нових знань.

Під *міжпредметною компетентністю* розуміється здатність застосовувати щодо міжпредметного кола проблем знання, вміння, навички, способи діяльності та ставлення, які належать до певного кола навчальних предметів (дисциплін) і освітніх галузей.

Під *ключовими* розуміються [7] такі компетентності, що необхідні кожній сучасній людині для успішної життєдіяльності в умовах непередбачуваності та невизначеності, адаптивності до змін, вміння вчитися та перенавчання. До переліку ключових компетентностей відносять:

- вільне володіння державною мовою;
- здатність спілкуватися рідною (у разі відмінності від державної) та іноземними мовами;
- математична компетентність;
- компетентності у галузі природничих наук, техніки і технологій;
- інноваційність;

- екологічна компетентність;
- інформаційно-комунікаційна (нині інформаційно-цифрова компетентність);
- навчання впродовж життя;
- громадянські та соціальні компетентності;
- культурна компетентність;
- підприємливість та фінансова грамотність;
- інші компетентності, передбачені стандартом освіти.

Беручи до уваги предмет та завдання даної роботи більш докладно розглянемо тлумачення (визначення), структуру (компоненти), значущість й особливості формування та/або розвитку при навчанні інформатики трьох ключових компетентностей – інформаційно-цифрової і компетентності щодо здатності спілкуватися рідною та іноземними мовами – багатомовної компетентності.

Інформаційно-цифрова компетентність тлумачиться як така компетентність, що передбачає впевнене, критичне й відповідальне використання цифрових технологій для розвитку і спілкування, здатність безпечного та етичного використання засобів інформаційно-комунікаційної компетентності у навчанні та інших життєвих ситуаціях.

Інформаційно-цифрова компетентність проявляється у:

- розумінні того, як цифрові технології можуть підтримувати спілкування, творчість та інновації; обізнаності щодо можливостей, обмежень, впливів та ризиків цифрових технологій; розумінні загальних принципів, механізмів та логіки розвитку цифрових технологій та знаннях базових функцій та використання різних пристроїв, програмного забезпечення та мереж; критичному ставленні до валідності, надійності та значущості інформації та даних, що надаються за допомогою цифрових засобів; знанні правових та етичних принципів, пов'язаних із використанням цифрових технологій;
- уміннях використання цифрових технологій для підтримки активної громадянської та соціальної інтеграції, співпраці з іншими людьми та творчості для досягнення особистих, соціальних або комерційних цілей; здатності до ви-

користання, доступу, фільтрування, оцінки, створення, програмування та поширення цифрового контенту; вміння управляти та захищати інформацію, контент, дані (включно із особистими), а також розпізнавати та ефективно взаємодіяти з програмним забезпеченням, пристроями, штучним інтелектом або роботами;

– взаємодії з цифровими технологіями та контентом, передбачає рефлексивне та критичне, навіть допитливе, відкрите та перспективне ставлення до їхньої еволюції, а також етичного, безпечного та відповідального підходу до використання цих засобів [3; 14].

Багатомовна компетентність («здатність спілкуватися рідною (у разі відмінності від державної) та іноземними мовами») тлумачиться як така, що передбачає активне використання рідної мови в різних комунікативних ситуаціях, зокрема в побуті, освітньому процесі, культурному житті громади, можливість розуміти висловлювання іноземною мовою, спілкуватися нею у відповідних ситуаціях, оволодіння навичками міжкультурного спілкування [14].

Багатомовна компетентність проявляється у:

– знаннях лексики та функціональної граматики різних мов та усвідомлення основних типів словесної взаємодії та реєстрів мов; знаннях суспільних конвенцій, культурного аспекту та мінливості мов;

– у вмінні розуміти усні повідомлення, ініціювати, підтримувати та завершувати розмови та читати, розуміти та складати тексти з різним рівнем володіння різними мовами відповідно до потреб людини;

– у позитивному ставленні передбачати оцінку культурної різноманітності, зацікавленість та цікавість щодо різних мов та міжкультурного спілкування; у повазі до індивідуального мовного профілю кожної людини, включаючи повагу до рідної мови осіб, що належать до меншин, та/або з міграційним походженням, та оцінювання офіційної мови (мов) країни як загальної основи взаємодії.

На уроках інформатики удосконалюються такі компоненти багатомовної компетентності як:

- уміння використовувати програмні засоби та ресурси з англomовним інтерфейсом;
- уміння використовувати програмні засоби для перекладу текстів та тлумачення іноземних слів;
- уміння оперувати базовою міжнародною ІТ-термінологією.
- усвідомлення ролі ІТ в інтерперсональній комунікації у глобальному контексті;
- розуміння необхідності володіння іноземними мовами для онлайн-навчання й активного залучення до європейської та глобальної спільнот, усвідомлення своєї причетності до них [11].

Очевидно, що інформатика як шкільний предмет відіграє визначальну роль у формуванні та розвитку перш за все компонентів інформаційно-цифрової компетентності. Проте, ефективність розвитку компонентів інформаційно-цифрової компетентності певною мірою залежить від рівня сформованості окремих компонентів багатомовної компетентності. Перш за все сформованість багатомовної компетентності проявляється у відсутності мовного бар'єру при роботі з прикладними та інструментальними програмними засобами з англomовним інтерфейсом. У той же час, система доцільно дiбраних предметних (інформатичних) англomовних задач (зокрема, із математичним змістом) є певним потенціалом інтегрованого удосконалення окремих компонентів багатомовної (та математичної) компетентності.

Інтеграція змісту навчання подвоює (у випадку роботи із системою задач математичного змісту потроює) освітні цілі уроку, інтенсифікує хід уроку. Результативність таких уроків не буде задовільною за якістю та темпами удосконалення ані ключових, ані предметних компетентностей без залученням додаткових засобів. Компенсувати брак урочного часу (на ґрунтовну актуалізацію опорних знань англomовної тематичної лексики (та/або математичних основ), опрацювання інтегрованих міні-конспектiв, розв'язання достатньої кількості навчальних задач, персоналізованого підведення підсумків уроку) допоможуть засоби дистанційних технологій.

1.2 Реалізація дистанційної підтримки навчання інформатики

Беручи до уваги вітчизняний та регіональний досвід забезпечення дистанційної підтримки навчання предметів загальноосвітньої підготовки та інформатично-математичного циклу зокрема у закладах загальної середньої та вищої освіти [1; 4], можна відмітити три основні підходи:

– *перший* підхід базується на представленні освітніх ресурсів (навчально-методичних матеріалів) на сторінках офіційного сайту закладу освіти (чи його структурних підрозділів) або персональних сайтах вчителів (викладачів). Розміщення самих освітніх ресурсів як правило здійснюється у віддалених хмарних сховищах, найчастіше Google Drive (особливо, якщо є можливість використати переваги корпоративного акаунта Google, а саме – нелімітованість хмарного сховища);

– *другий* – базується на представленні освітніх ресурсів у вигляді електронних навчальних курсів, створених та впроваджених в освітній процес за допомогою систем чи сервісів управління навчанням, найпоширенішими з яких на сьогодні визнано сервіс Google Classroom і систему Moodle (або сервіс MoodleCloud);

– *третій* (тільки набуває поширення) – базується на використанні спеціалізованих середовищ та сервісів навчального і наукового призначення, зокрема інформатично-математичної галузі, наприклад, CoCalc (попередня назва SageMathCloud) та Jupyter Notebook.

Найбільшою популярністю на сьогодні користується підхід реалізації дистанційного підтримки навчання, зокрема інформатики, у закладах загальної середньої освіти за допомогою сервісу управління навчанням Google Classroom, а у закладах вищої освіти – системи Moodle.

Серед *основних спільних можливостей* системи Moodle та сервісу Google Classroom для вчителів (викладачів) слід зазначити:

– розміщення та швидка модифікація основних (базових) та допоміжних

навчальних ресурсів у вигляді файлів у різноманітних мультимедійних форматах або URL-посилань;

- розміщення та швидка модифікація *елементів навчальної діяльності* (як *індивідуальної, так і групової*) – перш за все *завдань* (для практичних та лабораторних занять, для самостійної індивідуальної та групової роботи) та *тестів* (форм). До змісту завдань та тестів (й варіантів відповідей на них) можуть бути додані різноманітні мультимедійні фрагменти;

- автоматизований контроль за своєчасністю виконання завдань слухачами та архівне зберігання наданих результатів;

- автоматизована* перевірка результатів тестування (*для системи Moodle – у разі відсутності питань типу «Есе»);

- автоматизація обліку результатів навчання у журналі курсу;

- інструменти для здійснення дистанційного консультування у режимі реального часу – чат тощо;

- автоматизований менеджмент користувачами курсу (співрозробниками (асистентами) та слухачами (учнями або студентами)).

Серед *основних спільних можливостей* системи Moodle та сервісу Google Classroom для *слухачів курсу* (учнів або студентів) слід зазначити:

- мобільний (“завжди та всюди”) доступ до всіх елементів курсу – основних та допоміжних навчальних ресурсів уроку (заняття), завдань для практичного виконання, завдань домашньої (самостійної) роботи, тренувально-навчальних тестів, тощо;

- календар подій та нагадування про кінцеві терміни виконання завдань;

- інструменти для спілкування з вчителем (викладачем) та іншими слухачами курсу;

- постійний доступ до перегляду власних досягнень у курсі.

Очевидно, що ані система Moodle, ані сервіс Google Classroom не є спеціалізованим інструментом для навчання інформатики в цілому та програмування зокрема. Проте відкритість та розширюваність системи Moodle за рахунок нових рішень (модулів, плагінів чи фільтрів), спектр зазначених можливостей та

переваг для вчителів (викладачів) та учнів (студентів) стали визначальними у доборі засобу реалізації дистанційної підтримки навчання учнів та майбутніх учителів основ програмування мовою Python на засадах інтегрованого підходу щодо розвитку ключових компетентностей.

РОЗДІЛ 2

ЕЛЕКТРОННИЙ НАВЧАЛЬНИЙ MOODLE-КУРС «ОСНОВИ ПРОГРАМУВАННЯ (ІНТЕГРОВАНІЙ ПІДХІД)»: ЗАГАЛЬНА ХАРАКТЕРИСТИКА ТА МЕТОДИЧНІ РЕКОМЕНДАЦІЇ ЩОДО ВИКОРИСТАННЯ

2.1 Зміст та інструментальні засоби навчання основ програмування у шкільній інформатиці

Згідно навчальної програми з інформатики для 5-9-х класів, учні яких не вивчали інформатику у 2-4-х класах, початок вивчення основ програмування припадає на 8-й клас із продовженням у 9-у. Право обрати мову програмування та інструментальне середовище залишається за вчителем.

Аналіз рекомендованих МОН підручників з інформатики для 8-х та 9-х класів показав, що у більшості підручників з інформатики (авторські колективи яких очолюють Й. Я. Ривкінд [15; 16], О. О. Бондаренко [5; 6], А. М. Гуржій [8]) предметом вивчення є компільована мова програмування Pascal, а в якості інструментального середовища переважно пропонується Lazarus. І тільки у підручнику [9;10], авторський колектив якого очолює Н. В. Морзе, предметом вивчення є не тільки компільована мова програмування Pascal у середовищі Lazarus, а й інтерпретована мова програмування Python у середовищі PyCharm. У методичних рекомендаціях автори підручника зазначають про різні варіанти організації навчання програмування:

- або однієї із запропонованих мов (знов-таки на вибір вчителя);
- або, за умови готовності учнів, паралельного вивчення обох.

Для учнів, які вивчали інформатику у початковій школі (див. табл. 2.1), початок вивчення основ програмування текстовою мовою має, в обов'язковому порядку, припадати на 7-й клас (вперше для 7-х класів у навчальному році 2018-2019).

Таблиця 2.1

Витяг із навчальної програми

«Інформатика для 5-9-х класів» (2017)

Очікувані результати навчально-пізнавальної діяльності учнів	Зміст навчального матеріалу
7 клас	
<p>Учень/учениця Знаннєва складова <i>Пояснює</i> поняття величини, змінної та операції присвоювання. <i>Знає</i> базові алгоритми роботи зі змінними: обмін значеннями, визначення найбільшого й найменшого з двох значень</p> <p>Діяльнісна складова <i>Використовує</i> різні алгоритмічні структури та змінні для розв'язання навчальних і життєвих задач. <i>Застосовує</i> засоби програмування для побудови моделей</p> <p>Ціннісна складова <i>Усвідомлює</i> роль програмування та моделювання для розв'язання навчальних і життєвих задач</p>	<p>Величини. Змінні. Вказівка присвоювання.</p> <p>Створення алгоритмів і програм з використанням змінних і різних алгоритмічних структур: лінійних, розгалужень і повторень.</p> <p>Опис моделей у середовищі програмування</p>
8 клас	
<p>Учень/учениця Знаннєва складова <i>Розуміє</i> призначення мови програмування та основних її елементів. Наводить приклади сучасних мов програмування. <i>Знає</i> відмінність між змінними та константами. <i>Порівнює</i> особливості різних середовищ програмування. <i>Розуміє</i> поняття об'єкта в мові програмування, його властивостей і методів. <i>Пояснює</i> структуру програми. <i>Пояснює</i> функції елементів графічного інтерфейсу та користується ними. <i>Розрізняє</i> властивості і методи елементів управління</p> <p>Діяльнісна складова <i>Планує</i> процес розв'язування задачі з використанням програмування. <i>Створює і налагоджує</i> програми, зокрема подійно- й об'єктно-орієнтовані. <i>Використовує</i> в програмах вирази, коректно добирає типи даних. <i>Розв'язує</i> задачі з використанням усіх базових</p>	<p>Сучасні мови програмування.</p> <p>Поняття об'єкта в мові програмування, його властивостей і методів. Графічний інтерфейс, основні компоненти програми з графічним інтерфейсом. Поняття елемента керування. Обробники подій, пов'язаних з елементами керування. Властивості та методи елементів керування.</p> <p>Типи даних у програмуванні. Структура програми. Введення й виведення даних. Вирази. Логічні вирази та змінні й операції над ними. Умовні оператори (коротка та повна форма). Складені умови. Оператори циклу. Вкладені цикли. Пошук найбільшого та найменшого серед кількох значень</p>

Очікувані результати навчально-пізнавальної діяльності учнів	Зміст навчального матеріалу
<p>алгоритмічних структур, змінних та констант. <i>Обґрунтовує</i> вибір типів даних для розв'язування задачі</p> <p>Ціннісна складова <i>Оцінює</i> відповідність результатів виконання програми поставленій задачі. <i>Розпізнає</i> задачі, для розв'язання яких доцільно використовувати засоби програмування</p>	
9 клас	
<p>Учень/учениця Знаннєва складова <i>Пояснює</i> принцип організації даних за допомогою одновимірних масивів. <i>Пояснює</i> поняття масиву, елемента масиву, індексу та значення елемента. <i>Описує</i> алгоритми опрацювання елементів масиву, що задовольняють певній умові. <i>Описує</i> алгоритм знаходження підсумкових величин у масиві. <i>Описує</i> принаймні один алгоритм впорядкування масиву</p> <p>Діяльнісна складова <i>Складає й описує</i> мовою програмування алгоритми для опрацювання елементів масиву, що задовольняють певну умову, знаходження підсумкових величин у масиві та його впорядкування</p> <p>Ціннісна складова <i>Оцінює</i> часову та ємнісну складність алгоритмів. <i>Усвідомлює</i> важливість застосування ефективних методів для опрацювання великих наборів даних</p>	<p>Поняття одновимірного масиву. Введення й виведення значень елементів масиву.</p> <p>Алгоритми опрацювання масивів: знаходження підсумкових величин, зокрема для елементів, що задовольняють задані умови, а також пошук у масиві за певними критеріями.</p> <p>Алгоритми впорядкування масиву.</p> <p>Поняття складності алгоритмів</p>

Зміст підручників з інформатики для 7 класу 2015 року видання (що забезпечували освітній процес до появи актуальних підручників для 7 класу у 2020 році) не містить навчальних матеріалів з даної теми – вчителю надано повну свободу щодо використання навчальних матеріалів для виконання програми. Разом із тим, за підручниками з інформатики для 5-го та 6-го класів колективу авторів, який очолює О. Бондаренко (роки видання відповідно – 2018 та

2019), замість рекомендованого Scratch пропонується вивчення основ програмування мовою Python із продовженням у підручнику для 7-го класу (2020 року видання). Більш того, за рекомендованими МОН підручниками за програмою з інформатики профільного рівня для 10-11-х класів предметом вивчення є мова програмування Python ([17; 18]).

Отже, аналіз змісту підручників показав, що мова програмування Python набуває ролі основної, а не альтернативної, мови програмування на етапі здобування середньої освіти. Проблема недостатньої дидактичної підтримки її вивчення, перш за все у 7-у класі (станом на 1 вересня 2019 року), може бути вирішена шляхом упровадження матеріалів авторської розробки – електронного навчального Moodle-курсу «Основи програмування (інтегрований підхід)»

2.2 Загальна характеристика та методичні рекомендації до використання електронного навчального Moodle-курсу «Основи програмування (інтегрований підхід)»

Курс «Основи програмування (інтегрований підхід)» розроблено у системі управління навчанням.

Moodle (абревіатура для модульного об'єктно-орієнтованого динамічного навчального середовища) – це навчальна платформа, розроблена Мартіном Дугіамасом, спрямована на залучення вчителів, адміністраторів та учнів (студентів) разом в одну надійну, безпечну та інтегровану систему створення персоналізованого навчального середовища; вільно поширювана, відкрита, розширювана система управління навчанням, що реалізує філософію „педагогіки соціального конструктивізму” (Ернст фон Глазерсфельд, Сеймур Паперт, Жан Піаже, Ілля О. Теплицький, Лев С. Виготський) і орієнтована насамперед на організацію взаємодії викладача з учнями, хоча може бути використана для організації традиційних дистанційних курсів, а також підтримки академічного навчання (навчання за денною формою); реалізована мовою PHP із використанням баз даних SQL (MySQL, PostgreSQL або Microsoft SQL Server) відповідає стан-

дартам SCORM; перекладена на десятки мов, у тому числі українську, і використовується у понад 190 країнах.

Курс «Основи програмування (інтегрований підхід)» (загальна кількість годин – 15, див. у додатку А рис. А.1) має модульно-тематичну структуру, що відповідає змісту програми вивчення основ структурного програмування текстовою мовою, а саме:

Тема 1. Вступ до програмування – 1 година.

Тема 2. Побудова та реалізація лінійних алгоритмів – 3 години.

Тема 3. Побудова та реалізація розгалужених алгоритмів – 3 години.

Тема 4. Побудова та реалізація циклічних алгоритмів: цикли з передумовою – 3 годин.

Тема 5. Побудова та реалізація циклічних алгоритмів: цикли з лічильником – 3 години.

Тема 6. Побудова та реалізація комбінованих алгоритмів – 2 години.

Основними елементами кожного тематичного модуля є:

Основними елементами кожного тематичного модуля є:

1) для набуття та удосконалення предметної компетентності у програмуванні:

– Moodle-сторінки з новими теоретичними відомостями (рис. А.2);

– Moodle-завдання для практичного виконання (рис. А.3);

– тренувальні Moodle-тести з основ програмування (рис. А.4);

2) для удосконалення багатомовної компетентності:

– тренувальні Moodle-тести англійського змісту (рис. А.5);

3) для комплексної перевірки результатів навчання:

– контрольні комплексні Moodle-тести з основ програмування та англійської мови.

Визначальною особливістю курсу, окрім його інтегрованого змісту, слід відмітити реалізацію інтерактивно-діяльнісного підходу.

По-перше, починаючи з етапу опанування нових теоретичних відомостей, слухачі курсу, не звертаючись до локальних засобів (встановлених інтегрова-

них середовищ програмування) чи інших Інтернет-сервісів (браузерних систем чи середовищ програмування), а безпосередньо на Moodle-сторінці із новими теоретичними відомостями, мають можливість перевірити виконуваність запропонованого програмного коду, за потреби внести правки до нього і знову виконати, спостерігаючи за результатами оновлень.

Можливість виконувати програмний код мовою Python у браузері безпосередньо, зокрема на Moodle-сторінці або на сторінці Moodle-завдання, реалізовано за рахунок додавання скрипта командної комірочки мережної системи комп'ютерної математики SageMath [13; 2]:

```
<script
src="https://sagecell.sagemath.org/static/embedded_sagecell.js">
</script>

<script>$(function()
{
  sagecell.makeSagecell({inputLocation: 'div.compute',
                        languages: ["python"],
                        evalButtonText: 'Виконати'});
});
</script>

...

<div class="compute">
<script type="text/x-sage">
import random
x=random.randint(0,10)
print("x=",x)
</script>
</div>
```

На етапі практичного застосування набутих теоретичних знань та розвитку компетентностей у програмуванні інтерактивно-діяльнісний підхід реалізовано через необхідність виконання практичних завдань на сторінках робочих зошитів, що являють собою зовнішні (відносно Moodle) цифрові освітні ресурси – колекція аркушів блокнотів Jupyter Notebook (рис. А.6).

Нашим слухачам робочі зошити пропонуються як завдання CoCalc-курсів [13], а також як їх резервні копії (ірунв-файли) для роботи в офлайн-режимі.

На етапі перевірки результатів навчання інтерактивно-діяльнісний підхід реалізовано за рахунок розробки системи тестів, до якої (окрім традиційних тестових питань – питань типу «множиний вибір», «відповідність», «коротка відповідь») включено тестові питання типу *CodeRunner*.

Тестові питання типу *CodeRunner* (рис. А.7) є традиційними питаннями курсів з програмування, де слухачам пропонується записати програмний код до якоїсь специфікації, і цей код потім оцінюється за результатами виконання/проходження серії тестів.

2.3. Методичні рекомендації щодо роботи з курсом «Основи програмування (розвиток багатомовної компетентності)»

Для початку роботи з Moodle-курсом потрібно відновити його резервну копію (пропонутся на GitHub за покликанням [19]).

Другою частиною курсу є інтеграція з курсом на CoCalc, який дозволяє учням виконувати практичні завдання на сторінках електронного робочого зошиту – аркушах Jupyter Notebook, а вчителю контролювати виконання завдань в реальному часі. Zip-архів аркушів робочого зошиту також пропонується на GitHub [19].

У проєкті CoCalc потрібно:

- 1) створити папку (наприклад «Робочий зошит», рис. 2.1) та завантажити в них розархівовані аркуші – ірпnb-файли;
- 2) створити файл типу «Manage a course» (результат представлений на рис. 2.1);
- 3) зарахувати на створений курс слухачів на вкладці «Students» (слухачам рекомендується попередньо зареєструватися у CoCalc), див. рис. 2.2;
- 4) на вкладці курсу «Assignments» визначити спільні ресурси курсу – папку (чи папки для послідовного призначення) з аркушами Jupyter Notebook – та поділитися із електронними робочими зошитами з усіма слухачами (натиснути на кнопку «Assign...», рис. 2.2).

	Type	Name
<input type="checkbox"/>		Programming and English language.course
<input type="checkbox"/>		Робочий зошит

Рис. 2.1. Вміст кореневого каталогу (домашньої папки) проєкту CoCalc
















Assignment Name	Due Date ▼	
Student	1. Assign to Student	2. Collect from Student
Nastya Borzenko	 Assign...  Open 16 minutes ago	 Collect
Nikita Kipen	 Assign...  Open 16 minutes ago	 Collect
Yana Manuilova	 Assign...  Open 23 minutes ago	 Collect
Svetlana Shokalyuk	 Assign...  Open 13 minutes ago	 Collect
Илья Харченко	 Assign...  Open 23 minutes ago	 Collect

Рис. 2.2. Вкладка «Students» після додавання слухачів

Першу експериментальну версію курсу «Основи програмування (інтегрований підхід)» упроваджено в освітній процес КДПУ:

– в межах забезпечення неформальної та інформальної освіти учнів 9-10-х класів закладів загальної середньої освіти міста;

– в межах підготовки здобувачів кваліфікації «вчитель інформатики».

Експериментальна група з 8 учнів 9-10-х класів за власним бажанням (мотивовано) успішно завершила опанування курсу. Упровадження здійснювалося у вільний від основного навчання час – після уроків або у вихідні дні. Неперервність та очікувану ефективність освітнього процесу забезпечено через комбінування аудиторних та дистанційних форм роботи.

Майбутні учителі інформатики – 39 магістрів першого року навчання із додатковою спеціальністю 014 Середня освіта (Інформатика), опановують зміст експериментальної версії курсу в межах варіативної складової дисципліни «Су-

часний урок інформатики».

Елементи представленої методики запроваджено у процес опанування дисциплін «Програмування мовою » та «Методика навчання інформатики» здобувачами спеціальності 014 Середня освіта (Інформатика) як основної.

Результати виконання навчальних тестів та періодичних опитувань (на наявність помилок та невідповідностей, зрозумілості контенту та зручності програм-них засобів та сервісів тощо) і учнів, і майбутніх учителів інформатики будуть враховані при проєктуванні удосконаленої версії курсу «Основи програмування (інтегрований підхід)». Після чого, разом із методичним рекомендаціями до курсу, буде розпочато його масштабне запровадження.

ВИСНОВКИ

Інформатика як шкільний предмет відіграє визначальну роль у формуванні інформаційно-цифрової компетентності як предметної, так і ключової. Потенціалом інтегрованого удосконалення окремих компонентів багатомовної компетентності на уроках інформатики є система доцільно дібраних предметних (інформатичних) задач із англомовним змістом.

Нестача урочного часу для успішної реалізації ускладнених освітніх цілей уроку – ефективне засвоєння нових інформатичних знань та набуття відповідних компетентностей, англомовної тематичної лексики – може бути компенсоване за рахунок побудови освітнього процесу із залученням дистанційних технологій.

На підтримку ґрунтовного вивчення основ програмування текстовою мовою, зважаючи на значущість змістової лінії «Основи алгоритмізації та програмування» у шкільному курсі інформатики, було спроектовано та впроваджено в освітній процес дистанційний Moodle-курс «Основи програмування (інтегрований підхід)».

Успіхам експериментального навчання учнів та майбутніх учителів за програмою інтегрованого курсу сприяла його побудова на засадах інтерактивно-діяльнісного підходу, а саме:

- інтерактивне засвоєння нових теоретичних відомостей з програмування (за рахунок виконання вихідних кодів безпосередньо на Moodle-сторінці);
- інтерактивне удосконалення англомовної тематичної лексики (за рахунок роботи з тестовими завданнями у режимі тренування без обмежень у кількості спроб)
- інтерактивне набуття практичних умінь у програмуванні (за рахунок виконання системи навчальних вправ на сторінках електронного робочого зошиту).

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Bondarenko, O.V., Mantulenko, S.V., Pikilnyak, A.V. Google Classroom as a Tool of Support of Blended Learning for Geography Students. In: Kiv, A.E., Soloviev, V.N. (eds.) Augmented Reality in Education, Proceedings of the 1st International Workshop (AREdu 2018), Kryvyi Rih, Ukraine, 2 October 2018. CEUR Workshop Proceedings (CEUR-WS.org), vol. 2257, pp. 182-191
2. Sage Cell Server [Electronic resource] / SageMathCell. – Access mode : <https://sagecell.sagemath.org/>
3. Council Recommendation of 22 May 2018 on key competences for lifelong learning (Text with EEA relevance) [Electronic resource] / The Council of the European Union // Official Journal of the European Union. – С 189 – P. C1-13. – 4.6.2018. – Access mode : [https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32018H0604\(01\)&from=EN](https://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32018H0604(01)&from=EN)
4. Viktoriia O. Ustinova, Svitlana V. Shokaliuk, Iryna S. Mintii and Andrey V. Pikilnyak Modern techniques of organizing computer support for future teachers' independent work in German language // Cloud Technologies in Education : Proceedings of the 6th Workshop on Cloud Technologies in Education (CTE 2018). Kryvyi Rih, Ukraine, December 21, 2018 / Edited by : Arnold E. Kiv, Vladimir N. Soloviev. – P. 308-321. – (CEUR Workshop Proceedings (CEUR-WS.org), Vol. 2433.
5. Бондаренко О. О. Інформатика : підруч. для 8 кл. загальноосвіт. навч. закладів / О. О. Бондаренко, В. В Ластовецький, О. П. Пилипчук, Є. А. Шестапалов. – Харків : Ранок, 2016. – 256 с.
6. Бондаренко О. О. Інформатика : підруч. для 9 кл. загальноосвіт. навч. закладів / О. О. Бондаренко, В. В Ластовецький, О. П. Пилипчук, Є. А. Шестапалов. – Харків : Ранок, 2017. – 261 с.

7. Закон України «Про освіту» [Електронний ресурс]. – Режим доступу : <http://zakon5.rada.gov.ua/laws/show/2145-19>
8. Інформатика : підруч. для 8 кл. загальноосвіт. навч. закладів / А. М. Гуржій, Л. А. Карташова, В. В. Лапінський, В. Д. Руденко. – Львів : Світ, 2016. – 256 с.
9. Морзе Н. В. Інформатика : підруч. для 8 кл. загальноосвіт. навч. закладів / Н. В. Морзе, О. В. Барна, В. П. Вембер. – К. : Оріон, 2016. – 240 с.
10. Морзе Н. В. Інформатика : підруч. для 9 кл. загальноосвіт. навч. закладів / Н. В. Морзе, О. В. Барна, В. П. Вембер. – К. : Оріон, 2017. – 222 с.
11. Навчальна програма з інформатики для учнів 5-9 класів, які вивчали інформатику у 2-4 класах [Електронний ресурс]. – Режим доступу : <https://mon.gov.ua/storage/app/media/zagalna%20serednya/programy-5-9-klas/onovlennya-12-2017/8-informatika.docx>
12. Нова українська школа. Концептуальні засади реформування середньої школи [Електронний ресурс]. – Режим доступу : <https://mon.gov.ua/storage/app/media/zagalna%20serednya/nova-ukrainska-shkola-compressed.pdf>
13. Попель М. В. Організація навчання математичних дисциплін у SageMathCloud: навчальний посібник / М. В. Попель. – 2-ге вид., виправлене // Теорія та методика навчання математики, фізики, інформатики. – Кривий Ріг : Видавничий відділ ДВНЗ «Криворізький національний університет», 2016. – Том XIV. – Випуск 1 (38) : спецвипуск «Навчальний посібник у журналі». – 111 с.
14. Проект Державного стандарту базової середньої освіти [Електронний ресурс]. – Режим доступу : <https://mon.gov.ua/ua/news/ministerstvo-osviti-i-nauki-ukrayini-proponuye-dlya-gromadskogo-obgovorennya-proyekt-derzhavnogo->

[standartu-bazovoyi-serednoyi-osviti](#)

15. Ривкінд Й. Я. Інформатика : підруч. для 8 кл. загальноосвіт. навч. закладів / Й. Я. Ривкінд, Т. І. Лисенко, Л. А. Чернікова, В. В. Шакоцько. – К. : Генеза, 2016. – 288 с.
16. Ривкінд Й. Я. Інформатика : підруч. для 9 кл. загальноосвіт. навч. закладів / Й. Я. Ривкінд, Т. І. Лисенко, Л. А. Чернікова, В. В. Шакоцько. – К. : Генеза, 2017. – 288 с.
17. Руденко В. Д. Інформатика (профільний рівень) : підруч. для 10 класу закладів загальної середньої освіти / В. Д. Руденко, Н. В. Речич, В. О. Потієнко. – Х. : Ранок, 2019. – 256 с.
18. Руденко В. Д. Інформатика (профільний рівень) : підруч. для 11 класу закладів загальної середньої освіти / В. Д. Руденко, Н. В. Речич, В. О. Потієнко. – Х. : Ранок, 2019. – 256 с.
19. LizaBohunenko/Integrat-English [Electronic resource]. – Access mode: <https://github.com/LizaBohunenko/Integrat-English>

ДОДАТКИ

Додаток А

Фрагменти-ілюстрації сторінок курсу Moodle “Основи програмування (інтегрований підхід)”

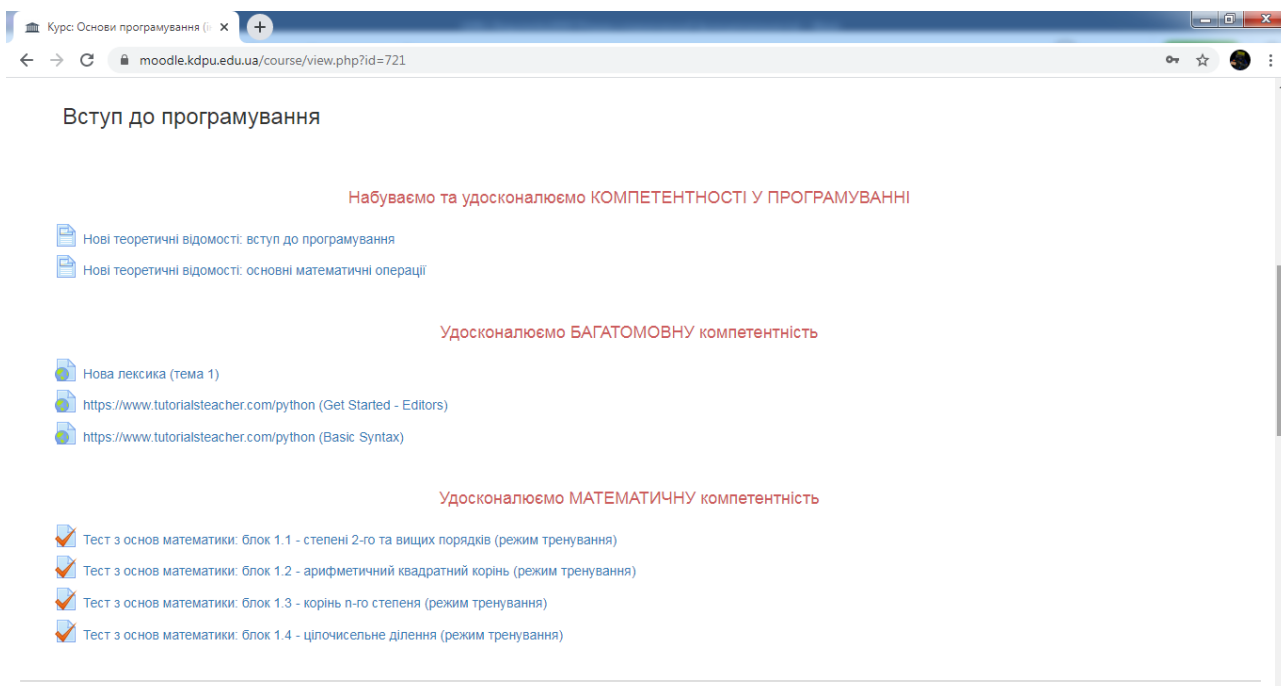


Рис. А.1. Головна сторінка Moodle-курсу «Основи програмування (інтегрований підхід)»

Окремий інформаційний об'єкт (число, символ, рядок тощо) називають величиною. Основними характеристиками величини є назва, вид, тип і значення.

Вид величини визначає спосіб використання величини в програмі. Величина може бути константою (тобто постійною) або змінною.

Константи — це величини, значення яких не можуть змінюватися в ході виконання програми. Прикладом константи може бути число (5, 1.23) або рядок: «Це рядок!».

Змінні — це величини, значення яких можуть змінюватися в ході виконання програми.

З'ясуємо правила, яких необхідно дотримуватися під час надання імен змінним.

- Першим символом імені має бути літера чи знак нижнього підкреслювання `_`.
- Решта імені може складатися з літер, чисел або знаків нижнього підкреслювання.
- Не можна використовувати спеціальні символи, такі як `/`, `#` або `@`. Не можна використовувати пробіли.
- Імена змінних чутливі до регістру. Наприклад, `myname` і `myName` — це різні змінні.
- Не можна надавати змінним імена команд, наприклад `print`.

Команда `input()` призначена для введення даних із клавіатури. Коли програма зустрічає команду `input`, вона припиняє роботу й очікує, поки користувач уведе дані й натисне `Enter`. У дужках може бути записана підказка користувачеві, що саме потрібно ввести. Ця підказка виводиться на екран.

Приклад 1. Виведення повідомлення - функція `print()`

```
1 print("Hello, World")
```

Виконати

Приклад 2. Виведення повідомлення - функція `print()`

```
1 print ("Hello, World") #це - простий коментар
```

Виконати

Приклад 3. Виведення повідомлення - функція `print()`

Рис. А.2. Фрагмент Moodle-сторінки нових теоретичних відомостей

Практичне завдання з вступу до програмування

Посилання на CoCalc: <https://cocalc.com>

Виконайте завдання та відправте файл на перевірку

Chapter_1.ipynb

1 червня 2020, 5:38

Статус роботи

Статус роботи	Немає спроб
Статус оцінення	Не оцінено
Востаннє змінено	-

Коментарі до
відповідей

▶ Коментарі (0)

Здати роботу

You have not made a submission yet.

Рис. А.3. Фрагмент Moodle-завдання для практичного виконання

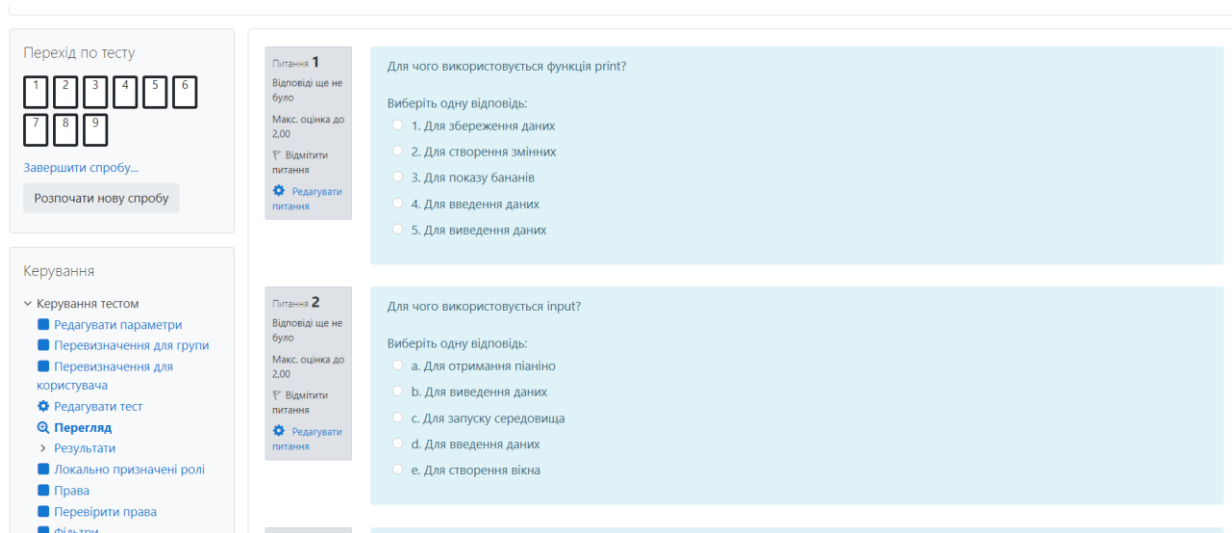


Рис. А.4. Фрагмент сторінки тренувальних Moodle-тестів загального виду з програмування

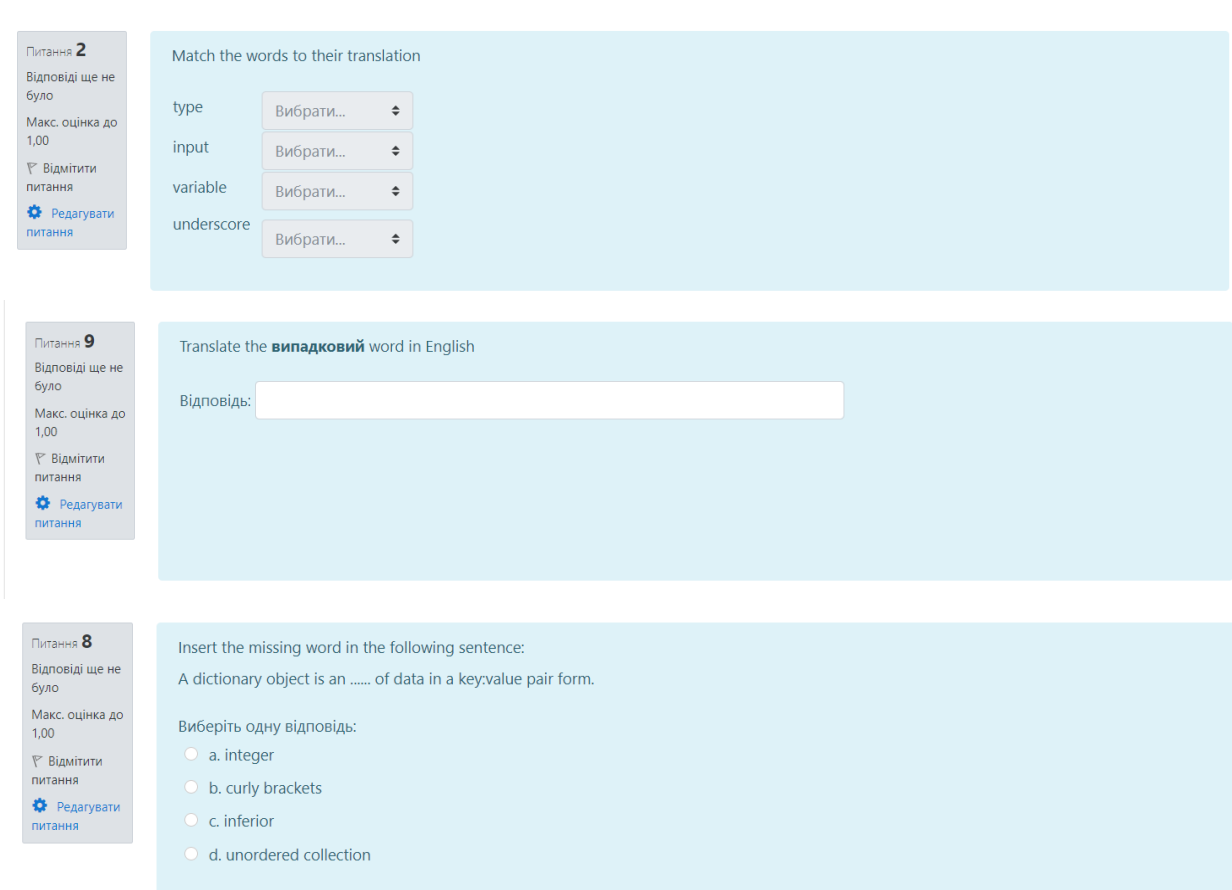


Рис. А.5. Фрагменти сторінок тренувальних Moodle-тестів для англійської мови

Завдання №1.
Вивести на екран 3 записи в яких вказати інформацію про ваших однокласників

In []: `#Рішення для завдання №1`

Завдання №2.
Доповнити завдання №1, тип, що зберегти всі дані в окремі змінні (наприклад, ім'я в змінну name, вік в змінну age)

In []: `#Рішення для завдання №2`

Завдання №3.
Створити змінні в яких вказати вулицю на якій проживаєте та вік. Додати однорядковий коментарі до кожної змінної з описом призначення змінної

In []: `#Рішення для завдання №3`

Завдання №4.
Вивести окремо суму, різницю та добуток двох довільних чисел на екран

In []: `#Рішення для завдання №4`

Завдання №5.
Доповнити завдання №4 багаторядковим коментарем на самому початку, та описати функціонал розробленої програми

Рис. А.6. Фрагмент сторінки робочого зошиту з основ програмування мовою Python

Question 1
Correct
Mark 1.00 out of 1.00
Flag question

Write a Python3 function `sqr(n)` that returns the square of its numeric parameter `n`.

For example:

Test	Result
<code>print(sqr(-3))</code>	9
<code>print(sqr(11))</code>	121

Answer:

```

1 def sqr(n):
2     return n * n

```

Check

Test	Expected	Got	
<code>print(sqr(-3))</code>	9	9	✓
<code>print(sqr(11))</code>	121	121	✓
<code>print(sqr(-4))</code>	16	16	✓
<code>print(sqr(0))</code>	0	0	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Рис. А.7. Фрагмент сторінки тренувальних Moodle-тестів з програмування (питання типу *CodeRunner*)

АНОТАЦІЯ

Актуальність теми. Модель будь-якої системи освіти має фіксувати відповіді на ключові питання – «чому навчати?» (зміст навчання), «за допомогою чого навчати?» (засоби навчання) та «як навчати?» (методи навчання). Згідно Концепції Нової української школи, новий зміст навчання має забезпечувати наскрізне (інтегроване) формування та удосконалення ключових компетентностей освіти впродовж життя, сучасне освітнє середовище (просторово-предметне оточення, програми та дидактичні засоби) має сприяти ефективному навчанню не лише в приміщенні закладу освіти, а діяльнісно-інтерактивні методи навчання мають набути першочергового значення. Окрім того, сучасне технологічно-цифрове суспільство потребує з одного боку – впевнених користувачів наявних програмних засобів та Інтернет-сервісів, з іншого – проєктування та реалізації нових засобів автоматизації різних предметних галузей. За таких умов інформатика та інформаційно-комунікаційні технології не втрачають зайнятих з 2011 року визначальних позицій (2011 рік – рік прийняття нового Державного стандарту базової та повної середньої освіти, згідно якого обов’язкове вивчення інформатики починається з другого класу), не дивлячись на те, що результати її вивчення не підлягають зовнішньому незалежному оцінюванню. Це обумовлює актуальність теми конкурсної роботи «Дистанційні технології навчання основ програмування на засадах інтегрованого підходу щодо розвитку ключових компетентностей».

Мета дослідження – дослідити особливості організації дистанційної підтримки навчання учнів інформатики та розробити електронний навчальний курс з основ програмування мовою Python, реалізуючи інтегрований підхід щодо розвитку інформаційно-цифрової та багатомовної компетентностей.

Завдання дослідження:

1. Узагальнити та систематизувати теоретичні відомості щодо інтегрованого розвитку ключових компетентностей на уроках інформатики.
2. Узагальнити та систематизувати теоретичні відомості щодо організації дистанційної підтримки навчання учнів інформатики.

3. Розробити електронний навчальний Moodle-курс з основ програмування мовою Python на засадах інтегрованого підходу та сформулювати методичні рекомендації щодо роботи з ним.

Методи дослідження: *теоретичні* – аналіз, узагальнення, систематизація наукових та науково-методичних джерел з проблеми дослідження; *емпіричні* – проектування та розробка електронних навчальних курсів.

Загальна характеристика роботи.

У *першому розділі* узагальнено та систематизовано теоретичні відомості щодо інтегрованого розвитку на уроках інформатики з дистанційною підтримкою інформаційно-цифрової та багатомовної компетентностей як ключових; відмічено переваги здійснення для вчителів (викладачів) та учнів (студентів) дистанційної підтримки за допомогою системи та сервісу управління навчанням Moodle та Google Classroom відповідно.

У *другому розділі* наведено загальну характеристику авторського електронного навчального Moodle-курсу “Основи програмування (інтегрований підхід)” – структуру та зміст, особливості реалізації та методики використання.

Практичне значення одержаних результатів полягає в тому, що автором розроблено електронний навчальний Moodle-курс «Основи програмування (інтегрований підхід)» та упроваджено в освітній процес КДПУ:

- в межах забезпечення неформальної та інформальної освіти учнів 9-10-х класів закладів загальної середньої освіти міста;

- при підготовці майбутніх учителів інформатики за основною та додатковою спеціальностями (окремі змістові модулі дисциплін «Програмування мовою Python», «Методика навчання інформатики» та «Сучасний урок інформатики»).

Апробація результатів дослідження: Shokaliuk S. V. Technologies of distance learning for programming basics on the principles of integrated development of key competences / Svitlana V. Shokaliuk, Yelyzaveta Yu. Bohunenko, Iryna V. Lovianova and Mariya P. Shyshkina // Cloud Technologies in Education : Proceedings of the 7th Workshop on Cloud Technologies in Education (CTE 2019).

Kyvyi Rih, Ukraine, December 20, 2019 / Edited by : Arnold E. Kiv, Mariya P. Shyshkina. – P. 548-562. – (CEUR Workshop Proceedings (CEUR-WS.org), Vol. 2643. – Access mode : <http://ceur-ws.org/Vol-2643/paper32.pdf> (збірник індексується у Scopus)