

ПОЛТАВСЬКИЙ УНІВЕРСИТЕТ ЕКОНОМІКИ І ТОРГІВЛІ
Навчально-науковий інститут денної освіти
Форма навчання денна
Кафедра комп'ютерних наук та інформаційних технологій

Допускається до захисту
Завідувач кафедри
_____ Олена ОЛЬХОВСЬКА
(підпис)

« ____ » _____ 2023 р.

КВАЛІФІКАЦІЙНА РОБОТА

на тему
«РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ТРЕНАЖЕРУ З ТЕМИ
«АНАЛІТИЧНА ГЕОМЕТРІЯ НА ПЛОЩИНІ» ДИСТАНЦІЙНОГО
НАВЧАЛЬНОГО КУРСУ «ПРИКЛАДНА МАТЕМАТИКА»

зі спеціальності 122 Комп'ютерні науки
освітня програма «Комп'ютерні науки»
ступеня бакалавра

Виконавець роботи Кизименко Сергій Анатолійович
_____ « ____ » _____ 2023 р.
(підпис)

Науковий керівник к.пед.н., доцент, Кошова Оксана Петрівна
_____ « ____ » _____ 2023 р.
(підпис)

Рецензент

ПОЛТАВА 2023 р.

ЗАТВЕРДЖУЮЗавідувач кафедри _____ Олена ОЛЬХОВСЬКА
(підпис)

«___» _____ 202__ р.

**ЗАВДАННЯ І КАЛЕНДАРНИЙ ГРАФІК
ВИКОНАННЯ КВАЛІФІКАЦІЙНОЇ РОБОТИ**

на тему «Розробка програмного забезпечення тренажеру з теми «Аналітична геометрія на площині» дистанційного навчального курсу «Прикладна математика»

зі спеціальності 122 Комп'ютерні науки
освітня програма «Комп'ютерні науки»
ступеня бакалавр

Прізвище, ім'я, по батькові Кизименко Сергій Анатолійович

Затверджена наказом ректора № 144-Н від «01» вересня 2022 р.

Термін подання студентом роботи «___» _____ 202__ р.

Вихідні дані до кваліфікаційної роботи: публікації з теми, системи дистанційного навчання.

Зміст пояснювальної записки (перелік питань, які потрібно розробити)

ВСТУП

1. РОЗДІЛ 1. ПОСТАНОВКА ЗАДАЧІ

2. РОЗДІЛ 2. ОГЛЯД СИСТЕМ АНАЛОГІЧНОГО ПРИЗНАЧЕННЯ

2.1. Платформа Classtime

2.2. Платформа На Урок

2.3. Платформа Kahoot!

2.4 Платформа Moodle

2.5 Платформа Claroline LMS

3. РОЗДІЛ 3. ТЕОРЕТИЧНА ЧАСТИНА

3.1. Застосування аналітичної геометрії на площині

3.2. Алгоритм роботи тренажеру з теми «Аналітична геометрія на площині»
дистанційного навчального курсу «Прикладна математика»

4. РОЗДІЛ 4. ПРАКТИЧЕНА ЧАСТИНА

4.1. Блок-схема тренажеру

4.2. Опис роботи тренажеру

ВИСНОВКИ

СПИСОК ЛІТЕРАТУРИ

ДОДАТОК А. ВИХІДНІ КОДИ

Консультанти розділів кваліфікаційної роботи

Розділ	Прізвище, ініціали, посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Постановка задачі	Кошова О.П.		
Інформаційний огляд	Кошова О.П.		
Теоретична частина	Кошова О.П.		
Практична частина	Кошова О.П.		

Календарний графік виконання кваліфікаційної роботи

Зміст роботи	Термін виконання	Фактичне виконання
1. Вступ		
2. Вивчення методичних рекомендацій та стандартів та звіт керівнику		
3. Постанова задачі		
4. Інформаційний огляд джерел бібліотек та інтернету		
5. Теоретична частина		
6. Практична частина		
7. Закінчення оформлення		
8. Доповідь студента на кафедрі		
9. Доробка (за необхідністю), рецензування		

Дата видачі завдання «__» _____ 202__ р.

Здобувач вищої освіти _____ Кизименко Сергій Анатолійович
(підпис)

Науковий керівник _____ к.пед.н., доц. Кошова О.П.
(підпис) (науковий ступінь, вчене звання, ініціали та прізвище)

Результати захисту кваліфікаційної роботи

Кваліфікаційна робота оцінена на _____
(балів, оцінка за національною шкалою, оцінка за ECTS)

Протокол засідання ЕК № _____ від «__» _____ 2023 р.

Секретар ЕК _____
(підпис) (ініціали та прізвище)

Затверджую

Зав. кафедрою _____
к.ф.-м.н. Олена ОЛЬХОВСЬКА
« ____ » _____ 202__ р.

Погоджено

Науковий керівник _____
к.пед.н., Оксана КОШОВА
« ____ » _____ 202__ р.

План

кваліфікаційної роботи на тему
«Розробка програмного забезпечення тренажеру з теми «Аналітична геометрія на площині» дистанційного навчального курсу «Прикладна математика»
зі спеціальності 122 Комп'ютерні науки
освітня програма 122 «Комп'ютерні науки»
ступеня бакалавр
Прізвище, ім'я, по батькові Кизименко Сергій Анатолійович

ВСТУП**1. РОЗДІЛ 1. ПОСТАНОВКА ЗАДАЧІ****2. РОЗДІЛ 2. ОГЛЯД СИСТЕМ АНАЛОГІЧНОГО ПРИЗНАЧЕННЯ**

2.1. Платформа Classtime

2.2. Платформа На Урок

2.3. Платформа Kahoot!

2.4 Платформа Moodle

2.5 Платформа Claroline LMS

3. РОЗДІЛ 3. ТЕОРЕТИЧНА ЧАСТИНА

3.1. Застосування аналітичної геометрії на площині

3.2. Алгоритм роботи тренажеру з теми «Аналітична геометрія на площині»
дистанційного навчального курсу «Прикладна математика»**4. РОЗДІЛ 4. ПРАКТИЧЕНА ЧАСТИНА**

4.1. Блок-схема тренажеру

4.2. Опис роботи тренажеру

ВИСНОВКИ**СПИСОК ЛІТЕРАТУРИ****ДОДАТОК А. ВИХІДНІ КОДИ**

Здобувач вищої освіти _____ Сергій Кизименко
« ____ » _____ 202__ р.

РЕФЕРАТ

Записка: 42 сторінок, 14 рисунків, 1 додаток, 15 літературних джерел.

РОЗРОБКА ПРОГРАМНОГО ТРЕНАЖЕРУ, «АНАЛІТИЧНА ГЕОМЕТРІЯ НА ПЛОЩИНІ», «ПРИКЛАДНА МАТЕМАТИКА», МОВА ПРОГРАМУВАННЯ PYTHON, SUBLIME TEXT.

Мета роботи – Розробка програмного забезпечення тренажеру з теми «Аналітична геометрія на площині» дистанційного навчального курсу «Прикладна математика»

Об'єкт роботи - система дистанційного навчання.

Методи дослідження – використання редактор коду «Visual Studio Code»,

«Sublime Text», мову програмування Python.

ЗМІСТ

ВСТУП.....	7
РОЗДІЛ 1. ПОСТАНОВКА ЗАДАЧІ.....	9
РОЗДІЛ 2. ОГЛЯД СИСТЕМ АНАЛОГІЧНОГО ПРИЗНАЧЕННЯ.....	10
2.1. Платформа Classtime	10
2.2. Платформа На Урок.....	11
2.3. Платформа Kahoot!	12
2.4. Платформа Moodle.....	13
2.5. Платформа Claroline LMS	14
РОЗДІЛ 3. ТЕОРЕТИЧНА ЧАСТИНА.....	16
3.1. Застосування аналітичної геометрії на площині	16
3.2. Алгоритм роботи тренажеру з теми «Аналітична геометрія на площині» дистанційного навчального курсу «Прикладна математика»	19
РОЗДІЛ 4. ПРАКТИЧНА ЧАСТИНА.....	25
4.1. Блок-схема прецедентів.	25
4.2. Опис роботи тренажеру	33
ВИСНОВКИ.....	36
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	38
ДОДАТОК А. ВИХІДНІ КОДИ	40

ВСТУП

В умовах військових дій технологія дистанційного навчання виявилась єдиною освітньою технологією, яка дозволила ВНЗ продовжувати надання освітніх послуг здобувачам вищої та середньої освіти. Воєнний стан - це особливий правовий режим, що запроваджено в Україні або в окремих її місцевостях у разі збройної агресії чи загрози нападу, небезпеки державній незалежності України, її територіальній цілісності.

ВНЗ використовують різноманітні LMS, серед яких найбільш популярні: Moodle, TrainingWare, Claroline LMS та інші технології дистанційного навчання дозволили не тільки реалізувати швидкий та зручний доступ до навчального матеріалу, але й застосовувати технології спільної творчої діяльності, кейс-методи, автоматизовані та відкриті форми контролю, як автоматизовані, так і відкриті види контролю та ін. Але ситуація під час війни має декілька відмінностей у порівнянні з ситуацією під час карантину, нестабільність або повна відсутність доступу до Інтернету, неможливість виходу до мережі у певні моменти або інтервали часу, психологічний стан студентів.

Дистанційне навчання – це сукупність інформаційних технологій, що забезпечує надання інтерактивної взаємодії між викладачем і студентом в процесі реального часу, можливість опрацьовувати самостійно матеріал на хмарних ресурсах у відкритому доступі, проходити тренінги в навчальних тренажерах та покращувати власні навички.

Під дистанційним навчанням розуміється індивідуальний процес набуття знань, умінь, навичок і способів пізнавальної діяльності людини, який відбувається в основному за опосередкованої взаємодії віддалених один від одного учасників навчального процесу у спеціалізованому середовищі, яке функціонує на базі сучасних інформаційно-комунікаційних технологій та навчальних тренажерів.

У цих умовах раціональним є використання асинхронних методів, що дозволяють студентам вивчати матеріал та виконувати завдання без врахування фактору часу або географічного місця знаходження студента.

У якості навчальних матеріалів доцільно використовувати текстові лекції або навчальні тренажери дисциплін. Слід обмежити випадки, коли у студентів під час опрацювання навчального матеріалу виникає необхідність використовувати додаткові джерела, які складно знайти в мережі.

Мета дипломної роботи – розробка програмного забезпечення тренажеру з теми «Аналітична геометрія на площині» дистанційного навчального курсу «Прикладна математика»

Об'єкт - система дистанційного навчання.

Предмет розробки – програмна реалізація тренажера з теми «Аналітична геометрія на площині «Прикладна математика».

РОЗДІЛ 1. ПОСТАНОВКА ЗАДАЧІ

Першочерговим завданням дипломної роботи – є розробка програмного забезпечення тренажеру з теми «Аналітична геометрія на площині» дистанційного навчального курсу «Прикладна математика»

Завдання дипломної роботи:

- переглянути лекційний матеріал з теми «Аналітична геометрія на площині»;
- написати алгоритм роботи тренажеру;
- створити блок-схему роботи тренажеру;
- програмування програми тренажеру;
- описати результати програмування тренажеру;
- тестування.

Технічне завдання представляє собою фундамент для подальшої роботи. В ньому прописуються всі деталі: структура програми, дизайн, функціонал, а також можливості зміни програми.

Програмування – це процес створення будь якого програмного коду, тобто на будь якому мові програмування.

Макет дизайну програми – це відображення як буде виглядати програмний продукт перед розробкою та його закінчені. Елементи які він може містити, це фото, іконки та інші елементи.

Тестування – це процес перевірки функціональності розробленого тренажеру.

РОЗДІЛ 2. ОГЛЯД СИСТЕМ АНАЛОГІЧНОГО ПРИЗНАЧЕННЯ

2.1. Платформа Classtime

Classtime – це гнучка онлайн-платформа, яка допомагає вчителю проводити онлайн-оцінювання знань і миттєво оцінити прогрес класу і кожного учня індивідуально (Рисунок 2.1).



Рисунок 2.1 – логотип «Classtime»

Classtime в свою чергу дозволяє викладачам вибирати з безлічі типів питань для створення унікальних форматів оцінювання за короткий проміжок часу. Хоча Classroom може використовуватися як LMS, Classtime краще підходить для оцінювань і часто використовується разом з Classroom.

Широкі можливості оцінювання і формування знань, фокус на розвитку мислення, наприклад:

- Інтуїтивно зрозумілий легкий у використанні інтерфейс платформи;
- 9 типів питань;
- 16 мов інтерфейсу доступно;
- 9 гнучких налаштувань кожної унікальної сесії (уроку);
- Можливість додавати аудіо, відео, картинки в завдання;
- Введення математичних, хімічних, фізичних формул;

- Відкрита Бібліотека з перевіреними питаннями;
- Інтеграція з Google Classroom;
- Детальна звітність: експорт результатів у Excel, PDF, індивідуальні звіти учнів і аналітика[10].

2.2. Платформа На Урок

Онлайн-тести «На Урок» – це безкоштовний багатофункціональний сервіс, розроблений для педагогів України, який містить інтерактивні завдання для контролю знань і залучення учнів до активної роботи у класі та вдома. Весь функціонал ресурсу доступний для кожного зареєстрованого користувача сайту без обмежень (Рисунок 2.2).



Рисунок 2.2 – логотип «На Урок»

Можливості онлайн-тестів «На Урок»:

- Робота в режимі реального часу.
- Формування матеріалів для тематичного контролю знань, для інтерактивної взаємодії під час уроку.
- Оцінювання роботи за 12-бальною, відсотковою та індивідуальною шкалою.
- Трансформація будь-якого тесту у флеш-картки або у формат гри «Відповідності».

- Використання власних тестів або 2 млн онлайн-тестів колег, які зберігаються в Бібліотеці
- Можливість трансформації шрифтів для дітей з дислексією.
- Збереження результатів тестувань в особистому кабінеті вчителя з можливістю їх експорту.
- Покрокові інструкції та Служба підтримки для ефективної роботи з сервісом[11].

2.3. Платформа Kahoot!

Kahoot! – онлайн сервіс для створення інтерактивних завдань. Дозволяє створювати тести, опитування, вікторини. Платформу можна використовувати під час роботи з будь-якими віковими категоріями. Добре підійде як для молодшої школи, так і для дорослих людей. У статті описано як створити тест Kahoot!, надіслати його учням (Рисунок 2.3).



Рисунок 2.3 – логотип «Kahoot!»

- Дозволяє залучати до тестування до 50 студентів;
- Питання можна створювати самостійно (кнопка Create (Створити) у лівому верхньому куті екрану) або використовувати вже готові з банку питань

(щоб переглянути весь банк питань потрібно натиснути кнопку Discover у верхній частині екрана зліва);

- Можна включити функцію для автоматичного перемішування відповідей у питанні;
- Для візуалізації питання можна користуватися банком зображень, додавати їх до запитань чи використовувати як відповіді;
- Є можливість обмежувати час, що дається студентові для відповіді на питання;
- Можна визначати кількість балів за кожну правильну відповідь;
- Дозволяє дізнатися як відповідав на запитання кожен студент, або будувати діаграми успішності академічної групи.

Платформа дозволяє проводити тестування двома способами:

- Віртуальний клас (virtual classroom) – тестування можна пройти разом із студентами в аудиторії. У цьому випадку питання та варіанти відповідей з'являються на екрані проектора або комп'ютера викладача, а відповідають студенти зі своїх мобільних телефонів або комп'ютерів.
- Самостійне навчання (for self placed learning) – студенти проходять тестування самостійно, питання та варіанти відповідей з'являються на екрані їхніх комп'ютерів чи смартфонів. Обравши цей спосіб, викладач має можливість встановити дату та період часу, протягом якого тестування буде відкритим[13].

2.4. Платформа Moodle

Moodle – це відкрите програмне забезпечення для створення та управління електронними курсами та навчальними матеріалами в онлайн-середовищі. За допомогою Moodle можна створювати віртуальні класи, де викладачі можуть розміщувати матеріали для навчання, виставляти завдання та тестування для студентів, проводити дистанційні заняття та спілкуватися з ними (Рисунок 2.4).



Рисунок 2.4 – логотип «Moodle»

Moodle має багатий функціонал, який дозволяє налаштувати курси під потреби конкретного навчального закладу та забезпечити ефективне навчання. Він є популярним інструментом для онлайн-навчання та дистанційного навчання, який дозволяє забезпечити доступ до якісної освіти у будь-якому місці та в будь-який час. Він також має розгалужену глобальну спільноту розробників та користувачів, яка постійно працює над вдосконаленням системи та надає підтримку користувачам.

Типова функціональність Moodle включає:

- здача завдань;
- дискусійні форуми;
- завантаження файлів;
- оцінювання;
- обмін повідомленнями;
- календар подій;
- новини та анонси подій;
- онлайн тестування[14].

2.5. Платформа Claroline LMS

Claroline LMS – це платформа для електронного навчання (e-Learning) та електронної діяльності (e-Working), що дозволяє вчителям створювати ефективні онлайн-курси та керувати процесом навчання та спільними діями на основі веб-технологій(Рисунок 2.5).



Рисунок 2.5 – логотип «Claroline LMS»

Claroline LMS випущена на основі ліцензії з Відкритим Кодом (Open Source). Вона застосовується у сотнях організацій 90 країн світу. Інтерфейс програми перекладено 35 мовами. Дозволяє створювати та адмініструвати курси в режимі он-лайн. Кожен курс містить низку інструментів, що дозволяють викладачеві:

- Вказати опис курсу.
- Опублікувати документи у будь-якому форматі (текст, PDF, HTML, відео...).
- Адмініструвати публічні та приватні форуми.
- Розробляти шляхи навчання.
- Об'єднувати студентів у групи.
- Підготовляти для студентів вправи (завдання).
- Керувати порядком денним із завданнями та строками виконання.
- Публікувати анонси (також і ел. поштою).
- Вивішувати онлайн інформацію про поточні завдання.
- Перегляд статистики активності користувачів.
- Використовувати технологію wiki для спільного написання документів[15].

РОЗДІЛ 3. ТЕОРЕТИЧНА ЧАСТИНА

3.1. Застосування аналітичної геометрії на площині

Відомі різноманітні застосування аналітичної геометрії на площині.

1. В алгебрі,
2. В геометрії.
3. У фізиці

Аналітична геометрія – це розділ геометрії, в якому простіші геометричні образи (прямі, площини, лінії і поверхні другого порядку) досліджуються засобами алгебри на основі методу координат. Аналітична геометрія дає нам можливість описувати просторові образи або фігури за допомогою алгебраїчних співвідношень між координатами точок, які належать цим образам або фігурам. Це необхідно для дослідження властивостей об'єктів розташованих в просторі, на площині через дослідження властивостей функцій – аналітичних образів цих об'єктів. Без таких співвідношень (функція – графічний образ) неможливо було б побудувати графік або зобразити рух на комп'ютері, спрогнозувати погоду по швидкості зміни показників барометра і т.д. В аналітичній геометрії простішим геометричним образам ставляться у відповідність їх аналітичні еквіваленти – алгебраїчні рівняння. Вивчаючи і аналізуючи ці рівняння одержують інформацію про властивості і про взаємне розташування цих геометричних об'єктів. В основі цих досліджень лежить метод координат .

Будь-яке лінійне рівняння зі змінними x, y, z можна розглядати як рівняння у декартових координатах площини у просторі. Різні форми рівняння площини наведені у таблиці .

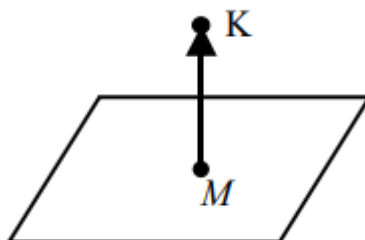
Якщо у рівнянні відсутній доданок з якою-небудь змінною, то площина паралельна відповідній координатній осі; наприклад, площина, яку задано рівнянням $Ax + Cz + D = 0$, паралельна осі Oy .

Якщо у рівнянні відсутні доданки з двома змінними, то площина паралельна відповідній координатній площині; наприклад, площина, яку задано рівнянням $Ax + D = 0$, паралельна площині Oyz .

Якщо у загальному або у нормальному рівнянні площини відсутній вільний член, тобто рівняння має вигляд $Ax + By + Cz = 0$, площина проходить через початок координат.

Задача 2.3.1. Записати рівняння площини, яка проходить через точку $M(3;4;5)$ перпендикулярно до вектора MK , де $K(-2;1;3)$.

Розв'язання. Нормаль до площини – це вектор MK $(-2-3; 1-4; 3-5)=(-5;-3;-2)$



Запишемо канонічне рівняння площини:

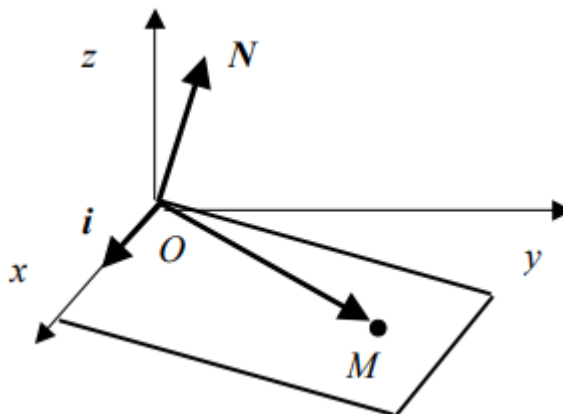
$$A(x-x_0)+B(y-y_0)+C(z-z_0)=0;$$

$$-5(x-3)-3(y-4)-2(z-5)=0;$$

$$5x+3y+2z-37=0.$$

Задача 2.3.2. Записати рівняння площини, яка проходить через точку $M(3;2;-1)$ та вісь Ox .

Розв'язання.



Площина проходить через вісь Ox , тобто у площині лежить напрямний вектор осі Ox – векторі .

Площина проходить через початок координат та точку M , отже у площині лежить також вектор $OM = (3; 2; -1)$ – радіус-вектор точки M .

Векторний добуток векторів, які лежать у площині, перпендикулярний цій площині, отже, його можна вважати її нормаллю, тобто $N = i \times OM = i \times (3i + 2j - k) = i \times 3i + i \times 2j - i \times k = 2k + j$ (звичайно, векторний добуток можна обчислювати також за допомогою символічного визначника).

Запишемо рівняння площини, яка проходить через точку M та має нормаль

$$N = (0; 1; 2):$$

$$0(x - 3) + 1(y - 2) + 2(z + 1) = 0;$$

$$y + 2z = 0.$$

Задача 2.3.3. Чи будуть перпендикулярними площини $3x + 2y + 4z - 1 = 0$ та $2x + 3y - z = 0$?

Розв'язання. Обчислимо скалярний добуток нормалей до площин $(3; 2; 4)$ $N_1 =$ та $(2; 3; 1)$ $N_2 =$: $N_1 \cdot N_2 = 3 \cdot 2 + 2 \cdot 3 + 4 \cdot (-1) = 8 \neq 0$. Отже, за ознакою перпендикулярності площин задані площини не будуть перпендикулярними.

Задача 2.3.4. Складіть рівняння прямої, яка проходить через точку $A(-1; 3)$ і паралельна прямій $3x - 2y - 1 = 0$.

Розв'язання. Зведемо рівняння прямої до прямої з кутовим коефіцієнтом $3x - 1 = 2y; y = 3/2 * x - 1/2$.

Оскільки прямі паралельні то кутові коефіцієнти в них рівні

$$k = 3/2; y = 3/2 * x + c.$$

підставимо точку і знайдемо невідому

$$3 = 3/2 * (-1) + c; c = 3 + 3/2 = 9/2.$$

Підставимо сталу у рівняння прямої

$$y = 3/2 * x + 9/2 \text{ або } 3x - 2y + 9 = 0.$$

Задача 2.3.5. Довести, що наступне рівняння є еліпс: $5x^2 - 30x + 9y^2 + 18y = -9$. Знайти координати його центра, піввісі та ексцентриситет.

Розв'язання. Виконаємо перетворення, виділяючи повні квадрати:

$$5x^2 - 30x = 5(x^2 - 6x) = 5[(x-3)^2 - 9] = 5(x-3)^2 - 45;$$

$$9y^2 + 18y = 9(y^2 + 2y) = 9[(y+1)^2 - 1] = 9(y+1)^2 - 9;$$

Підставляючи це у дане рівняння, одержимо:

$$5(x-3)^2 - 45 + 9(y+1)^2 - 9 + 9 = 0 \text{ або } 5(x-3)^2 + 9(y+1)^2 = 45.$$

Поділивши обидві частини на 45, маємо $\frac{(x-3)^2}{9} + \frac{(y+1)^2}{5} = 1$. Отже, одержано рівняння еліпса із зміщеним центром. [1]

3.2. Алгоритм роботи тренажеру з теми «Аналітична геометрія на площині» дистанційного навчального курсу «Прикладна математика»

Якщо запустити створене програмне забезпечення, на головній формі користувач побачить:

Перший крок. Користувачу відобразиться питання з теми, на нього можна дати відповідь натиснувши лише на один варіант відповіді: «Як називається рівняння прямої, що описується формулою $Ax + By + C = 0$?»

- А. Загальне рівняння прямої (Правильна відповідь)
- Б. Канонічне рівняння прямої
- В. Рівняння прямої у відрізках на осях
- Г. Рівняння прямої, що проходить через дві задані точки

Якщо користувач вибрав правильний варіант відповіді, то програма автоматично перейде на наступне запитання. Якщо користувач вибрав неправильний варіант відповіді, то програма не переходить на наступний крок.

Другий крок. Користувачу відобразиться питання, на нього можна дати відповідь натиснувши лише на один варіант відповіді: «Як називається рівняння прямої, що описується формулою $\frac{x - x_1}{x_2 - x_1} = \frac{y - y_1}{y_2 - y_1}$?»

- А. Загальне рівняння прямої
- Б. Канонічне рівняння прямої
- В. Рівняння прямої з кутовим коефіцієнтом

Г. Рівняння прямої, що проходить через дві задані точки(Правильна відповідь)

Якщо користувач вибрав правильний варіант відповіді, то програма автоматично перейде на наступне запитання. Якщо користувач вибрав неправильний варіант відповіді, то програма не переходить на наступний крок.

Третій крок. Користувачу відобразиться питання, на нього можна дати відповідь натиснувши лише на один варіант відповіді: «Рівняння прямої, що проходить через задану точку $M_0(x_0; y_0)$ в заданому напрямку $tg\alpha = k$ має вигляд:»

А. $y = kx + b$.

Б. $\frac{x - x_1}{x_2 - x_1} = \frac{y - y_1}{y_2 - y_1}$

В. $y - y_0 = k(x - x_0)$ (Правильна відповідь)

Г. $Ax + By + C = 0$

Якщо користувач вибрав правильний варіант відповіді, то програма автоматично перейде на наступне запитання. Якщо користувач вибрав неправильний варіант відповіді, то програма не переходить на наступний крок.

Четвертий крок. Користувачу відобразиться питання, на нього можна дати відповідь натиснувши лише на один варіант відповіді: «Дано дві точки А(1; 2) і В(3; -1). Рівняння прямої АВ має вигляд:»

А. $x - y - 1 = 0$

Б. $3x + y - 1 = 0$

В. $2y + 3x - 7 = 0$ (Правильна відповідь)

Г. $2x - 3y - 1 = 0$

Якщо користувач вибрав правильний варіант відповіді, то програма автоматично перейде на наступне запитання. Якщо користувач вибрав неправильний варіант відповіді, то програма не переходить на наступний крок.

П'ятий крок. Користувачу відобразиться питання, на нього можна дати відповідь натиснувши лише на один варіант відповіді: «Дано дві точки А(3; -1)

і $B(2; 0)$. Знайдіть рівняння прямої АВ. У відповідь запишіть коефіцієнт С із загального рівняння прямої.»

- А. 1
- Б. -2 (Правильна відповідь)
- В. -5
- Г. 0
- Д. 2

Якщо користувач вибрав правильний варіант відповіді, то програма автоматично перейде на наступне запитання. Якщо користувач вибрав неправильний варіант відповіді, то програма не переходить на наступний крок.

Шостий крок. Користувачу відобразиться питання, на нього можна дати відповідь натиснувши лише на один варіант відповіді: «Прямі $l_1 : y = k_1x + b_1$, $l_2 : y = k_2x + b_2$ перпендикулярні, якщо:»

- А. $k_1 = k_2, b_1 \neq b_2$
- Б. $k_1 = k_2, b_1 = b_2$
- В. $k_1 \cdot k_2 = -1$ (Правильна відповідь)
- Г. $k_1 \neq k_2$

Якщо користувач вибрав правильний варіант відповіді, то програма автоматично перейде на наступне запитання. Якщо користувач вибрав неправильний варіант відповіді, то програма не переходить на наступний крок.

Сьомий крок. Користувачу відобразиться питання, на нього можна дати відповідь натиснувши лише на один варіант відповіді: «Множина всіх точок площини, рівновіддалених від даної точки (центра) називається ...»

- А) колом. (Правильна відповідь)
- Б) еліпсом.
- В) гіперболою.
- Г) параболою.

Якщо користувач вибрав правильний варіант відповіді, то програма автоматично перейде на наступне запитання. Якщо користувач вибрав неправильний варіант відповіді, то програма не переходить на наступний крок.

Восьмий крок. Користувачу відобразиться питання, на нього можна дати відповідь натиснувши лише на один варіант відповіді: «За якою із перерахованих нижче формул можна знайти рівняння асимптот гіперболи?»

А) $y = a + \frac{c}{a}x$

Б) $y = \pm \frac{b}{a}x$ (Правильна відповідь)

В) $y = a - \frac{c}{a}x$

Г) $y = \pm \frac{a}{b}x$

Якщо користувач вибрав правильний варіант відповіді, то програма автоматично перейде на наступне запитання. Якщо користувач вибрав неправильний варіант відповіді, то програма не переходить на наступний крок.

Дев'ятий крок. Користувачу відобразиться питання, на нього можна дати відповідь натиснувши лише на один варіант відповіді: «Еліпсом називається...»

А) геометричне місце точок площини, сума відстаней від кожної з яких до двох даних точок F_1 і F_2 (фокусів) є величина стала і рівна $2a$, тобто $r_1 + r_2 = 2a$ (Правильна відповідь)

Б) геометричне місце точок площини, для яких абсолютна величина різниці відстаней до двох даних точок F_1 і F_2 (фокусів) є величиною сталою і рівною $2a$, тобто $|r_1 - r_2| = 2a$

В) множина всіх точок площини, рівновіддалених від даної точки (центра)

Г) геометричне місце точок площини, однаково віддалених від однієї точки F (фокуса) і даної прямої (директриси)

Якщо користувач вибрав правильний варіант відповіді, то програма автоматично перейде на наступне запитання. Якщо користувач вибрав неправильний варіант відповіді, то програма не переходить на наступний крок.

Десятий крок. Користувачу відобразиться питання, на нього можна дати відповідь натиснувши лише на один варіант відповіді: «Серед наведених рівнянь рівнянням еліпса є: »

А) $3x^2 - 4y^2 = 12$;

В) $3x^2 + 4y^2 = 12$; (Правильна відповідь)

Б) $4x^2 + 3y^2 = -24$;

Г) $4x^2 - 3y^2 = 24$.

Якщо користувач вибрав правильний варіант відповіді, то програма автоматично перейде на наступне запитання. Якщо користувач вибрав неправильний варіант відповіді, то програма не переходить на наступний крок.

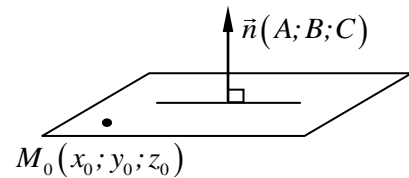
Одинадцятий крок. Користувачу відобразиться питання, на нього можна дати відповідь натиснувши лише на один варіант відповіді: «Вектор $\vec{n}(A; B; C)$, який зображено на рисунку називається ... вектором площини. Вставте пропущений термін.»

А. нормальним (Правильна відповідь)

Б. перпендикулярним

В. напрямним

Г. паралельним



Якщо користувач вибрав правильний варіант відповіді, то програма автоматично перейде на наступне запитання. Якщо користувач вибрав неправильний варіант відповіді, то програма не переходить на наступний крок.

Дванадцятий крок. Користувачу відобразиться питання, на нього можна дати відповідь натиснувши лише на один варіант відповіді: «Рівняння площини за даною точкою $M_0(x_0; y_0; z_0)$ і заданим ненульовим нормальним вектором $\vec{n}(A; B; C)$ має вигляд:»

$$\text{A. } \begin{vmatrix} x-x_1 & y-y_1 & z-z_1 \\ x_2-x_1 & y_2-y_1 & z_2-z_1 \\ x_3-x_1 & y_3-y_1 & z_3-z_1 \end{vmatrix} = 0.$$

$$\text{Б. } A(x-x_0)+B(y-y_0)+C(z-z_0)=0 \text{ (Правильна відповідь)}$$

$$\text{В. } Ax+By+Cz+D=0$$

$$\text{Г. } \frac{x-x_1}{x_2-x_1} = \frac{y-y_1}{y_2-y_1} = \frac{z-z_1}{z_2-z_1}$$

Якщо користувач вибрав правильний варіант відповіді, то програма автоматично перейде на наступне запитання. Якщо користувач вибрав неправильний варіант відповіді, то програма не переходить на наступний крок.

РОЗДІЛ 4. ПРАКТИЧНА ЧАСТИНА

4.1. Блок-схема прецедентів.

Для опису роботи програми було побудовано блок-схему прецедентів (Рисунок 4.1).

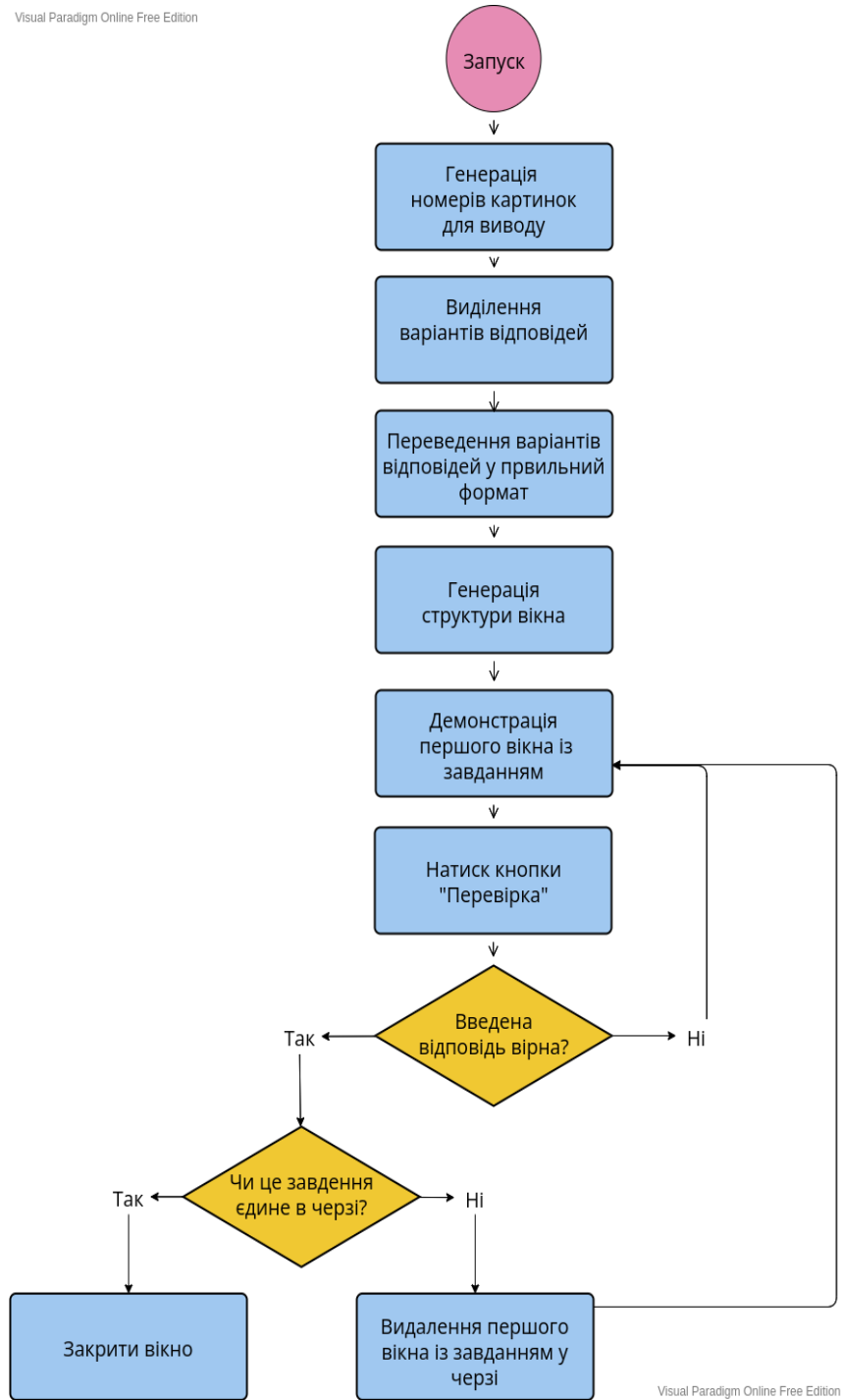


Рисунок 4.1 – Діаграма прецедентів

Пояснення блок-схеми прецедентів. Для початку треба запустити програму main.py, потім вона автоматично генерує номер картинок для виводу на екран. Також ця платформа автоматично зображує відповіді під кожен картинку та переводить варіант відповідей у правильний формат, що створює генерацію структури вікна. Після цього ми побачимо на екрані демонстрацію першої картинки із завданням, далі ми повинні вибрати відповідь до запитання та натиснути кнопку «перевірити». Якщо відповідь вірна, вас переведе до наступного питання, якщо відповідь не вірна програма не пропустить далі доки відповідь не буде вірною.

Visual Paradigm – це набір програмних рішень для трансформації бізнесу та ІТ. Віднести її до якогось одного класу програмного забезпечення важко. Це платформа з дуже широкими функціональними можливостями (перераховані не всі можливості):

- Планування архітектури підприємства та трансформація бізнесу (Zachman Framework, TOGAF, ArchiMate та ін.);
- візуальне моделювання процесів (BPMN, CMMN, VAD, EPC, DFD, UML та ін.);
- імітаційне моделювання (симуляція процесу для BPMN);
- управління проектами (PMBOK, Agile & Scrum, Just-in-Time Process Map, PERT, Roadmap, WBS та ін.);
- розробка коду та баз даних, інструменти проектування баз даних (ERD та ін.);
- моделювання структури організації.

Є можливість створити єдиний у всій системі глосарій термінів. Visual Paradigm підтримує функцію мультимовності - та сама модель процесу може відобразитися різними мовами (через перемикання мови). Тому нема потреби створювати клони моделей одного процесу для різних мов.

Процесна архітектура необхідна для:

- Моделювання процесів верхнього рівня;

- Декомпозиції та аналізу ситуації “згори донизу”;
- Розуміння основних етапів робіт;
- Розуміння взаємозв'язків між головними етапами.

Python — інтерпретована об'єктно-орієнтована мова програмування високого рівня зі строгою динамічною типізацією. Розроблена в 1990 році Гвідо ван Россумом. Структури даних високого рівня разом із динамічною семантикою та динамічним зв'язуванням роблять її привабливою для швидкої розробки програм, а також як засіб поєднування наявних компонентів. Python підтримує модулі та пакети модулів, що сприяє модульності та повторному використанню коду. Інтерпретатор Python та стандартні бібліотеки доступні як у скомпільованій, так і у вихідній формі на всіх основних платформах. В мові програмування Python підтримується кілька парадигм програмування, зокрема: об'єктно-орієнтована, процедурна, функціональна та аспектно-орієнтована.

Серед основних її переваг можна назвати такі:

- чистий синтаксис (для виділення блоків слід використовувати відступи);
- переносність програм (що властиве більшості інтерпретованих мов);
- стандартний дистрибутив має велику кількість корисних модулів (включно з модулем для розробки графічного інтерфейсу);
- можливість використання Python в діалоговому режимі (дуже корисне для експериментування та розв'язання простих задач);
- стандартний дистрибутив має просте, але разом із тим досить потужне середовище розробки, яке зветься IDLE і яке написане мовою Python;
- зручний для розв'язання математичних проблем (має засоби роботи з комплексними числами, може оперувати з цілими числами довільної величини, у діалоговому режимі може використовуватися як потужний калькулятор);

- відкритий код (можливість редагувати його іншими користувачами).

Python має ефективні структури даних високого рівня та простий, але ефективний підхід до об'єктно-орієнтованого програмування. Елегантний синтаксис Python, динамічна обробка типів, а також те, що це інтерпретована мова, роблять її ідеальною для написання скриптів та швидкої розробки прикладних програм у багатьох галузях на більшості платформ.

Інтерпретатор мови Python і багата Стандартна бібліотека (як вихідні тексти, так і бінарні дистрибутиви для всіх основних операційних систем) можуть бути отримані з сайту Python і можуть вільно розповсюджуватися. Цей самий сайт має дистрибутиви та посилання на численні модулі, програми, утиліти та додаткову документацію.

Інтерпретатор мови Python може бути розширений функціями та типами даних, розробленими на C чи C++ (або на іншій мові, яку можна викликати із C). Python також зручна як мова розширення для прикладних програм, що потребують подальшого налагодження.

Також бувають впливи інших мов на Python:

- ABC — відступи (поля) для групування операторів, високорівневі структури даних (map) (фактично, Python створювався як спроба виправити помилки, допущені при проектуванні ABC);
- Modula-3[en] — пакети, модулі, використання else спільно з try та except, іменовані аргументи функцій (на це також вплинув Common Lisp);
- C, C++ — деякі синтаксичні конструкції (як пише сам Гвідо ван Россум — він використовував найбільш несуперечливі конструкції з C, щоб не викликати неприязнь у Cі-програмістів до Python);
- Smalltalk — об'єктно-орієнтоване програмування;
- Lisp — окремі риси функціонального програмування (lambda, map, reduce, filter та інші);
- Fortran — зрізи масивів, комплексна арифметика;

- Miranda[en] — спискові вирази;
- Java — модулі logging, unittest, threading (частина можливостей оригінального модуля не реалізована), xml.sax стандартної бібліотеки, спільне використання finally та except при обробці винятків, використання @ для декораторів;
 - Icon[en] — генератори.
 - TypeScript — ідея синтаксису підказок типів (type hinting). З версії Python 3.10 система підказок типів стала унікальною, «злившись» з мовою в одне ціле. До того ж у Python підказки типів ніяк не впливають на роботу інтерпретатора.

Більша частина інших можливостей Python (наприклад, байт-компіляція вихідного коду) також була реалізована раніше в інших мовах. [3]

Програма написана та перевірялась на декількох платформах, такі як Sublime Text, Visual Studio Code, IDLE Shell 3.8.10.

Sublime Text — це швидкий кросплатформенний текстовий редактор. Підтримує плагіни, розроблені за допомогою мови програмування Python. Sublime Text не є вільним чи відкритим програмним забезпеченням, але деякі його плагіни розповсюджуються з вільною ліцензією, розробляються і підтримуються спільнотою розробників. Редактор містить різні візуальні теми, з можливістю завантаження додаткових. Користувачі бачать весь свій код в правій частині екрану у вигляді міні-карти, при кліці на яку можна здійснювати навігацію. Є кілька режимів екрану. Один з них включає від 1 до 4 панелей, за допомогою яких можна показувати до чотирьох файлів одночасно. Повноцінний (free modes) режим показує тільки один файл без будь-яких додаткових навколо нього меню. [5]

Visual Studio Code — це редактор вихідного коду, який можна використовувати з різними мовами програмування, включаючи C, C#, C++, Fortran, Go, Java, JavaScript, Node.js, Python, Rust. Він базується на структурі Electron, яка використовується для розробки веб-додатків Node.js, які працюють

на механізмі компонування Blink. Visual Studio Code використовує той самий компонент редактора (під кодовою назвою «Monaco»), який використовується в Azure DevOps (раніше називався Visual Studio Online і Visual Studio Team Services).

З коробки Visual Studio Code містить базову підтримку для більшості поширених мов програмування. Ця базова підтримка включає підсвічування синтаксису, зіставлення дужок, згортання коду та настроювані фрагменти. Visual Studio Code також постачається з IntelliSense для JavaScript, TypeScript, JSON, CSS і HTML, а також підтримує налагодження Node.js. Підтримка додаткових мов може бути забезпечена безкоштовними розширеннями на VS Code Marketplace. [7]

IDLE (Integrated Development and Learning Environment) — це інтегроване середовище розробки та навчання мовою Python, створене за допомогою бібліотеки Tkinter. Офіційно — спотворення IDE, але насправді названа на честь Еріка Айдла (Eric Idle) з Монті Пайтон. Постачається разом з Python і завдяки використанню Tkinter може використовуватися на багатьох платформах, серед яких Windows, Mac OS, Unix-подібні ОС.

За допомогою IDLE можна виконувати звичайні завдання для інтегрованого середовища: переглядати, редагувати, запускати, налагоджувати програми на Python. Редактор коду використовує підсвічування синтаксису. IDLE пропонує додаткові можливості для досвідчених користувачів, наприклад, засіб перегляду об'єктів. [9]

4.2. Опис програмної реалізації

Елементи програми тренажеру було реалізовано на мові програмування Python, в середовищі розробки IDLE(Shell 3.8.10). (рис. 4.2)

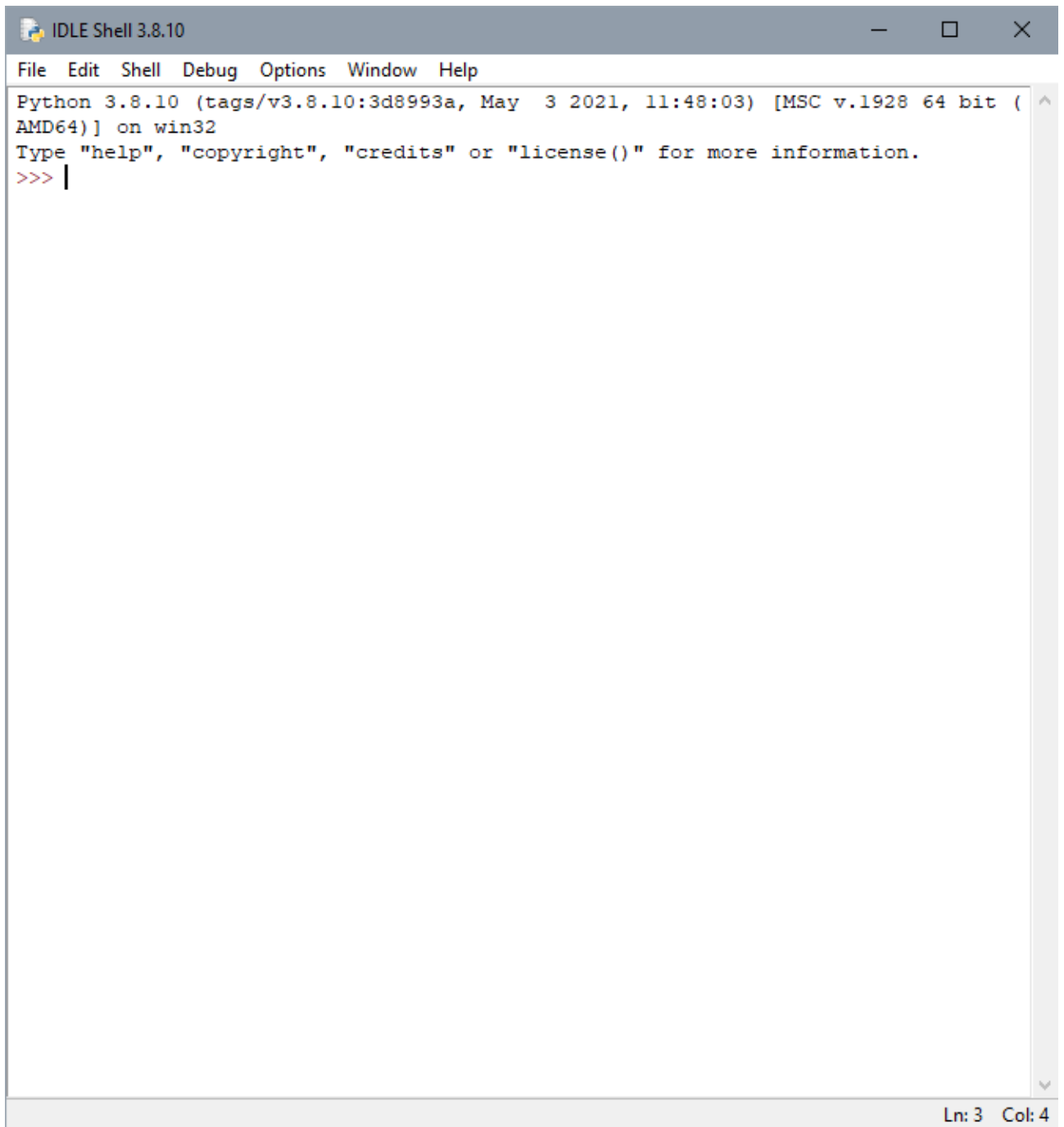
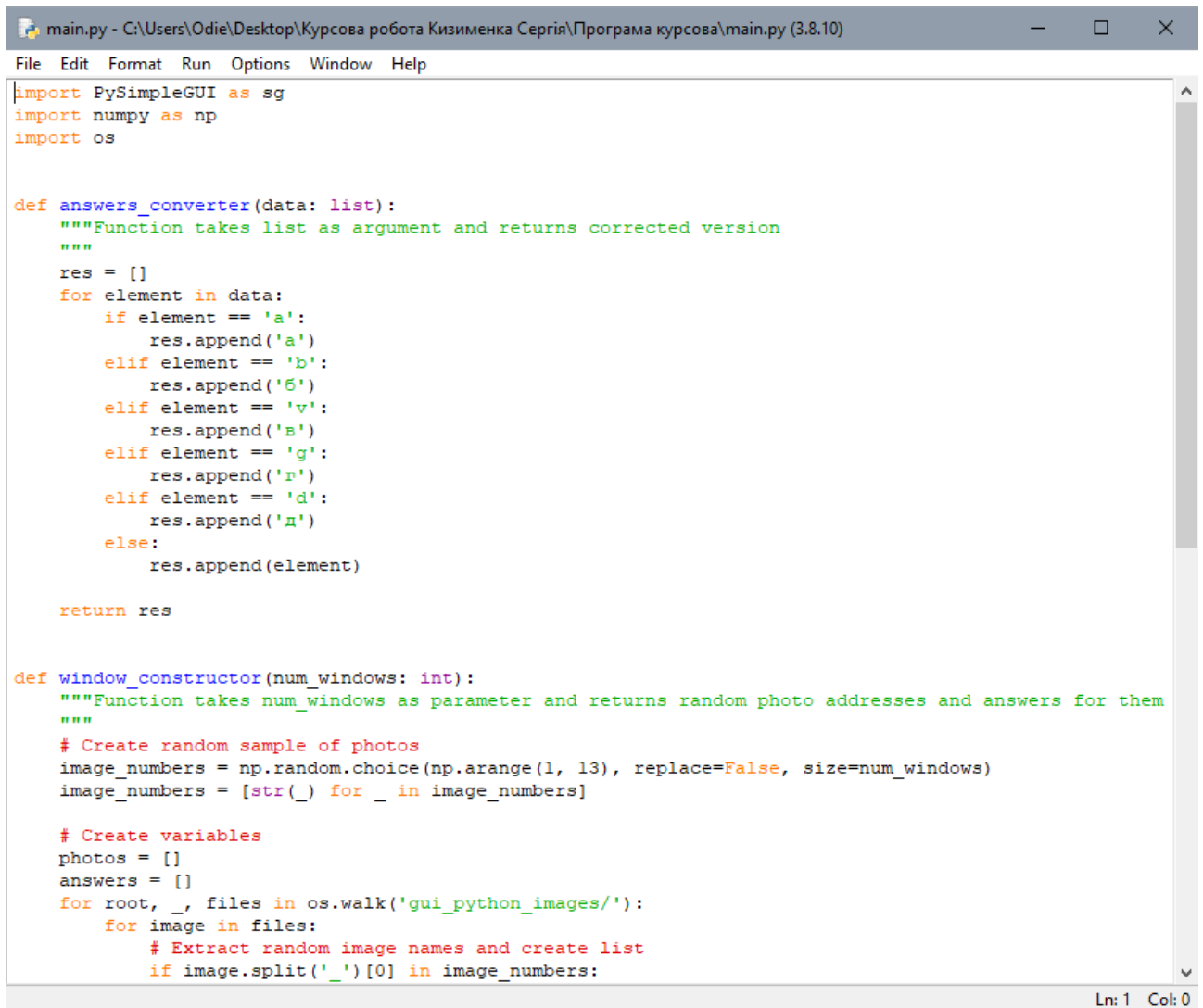


Рисунок 4.2 – процес створення нового файлу

Після створення нового файлу в середовищі розробки, далі розробляється програмний код тренажеру (рис. 4.3 – рис. 4.4)



```
main.py - C:\Users\Odie\Desktop\Курсова робота Кизименка Сергія\Програма курсова\main.py (3.8.10)
File Edit Format Run Options Window Help
import PySimpleGUI as sg
import numpy as np
import os

def answers_converter(data: list):
    """Function takes list as argument and returns corrected version
    """
    res = []
    for element in data:
        if element == 'a':
            res.append('а')
        elif element == 'b':
            res.append('б')
        elif element == 'v':
            res.append('в')
        elif element == 'g':
            res.append('г')
        elif element == 'd':
            res.append('д')
        else:
            res.append(element)

    return res

def window_constructor(num_windows: int):
    """Function takes num_windows as parameter and returns random photo addresses and answers for them
    """
    # Create random sample of photos
    image_numbers = np.random.choice(np.arange(1, 13), replace=False, size=num_windows)
    image_numbers = [str(_) for _ in image_numbers]

    # Create variables
    photos = []
    answers = []
    for root, _, files in os.walk('gui_python_images/'):
        for image in files:
            # Extract random image names and create list
            if image.split('_')[0] in image_numbers:
```

Ln: 1 Col: 0

Рисунок 4.3 – программный код тренажеру.


```

main.py - C:\Users\Odie\Desktop\Курсова робота Кизименка Сергія\Програма курсова\main.py (3.8.10)
File Edit Format Run Options Window Help

    if image.split('_')[0] in image_numbers:
        new_photo = f"gui_python_images/{image}"
        photos.append(new_photo)

        # Extract answers from image name
        answer = image.split('_')[1].split('.')[0]
        answers.append(answer)

        # Convert answers into ukrainian
        answers = answers_converter(answers)
    return photos, answers

photos, answers = window_constructor(12)

# Our GUI
layout = [
    [sg.Image(filename=f"{photos[0]}", key='-IMAGE-')], # Photo with geometry exercise
    [sg.Text("Виберіть правильну відповідь: ")],
    [sg.Text("Ваша відповідь"), sg.In(size=(25, 1), enable_events=True, key="-ANSWER-")], # Panel for
    [sg.Button('Перевірити')] # Check button
]

window = sg.Window(title='Тест', layout=layout)

print(photos)

while True:
    event, values = window.read()
    if event == "Exit" or event == sg.WIN_CLOSED:
        break
    if event == 'Перевірити':
        answer = values['-ANSWER-']
        if answer == answers[0]:
            if len(photos) != 1:
                del photos[0]
                del answers[0]
                window['-IMAGE-'].update(filename=photos[0])
            else:
                break

```

Ln: 1 Col: 0

Рисунок 4.4 – програмний код тренажеру.

4.2. Опис роботи тренажеру

Після відкриття програми тренажеру користувачу відкривається інтерфейс з запитаннями (рис. 4.5)

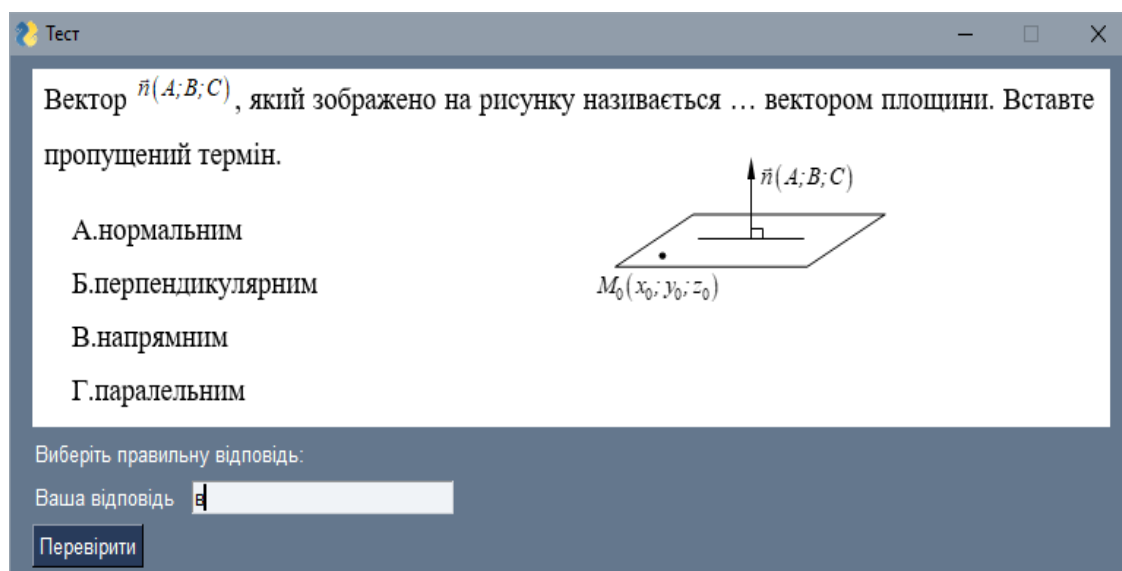


Рисунок 4.5 – інтерфейс з запитаннями.

Користувачу відкривається питання, він може вказати свою відповідь в рядку «Ваша відповідь:» варіант відповіді на питання (На відповіді відповідаємо тільки буквою) Після вибору відповіді, студент(ка) натискає кнопку «Перевірити», якщо вибрана відповідь правильна, то програма автоматично дає друге запитання (рис. 4.6)

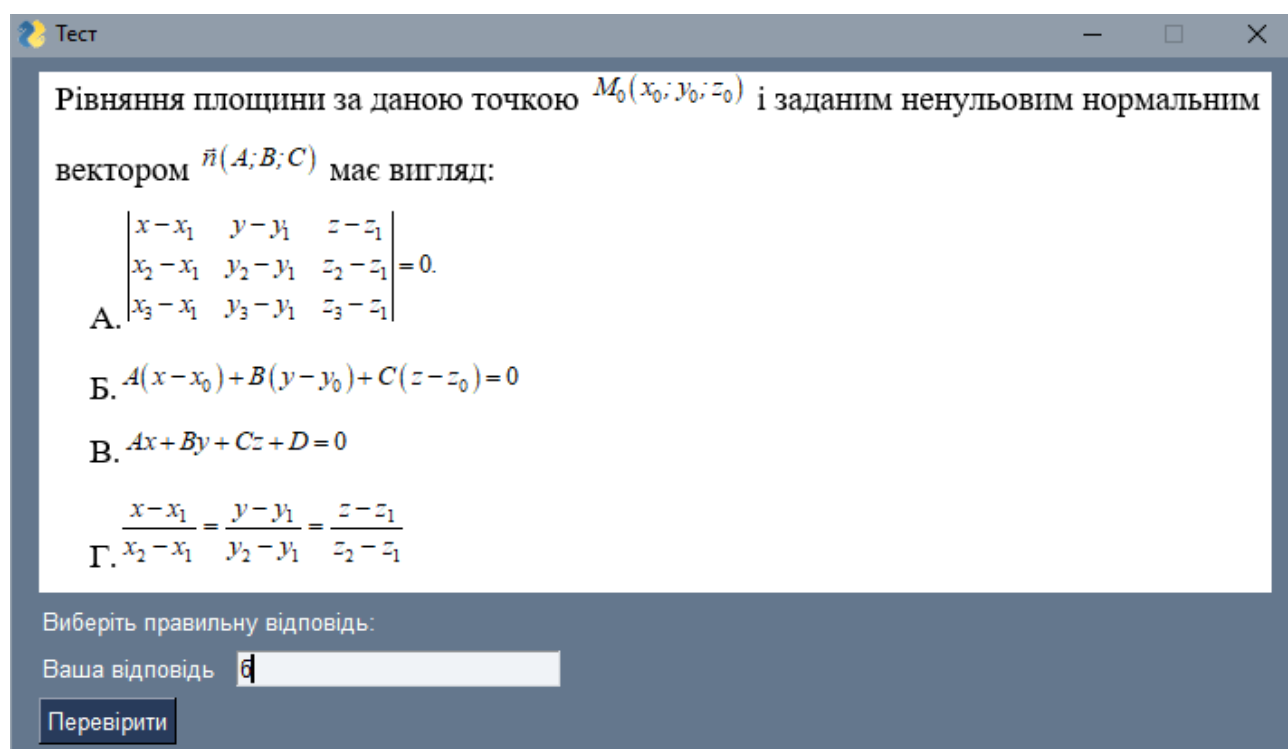
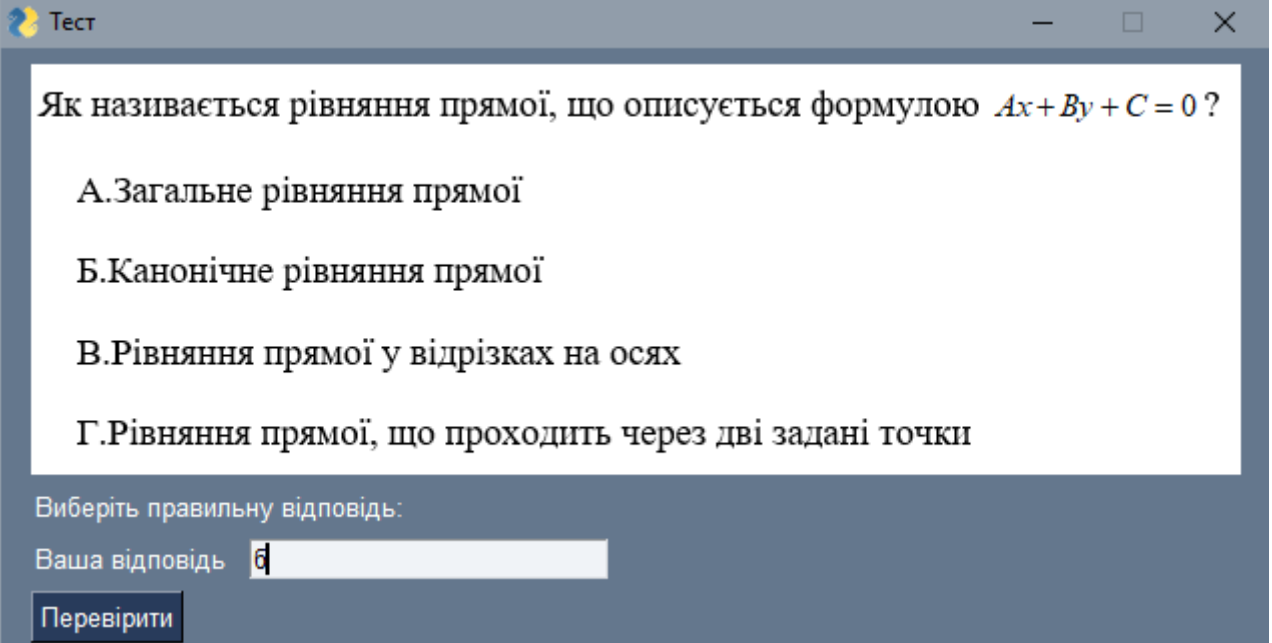


Рисунок 4.6 – перше питання.

Якщо студент вибрав неправильний варіант відповіді, програма не дає наступне запитання, а чекає поки користувач не дасть вірну відповідь (рис. 4.7)



Тест

Як називається рівняння прямої, що описується формулою $Ax + By + C = 0$?

- А. Загальне рівняння прямої
- Б. Канонічне рівняння прямої
- В. Рівняння прямої у відрізках на осях
- Г. Рівняння прямої, що проходить через дві задані точки

Виберіть правильну відповідь:

Ваша відповідь

Рисунок 4.7 – вікно очікування правильної відповіді.

ВИСНОВКИ

Отже, дистанційне навчання надає здобувачам освіти доступ до навчання в будь-яких умовах, підвищує рівень самостійності студентів, надає можливість опанувати навчальну програму. Викладачам в свою чергу за допомогою сучасних технологій дозволяє реалізувати процес викладання свого предмету, контролювати учнів, перевіряти роботи в зручному форматі. До недоліків дистанційного навчання в умовах воєнного часу можна віднести наступне: нестабільне підключення до мережі Інтернет, віялові відключення електропостачання, в поодиноких ситуаціях відсутність смартфона або будь-якого комп'ютерного пристрою в студентів, які знаходяться в скрутних ситуаціях. Важливо пам'ятати, що в обов'язковому форматі навчання в період воєнного стану повинна бути психолого-педагогічна підтримка для кожного учасника освітнього процесу.

Програма призначена для тренажеру «Аналітична геометрія на площині» дистанційного навчального курсу «Прикладна математика». Застосувати програму можливо тільки на комп'ютерах, що здатні виконувати файли формату .exe. Основний файл буде виконуватись тільки за наявності папки `python_gui_images` із її вмістом в одній директорії із ним. Програма становить візуальний інтерфейс із можливістю взаємодії із ним.

Для розробки застосунку була використана мова Python та бібліотеки SimpleGUI, numpy, os

У дипломній роботі виконано поставлені завдання:

1. Постановка задачі;
2. Використання геометрії на площині;
3. Застосування аналітичної геометрії на площині;
4. Алгоритм роботи тренажеру з теми «Аналітична геометрія на площині» дистанційного навчального курсу «Прикладна математика»;
5. Блок-схема тренажеру;
6. Опис процесу програмної реалізації;

7. Опис роботи тренажеру.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Аналітична геометрія на площині [Електронний ресурс]. – Режим доступу до ресурсу: URL: https://nmetau.edu.ua/file/kaplmath_17942.pdf – Назва з екрану.
2. Аналітична геометрія на площині [Електронний ресурс]. – Режим доступу до ресурсу: URL: https://org2.knuba.edu.ua/pluginfile.php/45991/mod_resource/content/1/Пряма%20на%20площині.pdf – Назва з екрану.
3. Python [Електронний ресурс]. – Режим доступу до ресурсу: URL: [https://uk.wikipedia.org/wiki/Python#:~:text=Python%20\(найчастіше%20вживане%20прочитання%20—%20«1990%20році%20Гвідо%20ван%20Россумом](https://uk.wikipedia.org/wiki/Python#:~:text=Python%20(найчастіше%20вживане%20прочитання%20—%20«1990%20році%20Гвідо%20ван%20Россумом) – Назва з екрану
4. Python [Електронний ресурс]. – Режим доступу до ресурсу: URL: <https://www.python.org> – Назва з екрану
5. Sublime text [Електронний ресурс]. – Режим доступу до ресурсу: URL: https://uk.wikipedia.org/wiki/Sublime_Text – Назва з екрану
6. Sublime Text Коротке керівництво [Електронний ресурс]. – Режим доступу до ресурсу: URL: https://pns.hneu.edu.ua/pluginfile.php/773083/mod_resource/content/1/Sublime%20Text%20-%20Краткое%20руководство%20-%20CoderLessons.pdf – Назва з екрану
7. Visual Studio Code [Електронний ресурс]. – Режим доступу до ресурсу: URL: https://uk.wikipedia.org/wiki/Visual_Studio_Code – Назва з екрану
8. Знайомство з Visual Studio Code [Електронний ресурс]. – Режим доступу до ресурсу: URL: <https://romul.name/blog/znayomstvo-z-visual-studio-code/>
9. IDLE Shell [Електронний ресурс]. – Режим доступу до ресурсу: URL: <https://ru.wikipedia.org/wiki/IDLE> – Назва з екрану
10. Classtime [Електронний ресурс]. – Режим доступу до ресурсу: URL: <https://www.classtime.com/compare/uk/classtime-vs-naurok/> – Назва з екрану
11. На Урок [Електронний ресурс]. – Режим доступу до ресурсу: URL: <https://mooc4ua.online/products/3> – Назва з екрану

12. Найкращі платформа для оцінювання знань [Електронний ресурс]. – Режим доступу до ресурсу: URL: <https://www.classtime.com/compare/uk/переваги-класстайм/> – Назва з екрану
13. Kahoot! [Електронний ресурс]. – Режим доступу до ресурсу: URL: https://osvita.ua/vnz/high_school/73080/ – Назва з екрану
14. Moodle [Електронний ресурс]. – Режим доступу до ресурсу: URL: <https://uk.wikipedia.org/wiki/Moodle> – Назва з екрану
15. Claroline LMS [Електронний ресурс]. – Режим доступу до ресурсу: URL: <https://e-asveta.adu.by/index.php/distancionni-vseobuch/obuchenie-online/sredstva-dlya-organizatsii-obucheniya/56-platformi-dlya-sdo/120-claroline> – Назва з екрану

ДОДАТОК А. ВИХІДНІ КОДИ

```
import PySimpleGUI as sg
import numpy as np
import os

def answers_converter(data: list):
    """Function takes list as argument and returns corrected version
    """
    res = []
    for element in data:
        if element == 'a':
            res.append('a')
        elif element == 'b':
            res.append('б')
        elif element == 'v':
            res.append('в')
        elif element == 'g':
            res.append('г')
        elif element == 'd':
            res.append('д')
        else:
            res.append(element)

    return res

def window_constructor(num_windows: int):
```


"""Function takes num_windows as parameter and returns random photo addresses and answers for them

"""

Create random sample of photos

```
image_numbers = np.random.choice(np.arange(1, 13), replace=False,
size=num_windows)
```

```
image_numbers = [str(_) for _ in image_numbers]
```

Create variables

```
photos = []
```

```
answers = []
```

```
for root, _, files in os.walk('gui_python_images/')
```

```
    for image in files:
```

```
        # Extract random image names and create list
```

```
        if image.split('_')[0] in image_numbers:
```

```
            new_photo = f"gui_python_images/{image}"
```

```
            photos.append(new_photo)
```

```
        # Extract answers from image name
```

```
        answer = image.split('_')[1].split('.')[0]
```

```
        answers.append(answer)
```

```
# Convert answers into ukrainian
```

```
answers = answers_converter(answers)
```

```
return photos, answers
```

```
photos, answers = window_constructor(12)
```

```
# Our GUI
```

```

layout = [
    [sg.Image(filename=f"{photos[0]}", key='-IMAGE-')], # Photo with geometry
exercise
    [sg.Text("Виберіть правильну відповідь: ")],
    [sg.Text("Ваша відповідь"), sg.In(size=(25, 1), enable_events=True, key="-
ANSWER-")], # Panel for answers
    [sg.Button('Перевірити')] # Check button
]

```

```

window = sg.Window(title='Тест', layout=layout)

```

```

print(photos)

```

```

while True:

```

```

    event, values = window.read()

```

```

    if event == "Exit" or event == sg.WIN_CLOSED:

```

```

        break

```

```

    if event == 'Перевірити':

```

```

        answer = values['-ANSWER-']

```

```

        if answer == answers[0]:

```

```

            if len(photos) != 1:

```

```

                del photos[0]

```

```

                del answers[0]

```

```

                window['-IMAGE-'].update(filename=photos[0])

```

```

            else:

```

```

                break

```