

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ВОЛИНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ ІМЕНІ ЛЕСІ УКРАЇНКИ**

Кафедра експериментальної фізики, інформаційних та освітніх технологій

На правах рукопису

САВЧЕНКО ВОЛОДИМИР АНАТОЛІЙОВИЧ

**МОДЕЛЮВАННЯ У ПРОЦЕСІ НАВЧАННЯ ФІЗИКИ В СТАРШІЙ
ШКОЛІ**

Спеціальність: 014.08 «Середня освіта (Фізика та астрономія)»

Освітньо-професійна програма «Середня освіта. Фізика»

Робота на здобуття освітнього ступеня «Магістр»

Науковий керівник:

МУЛЯР ВАДИМ ПЕТРОВИЧ,

кандидат педагогічних наук,

доцент кафедри експериментальної фізики,
інформаційних та освітніх технологій

РЕКОМЕНДОВАНО ДО ЗАХИСТУ

Протокол № ____

засідання кафедри експериментальної фізики,
інформаційних та освітніх технологій

від _____ 2025 р.

Завідувач кафедри

_____ проф. Галян В. В.

ЛУЦЬК – 2025

ЗМІСТ

ВСТУП	5
РОЗДІЛ 1. ТЕОРЕТИЧНІ ЗАСАДИ ВИКОРИСТАННЯ МОДЕЛЮВАННЯ У НАВЧАННІ ФІЗИКИ.....	10
1.1. Моделювання як метод наукового пізнання	10
1.2. Психолого-педагогічні особливості сприйняття фізичних моделей учнями старшого шкільного віку	14
1.3. Класифікація моделей у навчанні фізики: матеріальні, знакові, комп'ютерні	16
1.4. Огляд сучасного стану впровадження комп'ютерного моделювання в освіті (аналіз вітчизняного та зарубіжного досвіду)	19
РОЗДІЛ 2. ПРОЄКТУВАННЯ ТА ПРОГРАМНА РЕАЛІЗАЦІЯ КОМП'ЮТЕРНОЇ МОДЕЛІ ОПТИЧНИХ ЯВИЩ.....	24
2.1. Аналіз інструментарію та обґрунтування вибору мови Python для шкільного курсу фізики.....	24
2.2. Побудова фізико-математичної моделі геометричної оптики	27
2.3. Алгоритмізація та програмна реалізація моделі заломлення світла... ..	29
РОЗДІЛ 3. РОЗРОБКА МЕТОДИЧНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ ВИВЧЕННЯ ОПТИКИ ЗАСОБАМИ PYTHON.....	36
3.1. Методика використання комп'ютерної моделі під час викладання нового матеріалу (демонстраційний експеримент)	36
3.2. Розробка інструктивно-методичних матеріалів для виконання лабораторних робіт	38
3.3. Організація навчально-дослідницької та проєктної діяльності учнів	40
ВИСНОВКИ.....	45
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	48
ДОДАТКИ.....	50

АНОТАЦІЯ

САВЧЕНКО В. А.

МОДЕЛЮВАННЯ У ПРОЦЕСІ НАВЧАННЯ ФІЗИКИ В СТАРШІЙ ШКОЛІ

У роботі досліджено дидактичні можливості моделювання в освітньому процесі з фізики у старшій школі. Актуальність дослідження зумовлена тим, що моделювання є невід'ємним складником освітнього процесу як в середній та вищій школі. Причиною цього є те, що процес формування знань у здобувачів освіти пов'язаний з перетворенням у їх свідомості одних моделей у інші, які є більш наближеними до абсолютної істини. Окрім того, моделювання дає змогу вчителю розкрити важливе місце методів наукового дослідження в освітньому процесі з фізики, глибше розкрити на уроці сутність багатьох фізичних процесів і явищ. Наукова новизна одержаних результатів полягає у подальшому розвитку методики навчання фізики через поглиблення міжпредметних зв'язків з інформатикою, візуалізацію абстрактних фізичних понять за допомогою динамічних комп'ютерних моделей, створенні програмного засобу, який дозволяє моделювати хід променів на межі двох середовищ зі змінними параметрами. У роботі розкрито роль моделювання як методу пізнання, проаналізовано психолого-педагогічні особливості сприйняття моделей старшокласниками та здійснено огляд сучасного стану впровадження ІТ в освіту. Обґрунтовано вибір інструментарію (Python), описано математичний апарат моделі та наведено етапи її програмної реалізації (алгоритм, код, інтерфейс). Запропоновано методику використання створеної програми на різних етапах уроку, розроблено інструкції до лабораторних робіт та окреслено перспективи проєктної діяльності учнів.

Ключові слова: фізика, освітній процес, моделювання, комп'ютерна модель, програмне забезпечення, віртуальна лабораторна робота, Python, STEM-проєкт.

ABSTRACT

SAVCHENKO V. A.

MODELING IN THE PROCESS OF TEACHING PHYSICS IN UPPER SECONDARY SCHOOL

The study explores the didactic potential of modeling in the educational process of teaching physics in upper secondary school. The relevance of the research is determined by the fact that modeling is an integral component of the educational process both in secondary and higher education. This is due to the fact that the process of knowledge formation among learners is associated with the transformation of certain models in their consciousness into others that are closer to absolute truth. Furthermore, modeling enables the teacher to reveal the significant role of scientific research methods in physics education and to provide a deeper understanding of numerous physical processes and phenomena during lessons.

The scientific novelty of the results lies in the further development of physics teaching methodology through the enhancement of interdisciplinary connections with computer science, visualization of abstract physical concepts using dynamic computer models, and the creation of software that allows simulation of ray trajectories at the boundary of two media with variable parameters. The paper highlights the role of modeling as a method of cognition, analyzes the psychological and pedagogical features of students' perception of models, and reviews the current state of IT integration in education. The choice of tools (Python) is justified, the mathematical framework of the model is described, and the stages of its software implementation (algorithm, code, interface) are presented. A methodology for using the developed program at different stages of the lesson is proposed, instructions for laboratory work are provided, and prospects for students' project-based activities are outlined.

Keywords: physics, educational process, modeling, computer model, software, virtual laboratory work, Python, STEM project.

ВСТУП

Актуальність теми дослідження. Початок XXI століття ознаменувався переходом людства до постіндустріальної епохи, яку часто називають епохою цифрової економіки або Індустрією 4.0. Ці глобальні зміни диктують нові вимоги до системи освіти, яка повинна готувати молодь до життя в умовах невизначеності, динамічних змін технологій та стрімкого зростання інформаційних потоків. Реформування освітньої галузі України, що відбувається в рамках реалізації Концепції «Нова українська школа» (НУШ), спрямоване на зміну освітньої парадигми: від репродуктивної передачі знань до формування компетентностей – динамічної комбінації знань, умінь, навичок та цінностей.

Відповідно до Державного стандарту базової середньої освіти [22], метою освіти є всебічний розвиток особистості учня. Стандарт виокремлює 11 ключових компетентностей, серед яких для нашого дослідження пріоритетними є компетентності у галузі природничих наук, техніки і технологій, інформаційно-цифрова компетентність, а також математична компетентність. Особливої ваги у Державному стандарті набувають так звані «наскрізні вміння», які є спільними для всіх компетентностей: читання з розумінням, вміння висловлювати власну думку, критичне та системне мислення, здатність логічно обґрунтовувати позицію, творчість, ініціативність. Використання програмування на уроках фізики є потужним інструментом розвитку саме наскрізних умінь. Коли учень не просто розв'язує задачу, а створює її програмну модель, він проявляє ініціативність (шукає шляхи реалізації), творчість (оформлює інтерфейс), критичне мислення (аналізує адекватність моделі) та логіку (будує алгоритм). Таким чином, інтеграція фізики та інформатики безпосередньо реалізує державну освітню політику.

Важливим чинником, що зумовлює актуальність теми, є психологічні особливості когнітивного розвитку сучасних підлітків. Як зазначає у своїх працях С. П. Величко [3], формування абстрактного мислення старшокласників є складним процесом, який потребує якісної педагогічної підтримки. Фізика як наука оперує ідеалізованими об'єктами (матеріальна точка, ідеальний газ,

світловий промінь), які не існують у реальності в чистому вигляді. Розділ «Геометрична оптика» вимагає від учнів високого рівня просторової уяви та здатності до абстрагування. Учні стикаються з проблемою розриву між математичним записом закону (формула) та його реальним фізичним змістом. За відсутності наочності закон Снеліуса сприймається формально, як набір символів для підстановки чисел. Без візуалізації учні не розуміють фізику, втрачають мотивацію та інтерес до предмета. Комп'ютерне моделювання виступає тим «когнітивним містком», який поєднує абстрактну теорію з чуттєвим сприйняттям.

Глобалізація освітнього простору спонукає до врахування кращих світових практик. Аналіз освітніх систем країн-лідерів рейтингу PISA, зокрема Фінляндії та Сінгапуру, демонструє чітку тенденцію до інтеграції кодування (coding) у викладання природничих дисциплін. Так, у Фінляндії, згідно з дослідженнями Й. Лавонена [19], програмування не є ізольованим предметом, а наскрізним інструментом, що використовується на уроках математики та фізики. Це дозволяє формувати цілісну картину світу, де ІТ є універсальною мовою опису процесів. Аналогічний підхід спостерігається у Сінгапурі, де стратегія «ICT in Education Masterplan 4» [20] наголошує на розвитку навичок моделювання та симуляції як основи для підготовки майбутніх інноваторів.

Актуальність дослідження посилюється вимогами сучасного ринку праці. Роботодавці дедалі частіше очікують наявності ІТ-навичок не лише від програмістів, а й від інженерів, науковців, аналітиків, технологів. Вміння написати скрипт для обробки даних, побудувати графік засобами Python чи змодельовати експеримент стає базовою грамотністю фахівця технічного профілю. Впровадження Python у курс фізики виконує важливу профорієнтаційну функцію, підвищуючи престиж інженерно-технічних професій.

Попри очевидну необхідність цифровізації, вчителі фізики часто стикаються з проблемою вибору інструментарію. Готові пропрієтарні програмні продукти часто є платними, закритими для модифікації та ставлять учня в

пасивну позицію. Табличні процесори мають обмежені можливості для динамічної візуалізації складної геометрії. Вільна мова програмування Python з бібліотеками NumPy та Matplotlib є оптимальним рішенням. Вона доступна, безкоштовна, вивчається в курсі інформатики та має простий синтаксис. Однак, методичне забезпечення її використання саме на уроках фізики розроблено недостатньо, що й зумовило вибір теми магістерської роботи: «Моделювання у процесі навчання фізики у старшій школі».

Об'єкт дослідження – освітній процес з фізики у закладах загальної середньої освіти.

Предмет дослідження – методика створення та використання комп'ютерних моделей оптичних явищ засобами мови програмування Python під час вивчення геометричної оптики.

Мета дослідження полягає у теоретичному обґрунтуванні, практичній розробці та методичному забезпеченні використання комп'ютерних моделей, створених мовою Python, для формування предметної та інформаційно-цифрової компетентностей учнів.

Для досягнення поставленої мети визначено такі завдання дослідження:

1. Здійснити аналіз психолого-педагогічної, методичної та технічної літератури з проблеми впровадження комп'ютерного моделювання в освітній процес.
2. Визначити дидактичні можливості мови програмування Python та бібліотек NumPy і Matplotlib для моделювання фізичних процесів у закладах загальної середньої освіти.
3. Розробити математичну модель явищ геометричної оптики (заломлення світла, повне внутрішнє відбивання) та реалізувати її у вигляді програмного коду мовою Python.
4. Створити методичне забезпечення для використання розробленої моделі: сценарій демонстраційного експерименту, інструкцію до лабораторної роботи та тематику навчальних проєктів.

5. Обґрунтувати ефективність запропонованої методики для розвитку дослідницьких навичок учнів.

Методи дослідження. Для розв'язання поставлених завдань використано комплекс методів дослідження:

– теоретичні: аналіз наукової, навчально-методичної літератури та нормативних документів для з'ясування стану досліджуваної проблеми; порівняння та систематизація існуючих підходів до комп'ютерного моделювання; моделювання фізичних процесів для побудови математичного апарату дослідження;

– емпіричні: педагогічне спостереження за навчальним процесом; вивчення передового педагогічного досвіду впровадження STEM-технологій;

– практичні (експериментальні): алгоритмізація та програмування моделі мовою Python, тестування програмного забезпечення, розробка навчально-методичних матеріалів.

Наукова новизна одержаних результатів полягає у:

– *подальшому розвитку* методики навчання фізики через поглиблення міжпредметних зв'язків з інформатикою, що передбачає використання мови програмування не як об'єкта вивчення, а як інструменту наукового пізнання;

– *удосконаленні* підходів до візуалізації абстрактних фізичних понять (показник заломлення, граничний кут повного відбивання) за допомогою динамічних комп'ютерних моделей;

– *обґрунтуванні* доцільності використання бібліотек наукового стеку Python (NumPy, Matplotlib) у шкільному курсі фізики як альтернативи закритим пропрієтарним симуляціям.

Практичне значення одержаних результатів визначається тим, що:

1. Розроблено діючий програмний продукт (скрипт мовою Python), який дозволяє моделювати хід променів на межі двох середовищ зі змінними параметрами. Програма має інтуїтивно зрозумілий інтерфейс і не потребує глибоких знань програмування від користувача (вчителя чи учня).

2. Створено комплект методичного забезпечення, який включає: сценарій проведення уроку з використанням комп'ютерної демонстрації, інструктивні матеріали для виконання віртуальної лабораторної роботи та перелік тем для навчально-дослідницьких проєктів.

3. Матеріали дослідження можуть бути використані вчителями фізики та інформатики закладів загальної середньої освіти під час вивчення теми «Оптика», проведення факультативних занять, гурткової роботи та реалізації STEM-проєктів.

Структура та обсяг роботи. Магістерська робота складається зі вступу, трьох розділів, висновків до кожного розділу, загальних висновків, списку використаних джерел та додатків.

РОЗДІЛ 1. ТЕОРЕТИЧНІ ЗАСАДИ ВИКОРИСТАННЯ МОДЕЛЮВАННЯ У НАВЧАННІ ФІЗИКИ

1.1. Моделювання як метод наукового пізнання

Моделювання в наукових дослідженнях, яке почали застосовувати ще в глибоку давнину, охоплює нині все нові й нові сфери наукових знань. Однак методологія моделювання впродовж тривалого часу розвивалась незалежно від інших наук. Була відсутня єдина система понять, єдина термінологія. Лише згодом почали усвідомлювати роль моделювання як універсального методу наукового пізнання.

Термін «модель» широко використовується в різних сферах діяльності людини і має безліч семантичних значень. Розглядатимемо тільки такі моделі, котрі є інструментами для одержання нових знань.

Термін «модель» походить від латинського слова «modulus» – зразок, норма, міра. Модель – це спрощене, але точне уявлення об'єкта, процесу або явища, яке використовується для його вивчення, дослідження або створення. Це може бути матеріальна копія (макет, глобус), інформаційна модель (схема, карта, опис) або інший аналог, що зберігає найважливіші властивості оригіналу.

Математична модель – це абстракція реальної дійсності (світу), в якій відношення між реальними елементами, а саме ті, що цікавлять дослідника, замінені відношеннями між математичними категоріями. Ці відношення зазвичай подаються у формі рівнянь і/чи нерівностей, відношеннями формальної логіки між показниками (змінними), які характеризують функціонування реальної системи, що моделюється.

Фізична модель – це прототип або зменшена копія реального об'єкта, яка використовується для експериментального вивчення фізичних явищ, коли їх важко або недоцільно досліджувати в реальних масштабах [1]. Вона дозволяє вивчати об'єкти та процеси шляхом їх відтворення в зменшеному вигляді або за допомогою інших, подібних до них процесів тієї ж фізичної природи.

Неможливо уявити собі сучасну науку, зокрема техніку та природознавство, без широкого застосування фізичного моделювання. Сутність цієї методології полягає в заміні вихідного об'єкта його «матеріальним двійником» – фізичною моделлю – і подальшим вивченням (дослідженням) моделі на підставі вимірювальних методів та експериментальних процедур, які реалізуються за допомогою лабораторного обладнання. Робота не із самим об'єктом (явищем, процесом), а з його моделлю дає можливість відносно швидко і безболісно досліджувати його основні (суттєві) властивості та поведінку за будь-яких імовірних ситуацій.

Водночас лабораторні (стендові, натурні) експерименти з моделями об'єктів дозволяють, спираючись на точність сучасних вимірювальних методів і технічного інструментарію, ретельно та досить глибоко вивчати об'єкт у достатньо детальному вигляді, що недоступно суто теоретичним підходам. Не дивно, що методологія фізичного моделювання продовжує розвиватися, охоплюючи аналіз надзвичайно складних технічних і природних процесів.

У наш час фізичне моделювання входить у важливий етап свого розвитку, інтегруючись у структури сучасного експериментального дослідження. Прогрес засобів вимірювання, фіксації та обробки даних відповідає сучасним тенденціям. Без отримання достовірних фізичних даних не варто й думати про розв'язання дедалі більш складних проблем, які постають перед наукою. Однак експериментальні дані самі по собі здебільшого мало що дають без аналізу. Необхідні надійні способи опрацювання цієї «сировини» в готовий «продукт», тобто в точні знання. Історія методології фізичного моделювання переконує: вона може й повинна бути матеріальним базисом новітніх технологій.

Технічні, технологічні, природничі та інші системи, що їх вивчає сучасна наука, все меншою мірою піддаються дослідженню (в необхідній комплексності та точності) лише теоретичними розрахунками. Безпосередній натурний експеримент над оригіналом часто є надто тривалим, дорогим, небезпечним чи просто неможливим (наприклад, краш-тести або випробування космічних

апаратів). Тому фізичне моделювання є неминучою складовою науково-технічного прогресу.

Уже сама постановка питання щодо фізичного моделювання будь-якого об'єкта породжує чіткий план дій, який умовно можна поділити на три етапи: модель–методика–експеримент (рис.1).

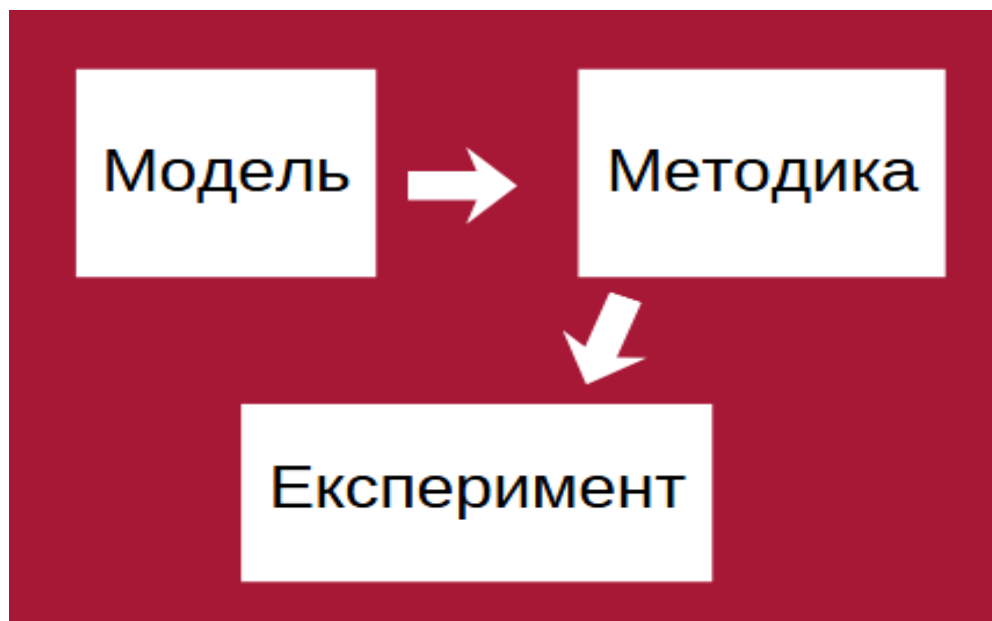


Рис. 1. Узагальнена схема фізичного моделювання

На першому етапі обирається (чи будується) матеріальний «еквівалент» об'єкта, що відтворює фізично найважливіші (ключові) його властивості – закони подібності, яким він підпорядковується, геометричні та фізичні параметри, що притаманні складовим його частинам тощо. Фізична модель (макет) перевіряється на відповідність критеріям подібності, що дозволяє отримати базу для отримання нових знань про об'єкт.

Другий етап – вибір (чи розроблення) методики для реалізації дослідження на стенді чи в лабораторії. Модель розміщується в умовах, зручних для застосування вимірювальних приладів, визначається послідовність фізичних дій та вимірів, котрі необхідно здійснити, щоб отримати шукані величини із заданою точністю. Експериментальні методики не повинні спотворювати основні властивості моделі, а отже, вихідного об'єкта (оригіналу), бути економними та адаптивними щодо особливостей обладнання.

На третьому етапі проводяться безпосередні експерименти (випробування), що «переносять» модель і методику в площину реальної фізичної взаємодії. До них також висуваються вимоги точності та повторюваності. Це і є дослідження матеріального еквівалента об'єкта, придатного для безпосереднього вимірювання параметрів.

Створивши тріаду: «модель–методика–експеримент», дослідник отримує універсальний, наочний і відносно недорогий інструмент, який тестується в «пробних» дослідах. Після того як адекватність (достатній рівень відповідності критеріям подібності) тріади щодо вихідного об'єкта засвідчена, з моделлю проводять різноманітні та детальні «досліди», які дають нову інформацію про необхідні якісні та кількісні властивості й характеристики об'єкта. Процес моделювання супроводжується поліпшенням та уточненням, за необхідності, конструкції моделі або умов експерименту.

Як методологія фізичне моделювання не підміняє собою теорію, розрахунки та інші дисципліни, не конкурує з ними. Навпаки, важко переоцінити його підтверджуючу роль. Створення та застосування тріади можливе лише за умови використання різноманітних технічних засобів. Воно дає додаткові стимули різним напрямам прикладної науки.

У широкому аспекті моделювання наявне майже в усіх видах інженерної активності людей. Привнесення в ці сфери точного експериментального знання допомагає обмежити інтуїтивне конструювання, розширює межі застосування раціональних методів. Звичайно ж, фізичне моделювання плідне лише за умови виконання професійних вимог: чітке дотримання теорії подібності, калібрування приладів, аналіз похибок, щоб пересвідчитися в адекватності використовуваних моделей. Якщо ж аналізувати проблеми моделювання складних систем, де діють стохастичні (випадкові) чинники, то до цих вимог необхідно додати ще низку, зокрема, акуратне відтворення зовнішніх умов середовища, завбачливе застосування вимірювальної апаратури (пріоритетним є шлях «від задачі до приладу», а не навпаки) та інші.

Розв'язуючи сучасні наукові проблеми, було б наївним покладати надію лише на теоретичні розрахунки. Постійне вдосконалення тріади фізичного моделювання та її впровадження у сучасні лабораторно-дослідні комплекси – методологічний імператив. Лише його виконання дає можливість отримати таку необхідну високотехнологічну, надійну та безпечну матеріальну продукцію.

Зазначимо, що умовою розробки моделі є принцип так званої фізичної подібності. Це означає, що дослідник повинен мати достатньо чітке уявлення про те, які фізичні параметри є визначальними для досліджуваної системи, які сили та чинники суттєво впливають на процес її функціонування. Якщо рівень забезпечення подібності низький, то створити модель, за допомогою котрої можна було б отримати достовірні знання про об'єкт-оригінал, майже неможливо.

1.2. Психолого-педагогічні особливості сприйняття фізичних моделей учнями старшого шкільного віку

прийняття фізичних моделей учнями старшого шкільного віку має виразні психолого-педагогічні особливості, пов'язані з переходом від конкретно-образного до абстрактно-логічного мислення. У цей період школярі вже здатні оперувати символічними репрезентаціями – формулами, графіками, схемами, – однак ефективність роботи з моделями залежить від того, наскільки вчитель забезпечує зв'язок між абстракціями та реальними фізичними явищами. Модель у навчанні виконує подвійну функцію: вона є інструментом пізнання та водночас засобом формування вмінь моделювання, тому її використання має враховувати вікові особливості учнів і обмеження їхньої робочої пам'яті [3].

Українські методичні підходи підкреслюють, що формування здатності працювати з фізичними моделями потребує системності: поєднання теоретичного аналізу, лабораторних досліджень і практичних задач сприяє формуванню стійких наукових уявлень. Важливо також застосовувати диференціацію навчання, опорні схеми та поетапне ускладнення моделей, що дозволяє зменшити когнітивне навантаження та забезпечити поступове

входження учнів у складні абстракції [4]. Пізнавальні труднощі старшокласників часто пов'язані з мовно-концептуальними бар'єрами: метафоричні вислови на кшталт «струм тече» або «тепло переходить» полегшують первинне розуміння, але можуть закріплювати хибні причинно-наслідкові уявлення. Тому педагог має працювати з термінологією, пояснювати межі застосовності метафор і забезпечувати поступовий перехід від наочного до символічного рівня через проміжні репрезентації – схеми, графіки, структурні моделі, що рекомендовано сучасними методичними матеріалами [5].

Для виявлення стартових ментальних моделей учнів доцільно застосовувати концептуальні тести, інтерв'ю з поясненням явищ, аналіз розв'язків задач, побудову концепт-мап і завдання типу Predict-Observe-Explain. Такі інструменти дозволяють виявити розрив між інтуїтивними уявленнями та науковими поясненнями, а регулярна діагностика допомагає будувати диференційовані навчальні траєкторії та планувати корекційні педагогічні втручання, що підтверджується українськими методичними розробками [6]. Корекція помилкових моделей у старшокласників ґрунтується на створенні когнітивного конфлікту, використанні реальних і віртуальних експериментів, застосуванні рефутаційних текстів та метакогнітивних практик – письмових пояснень, рефлексії, взаємооцінювання. Дослідження підкреслюють, що моделювання має бути не лише демонстрацією вчителя, а діяльністю учнів, яка включає побудову власних моделей, їх перевірку та аргументацію, що сприяє глибшому переосмисленню фізичних явищ [7]. Ефективна робота з моделями на уроці передбачає послідовний цикл: діагностику стартових уявлень, постановку проблеми та створення когнітивного конфлікту, побудову моделі (схеми, графіка, симуляції), експериментальну перевірку, рефлексію та формулювання оновленої моделі, а також формувальне оцінювання для фіксації змін у розумінні. Такий цикл описаний у сучасних українських методичних посібниках як оптимальний для розвитку компетентностей у фізиці [8].

Цифрові симуляції, віртуальні лабораторії та AR/VR-технології значно розширюють можливості моделювання, дозволяючи візуалізувати невидимі або

складні процеси – електричні поля, хвильові явища, мікроскопічні моделі. Вони сприяють корекції інтуїтивних помилок, але потребують педагогічного обґрунтування, адаптації до вікових особливостей та методичного супроводу, про що зазначається в сучасних оглядах методики навчання фізики [9].

Оцінювання ефективності роботи з моделями має враховувати не лише рівень знань, а й структуру уявлень: зміни в концептуальних тестах, якість усних і письмових пояснень, трансформацію концепт-мап, поведінку під час експериментів. Українські дослідники наголошують на необхідності порівняльних досліджень традиційних лабораторій і AR/VR-симуляцій, а також вивчення впливу метакогнітивних стратегій на формування фізичних моделей у старшокласників [10].

1.3. Класифікація моделей у навчанні фізики: матеріальні, знакові, комп'ютерні

Моделювання є одним із ключових методів наукового пізнання та фундаментальною складовою навчання фізики, оскільки воно дозволяє замінити складний реальний об'єкт його спрощеним аналогом для зручнішого вивчення та аналізу [11]. Цей процес виступає містком між теорією та реальністю, дозволяючи учням переходити від емпіричних спостережень до теоретичних узагальнень. У сучасній дидактиці фізики моделі прийнято класифікувати на три основні групи: матеріальні, знакові та комп'ютерні, кожна з яких виконує свою унікальну функцію та відповідає певному етапу формування фізичних понять [12].

Першою групою є матеріальні (або предметні) моделі – це реально існуючі об'єкти, які відтворюють геометричну форму, структуру або динаміку оригінального явища. Їхня головна перевага полягає у наочності та сенсорній доступності, оскільки вони мають фізичну природу, тотожну або відмінну від оригіналу, але зберігають суттєві для вивчення властивості. Використання таких моделей є необхідним на етапі первинного сприйняття матеріалу. Класичним прикладом такої моделі в механіці є пружинний маятник (Рис.2).



Рис.2. Пружинний маятник, як приклад матеріальної моделі

Це не просто вантаж на пружині, а фізична ідеалізація, що дозволяє учням безпосередньо спостерігати коливальний рух, вимірювати період та амплітуду, формуючи базове розуміння природи механічних коливань ще до введення складного математичного апарату [11]. Матеріальні моделі допомагають учням сформувати чуттєвий образ явища, що є критично важливим для уникнення формалізму в знаннях.

Друга група – це знакові (інформаційні) моделі, де властивості об'єктів представлені за допомогою умовних знаків, символів, графіків або рисунків. Цей вид моделювання вимагає від здобувача освіти розвиненого абстрактного мислення, оскільки реальний об'єкт замінюється його інформаційним кодом

[13]. До графічних знакових моделей належать, наприклад, схеми електричних кіл, де реальні прилади (джерела струму, споживачі, вимикачі) замінено стандартизованими умовними позначеннями (Рис.3).

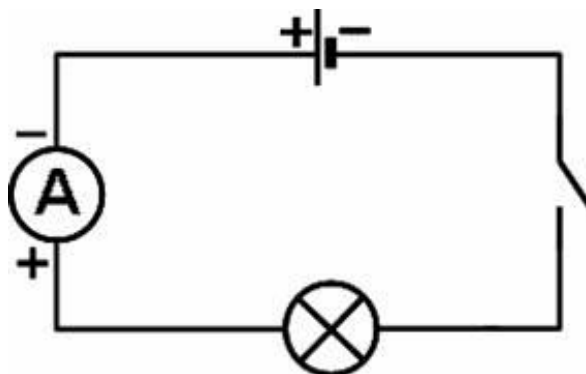


Рис.3. Схема електричного кола, знакова модель

Це дозволяє аналізувати шлях проходження струму та логіку з'єднань без відволікання на конструктивні особливості дротів чи зовнішній вигляд елементів. Найвищим рівнем знакових моделей є математичні моделі – формули та рівняння, які описують кількісні залежності між фізичними величинами (Рис.4.). Саме формули перетворюють якісний опис явища на точний інструмент для розрахунків і передбачень, формалізуючи фізичні закони мовою математики [13].

$$A = F \cdot s$$

Рис.4. Формула, приклад знакової моделі

Третім, найбільш динамічним видом, що набув особливого значення в умовах цифровізації освіти, є комп'ютерні моделі. Це програмні засоби, які імітують фізичні процеси на екрані пристрою, органічно поєднуючи в собі риси знакових моделей (через використання програмних алгоритмів) та матеріальних (через візуалізацію динаміки процесів) [14]. Особливу цінність у навчальному процесі мають інтерактивні симуляції, такі як моделі з платформи PhET. Вони

дозволяють реалізувати концепцію «обчислювального експерименту»: змінювати вхідні параметри системи (наприклад, масу тіла, коефіцієнт тертя, напругу чи опір) і миттєво спостерігати реакцію системи (Рис.5).

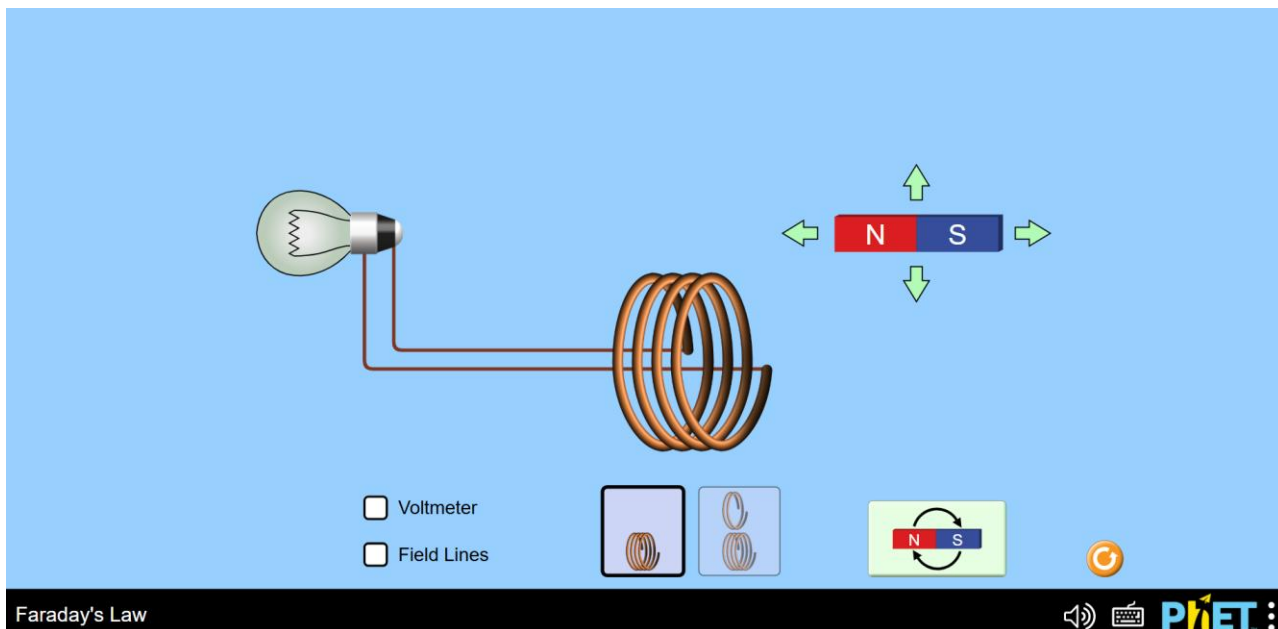


Рис.5. Симуляція закону Фарадея

Це дає змогу досліджувати процеси, які неможливо або небезпечно відтворити в умовах шкільної лабораторії, а також візуалізувати невидимі для ока об'єкти [15]. Найбільша ефективність навчання досягається при системній інтеграції всіх трьох типів моделей, що забезпечує перехід від конкретно-образного до абстрактно-логічного мислення.

1.4. Огляд сучасного стану впровадження комп'ютерного моделювання в освіті (аналіз вітчизняного та зарубіжного досвіду)

Сучасний етап розвитку освітніх технологій характеризується зміною парадигми від пасивного споживання інформації до активного конструювання знань, де комп'ютерне моделювання (КМ) виступає одним із ключових інструментів. Аналіз світового та вітчизняного досвіду дозволяє виявити як спільні тенденції, так і суттєві відмінності у підходах до імплементації КМ у навчальний процес.

Аналіз зарубіжного досвіду. В провідних країнах світу комп'ютерне моделювання вже інтегроване в навчальні плани як обов'язкова складова STEM-освіти та розвитку обчислювального мислення (Computational Thinking).

У Сполучених Штатах Америки впровадження КМ регламентується стандартами NGSS (Next Generation Science Standards), де моделювання визначено як одна з восьми ключових наукових практик. Американський підхід базується на концепції *Model-Based Inquiry*: учні не просто спостерігають за готовими симуляціями, а самостійно розробляють моделі для перевірки гіпотез [16]. Дослідження, проведені розробниками платформи PhET Interactive Simulations (Університет Колорадо), доводять, що поєднання фізичного експерименту з віртуальним моделюванням є найбільш ефективним для формування концептуального розуміння фізичних явищ [17].

У Великій Британії з 2014 року діє оновлена національна навчальна програма, яка ввела обчислювальне мислення як наскрізну навичку. Особливістю британського досвіду є тісна інтеграція програмування (переважно мовою Python) з природничими науками. Учні середньої школи використовують скрипти для обробки даних реальних експериментів та побудови математичних моделей процесів, що стирає межу між інформатикою та фізикою [18].

Досвід Фінляндії демонструє інтеграцію КМ через методику *Phenomenon-Based Learning* (навчання на основі явищ). Фінські педагоги використовують середовища динамічного моделювання (наприклад, GeoGebra, Algodoo) не ізольовано, а в контексті вирішення комплексних міждисциплінарних проблем, таких як енергоефективність або екологія [19].

У країнах Азії, зокрема в Сінгапурі, акцент зроблено на використанні високотехнологічних віртуальних лабораторій у рамках ініціативи «Smart Nation». Сінгапурська методика «Explore-Explain-Apply» використовує КМ на етапі дослідження (Explore) для візуалізації процесів, недоступних для прямого

спостереження (нанотехнології, астрофізика), що сприяє глибшому розумінню абстрактних понять [20].

Загальноєвропейський контекст визначається рамкою цифрових компетентностей DigCompEdu, яка рекомендує використання моделювання для персоналізації навчання та впровадження імерсивних технологій (VR/AR) [21].

Аналіз вітчизняного досвіду. В Україні стан впровадження комп'ютерного моделювання характеризується динамічним розвиток, проте має певні диспропорції порівняно із західними практиками. Нормативною основою є Державний стандарт базової середньої освіти в рамках реформи «Нова українська школа» (НУШ), де в інформатичній освітній галузі виокремлено змістову лінію «Моделювання» [22]. Однак, на практиці, у масовій школі часто спостерігається використання КМ переважно як демонстраційного засобу (заміна навчального фільму), а не інструменту дослідження. Значний прогрес демонструє система позашкільної освіти, зокрема Мала академія наук України (МАН). Науково-дослідницькі роботи учнів-членів МАН часто базуються на створенні власних складних моделей з використанням мов програмування C++, Python або інженерних пакетів, що відповідає високим світовим стандартам [23]. Це свідчить про наявність значного інтелектуального потенціалу, який, однак, поки що слабо інтегрований у загальноосвітній процес. Суттєвим поштовхом до впровадження КМ стали виклики дистанційного навчання під час пандемії та воєнного стану. Через неможливість доступу до шкільних лабораторій, вчителі природничих дисциплін масово почали використовувати віртуальні симуляції (PhET, MozaBook). Проте, як зазначають дослідники С. Семеріков та І. Теплицький, основною проблемою залишається методична неготовність частини педагогів організувати дослідницьку діяльність учнів у віртуальному середовищі, зводячи роботу до виконання інструкцій за алгоритмом [24]. Також варто відзначити дефіцит україномовних методичних розробок, які б поєднували сучасний програмний інструментарій із чинними навчальними програмами, хоча

останніми роками з'являється все більше адаптованих ресурсів від вітчизняних науковців та практиків [25].

Порівняльний аналіз свідчить, що в той час як зарубіжний досвід орієнтований на *створення* моделей учнями (coding to learn) та наскрізну інтеграцію, вітчизняна практика на даному етапі сфокусована переважно на *використанні* готових моделей для компенсації відсутності реального обладнання. Перспектива розвитку української освіти полягає у переході від споживацької моделі використання КМ до дослідницько-конструкторської.

З'ясовано, що моделювання є не лише універсальним методом наукового пізнання, а й ефективним дидактичним засобом. У навчанні фізики воно виконує низку важливих функцій: пізнавальну (джерело нових знань), ілюстративну (візуалізація явищ), евристичну (пошук розв'язків) та систематизуючу. Використання моделей дозволяє спростити складні фізичні об'єкти, зберігаючи їхні ключові властивості для вивчення. Визначено, що учні старшого шкільного віку (15–17 років) перебувають на етапі розвитку абстрактно-логічного мислення. Проте для глибокого розуміння фізичних процесів, які неможливо спостерігати безпосередньо (мікросвіт, астрономічні явища, швидкоплинні процеси), їм необхідна опора на наочні образи. Моделювання виступає необхідною ланкою, що поєднує абстрактну теорію з чуттєвим сприйняттям, сприяючи формуванню наукового світогляду. Проаналізовано класифікацію навчальних моделей, серед яких виділяють матеріальні (предметні), знакові (математичні, графічні) та комп'ютерні. Встановлено, що в сучасній освіті відбувається інтеграція цих видів, де особливого значення набувають комп'ютерні моделі. Вони поєднують в собі наочність матеріальних моделей та абстрактність знакових, надаючи можливість динамічного керування процесом. Огляд вітчизняного та зарубіжного досвіду засвідчив стійку тенденцію до цифровізації фізичної освіти. Використання комп'ютерного моделювання (віртуальні лабораторії, інтерактивні симуляції типу PhET, середовища

програмування) дозволяє подолати обмеження традиційного шкільного експерименту. Водночас виявлено потребу в розробці чітких методичних рекомендацій для вчителів щодо доцільного та збалансованого використання комп'ютерних моделей на уроках фізики.

Теоретичний аналіз підтверджує, що моделювання є потужним інструментом модернізації навчання фізики, проте його ефективність залежить від методично грамотного поєднання з традиційними формами навчання та врахування психологічних особливостей старшокласників.

РОЗДІЛ 2. ПРОЄКТУВАННЯ ТА ПРОГРАМНА РЕАЛІЗАЦІЯ КОМП'ЮТЕРНОЇ МОДЕЛІ ОПТИЧНИХ ЯВИЩ

2.1. Аналіз інструментарію та обґрунтування вибору мови Python для шкільного курсу фізики.

Відповідно до Державного стандарту базової середньої освіти [22], ключовим завданням сучасної школи є формування не лише предметних, а й цифрових компетентностей учнів. Впровадження комп'ютерного моделювання у процес навчання фізики дозволяє реалізувати діяльнісний підхід, де учень виступає не пасивним споживачем інформації, а активним дослідником [23].

Як зазначають С. О. Семеріков та І. О. Теплицький [24], комп'ютерне моделювання є містком між абстрактною фізичною теорією та реальним експериментом. Для реалізації моделі оптичних явищ (законів заломлення та відбивання світла) нами було обрано мову програмування *Python*. Цей вибір ґрунтується на аналізі педагогічної доцільності та технічних можливостей.

Обґрунтування вибору мови Python

На відміну від використання готових симуляцій, таких як PhET [15], самостійне створення або модифікація програмного коду сприяє формуванню обчислювального мислення (computational thinking), що є важливою складовою сучасної освіти [25].

Вибір Python зумовлений такими факторами:

1. **Міжпредметні зв'язки.** Python є однією з основних мов у шкільному курсі інформатики. Використання її на уроках фізики дозволяє реалізувати наскрізні змістові лінії та показати прикладне значення програмування [18].

2. **Доступність та універсальність.** Синтаксис мови є зрозумілим для старшокласників, що знижує когнітивне навантаження при написанні коду і дозволяє зосередитися на фізичній суті явища, а не на технічних деталях реалізації [14].

3. Гнучкість візуалізації. Порівняно з табличними процесорами (наприклад, Excel), Python надає значно ширші можливості для побудови геометричних моделей ходу променів у реальному часі.

Аналіз використаних бібліотек

Для програмної реалізації моделі геометричної оптики було використано стандартний стек бібліотек для наукових обчислень, що дозволяє створити «віртуальну лабораторію» [9]. У розробленій програмі застосовуються наступні модулі:

1. Модуль математичних обчислень NumPy

Бібліотека numpy є стандартом для наукових розрахунків.

```
import numpy as np
```

У контексті нашої моделі вона використовується для роботи з тригонометричними функціями (`np.sin`, `np.arcsin`, `np.cos`), які є основою математичного запису закону Снеліуса:

$$n_1 \sin \alpha = n_2 \sin \gamma$$

Використання numpy дозволяє оперувати масивами даних, що забезпечує миттєвий перерахунок траєкторії променя при зміні параметрів середовища.

2. Модуль візуалізації Matplotlib

Для графічного відображення фізичних процесів використано бібліотеку matplotlib.

```
import matplotlib.pyplot as plt
```

Вона дозволяє будувати межі поділу середовищ, перпендикуляр та самі промені у декартовій системі координат. Як зазначає С. М. Савгира [14], візуалізація є критично важливою для формування абстрактних понять, таких як «оптична густина» чи «граничний кут повного внутрішнього відбивання».

3. Реалізація інтерактивності (Matplotlib Widgets)

Для перетворення статичного графіка на динамічну модель було використано модуль віджетів:

```
from matplotlib.widgets import Slider, TextBox
```

Це дозволило реалізувати інструментарій, аналогічний реальним лабораторним установкам:

– **Slider (Повзунок):** Дозволяє плавно змінювати кут падіння променя, спостерігаючи перехід від заломлення до повного внутрішнього відбивання.

– **TextBox (Текстове поле):** Надає можливість вводити точні значення показників заломлення (n_1 , n_2) для різних речовин (води, скла, алмазу), що необхідно для проведення лабораторних робіт з високою точністю.

Таким чином, обраний інструментарій (Python + NumPy + Matplotlib) повністю відповідає вимогам до сучасних засобів ІКТ підтримки навчально-дослідницької діяльності [23] та дозволяє створити ефективне середовище для вивчення геометричної оптики (Рис.6).

```

1 import numpy as np
2 import matplotlib.pyplot as plt
3 from matplotlib.widgets import Slider, TextBox
4
5 # --- Налаштування фізики ---
6 def calculate_refraction(theta1_deg, n1, n2):
7     """
8     Розраховує кут заломлення за законом Снеліуса.
9     Повертає кут у градусах або None, якщо виникає повне внутрішнє відбиття
10    """
11    theta1_rad = np.radians(theta1_deg)
12
13    # Закон Снеліуса: n1 * sin(t1) = n2 * sin(t2)
14    # sin(t2) = (n1 / n2) * sin(t1)
15    sin_theta2 = (n1 / n2) * np.sin(theta1_rad)
16
17    # Перевірка на повне внутрішнє відбиття (якщо sin > 1, розв'язку немає)
18    if abs(sin_theta2) > 1.0:
19        return None
20
21    theta2_rad = np.arcsin(sin_theta2)

```

Рис.6. Фрагмент програмного коду: підключення бібліотек та реалізація функції розрахунку кута заломлення

2.2. Побудова фізико-математичної моделі геометричної оптики

Розглянемо заломлення світла в разі його переходу з повітря у воду: α – кут падіння β – кут відбивання γ (гамма) – кут заломлення (Рис.7).



Рис.7. Заломлення світла в разі його переходу з повітря у воду

Першу згадку про заломлення світла можна знайти в працях давньогрецького філософа Арістотеля (4 ст. до н.е.), який замислювався над питанням: «Чому палиця, занурена у воду, здається переламаною?» А ось кількісний закон, що описує заломлення світла, був установлений Зміну напрямку поширення світла на межі поділу двох середовищ називають заломленням світла. Заломлений промінь – це промінь, що задає напрямок заломленого пучка світла. Кут заломлення (γ (гамма)) – це кут, утворений заломленим променем і перпендикуляром до межі поділу двох середовищ, проведеним із точки падіння променя З експериментально в 1621 р. голландським природознавцем Віллебрордом Снелліусом (1580–1626) й отримав назву закон Снелліуса [2] (Рис.8).

Закони заломлення світла (закони Снелліуса):

1 закон: Промінь падаючий, промінь заломлений і перпендикуляр до межі поділу двох середовищ, проведений із точки падіння променя, лежать в одній площині.

2 закон: Для двох даних середовищ відношення синуса кута падіння до синуса кута заломлення є величиною незмінною:

(– відносний показник заломлення)

Відносний показник заломлення показує, у скільки разів швидкість поширення світла в середовищі 1 більша (або менша), ніж швидкість поширення світла в середовищі 2:

Рис.8. Закони Снелліуса

Світло, переходячи з одного середовища і інше, змінює свій напрямок. В різних середовищах світло поширюється з різною швидкістю, але завжди повільніше, аніж у вакуумі.

У будь-якому іншому середовищі швидкість світла є меншою, ніж у вакуумі (наприклад, у воді швидкість світла в 1,33 разу менша, ніж у вакуумі, а у повітрі швидкість поширення світла в 1,7 разів більша, аніж у склі). Саме зміна швидкості поширення світла в разі його переходу з одного прозорого середовища в інше є причиною заломлення світла. Прийнято говорити про оптичну густину середовища: чим більшою є оптична густина середовища, тим меншою є швидкість поширення світла в цьому середовищі.

Абсолютний показник заломлення середовища – це фізична величина, яка характеризує оптичну густину середовища і показує, у скільки разів швидкість поширення світла в середовищі менша, ніж у вакуумі. Абсолютний показник заломлення залежить від фізичного стану середовища (температури, густини, тиску та ін.) і від частоти світлової хвилі. Тому в таблицях вказують, зазвичай,

або стан середовища, або середній показник заломлення для даного діапазону довжин хвиль.

Зв'язок між абсолютним та відносним показниками заломлення

$$n_{21} = \frac{n_2}{n_1}$$

Закон заломлення світла дає змогу пояснити цікаве й практично важливе явище – повне відбиття світла. Якщо промінь світла поширюється від точкового джерела з оптично більш густого в оптично менш густе середовище, то кут заломлення більший за кут падіння. Зі збільшенням кута падіння світлового променя збільшується і кут заломлення, водночас, інтенсивність заломленого променя зменшується. Досягши певного значення кута падіння α_0 , промінь взагалі не перетинає межу поділу середовищ – кут заломлення становить 90° . Коли кут падіння $\alpha > \alpha_0$ світло не переходить у інше середовище, воно лише відбивається від його межі, тобто відбувається явище повного внутрішнього відбиття.

Повне внутрішнє відбиванням світла – явище, за якого заломлення світла відсутнє, тобто світло повністю відбивається від середовища з меншою оптичною густиною.

2.3. Алгоритмізація та програмна реалізація моделі заломлення світла

Процес створення комп'ютерної моделі розпочинається з розробки алгоритму – точного припису послідовності дій для виконавця (комп'ютера). Для реалізації інтерактивної моделі заломлення світла нами було розроблено лінійний алгоритм з розгалуженням, який працює у циклічній залежності від дій користувача (подієво-орієнтований підхід).

Логіку роботи програми можна представити у вигляді наступної послідовності етапів:

- 1. Ініціалізація:** Підключення бібліотек, створення графічного вікна, налаштування осей координат, встановлення початкових значень ($n_1=1.0$, $n_2=1.5$, $\alpha=45^\circ$).

2. **Очікування події:** Програма очікує зміни положення повзунка (слайдера) користувачем.

3. **Зчитування даних:** Отримання нових значень кута падіння та показників заломлення.

4. **Розрахунок (Блок фізики):** Обчислення синуса кута заломлення.

5. **Логічна перевірка (Розгалуження):**

– *Умова:* Чи модуль синуса кута заломлення > 1 ?

– *ТАК (Повне внутрішнє відбивання):* Кут заломлення не існує. Будується лише відбитий промінь.

– *НІ (Звичайне заломлення):* Обчислюється $\arcsin$, знаходиться кут γ . Будується заломлений промінь.

6. **Візуалізація:** Оновлення координат ліній на графіку та текстових підписів.

Графічне представлення алгоритму наведено на рисунку 9

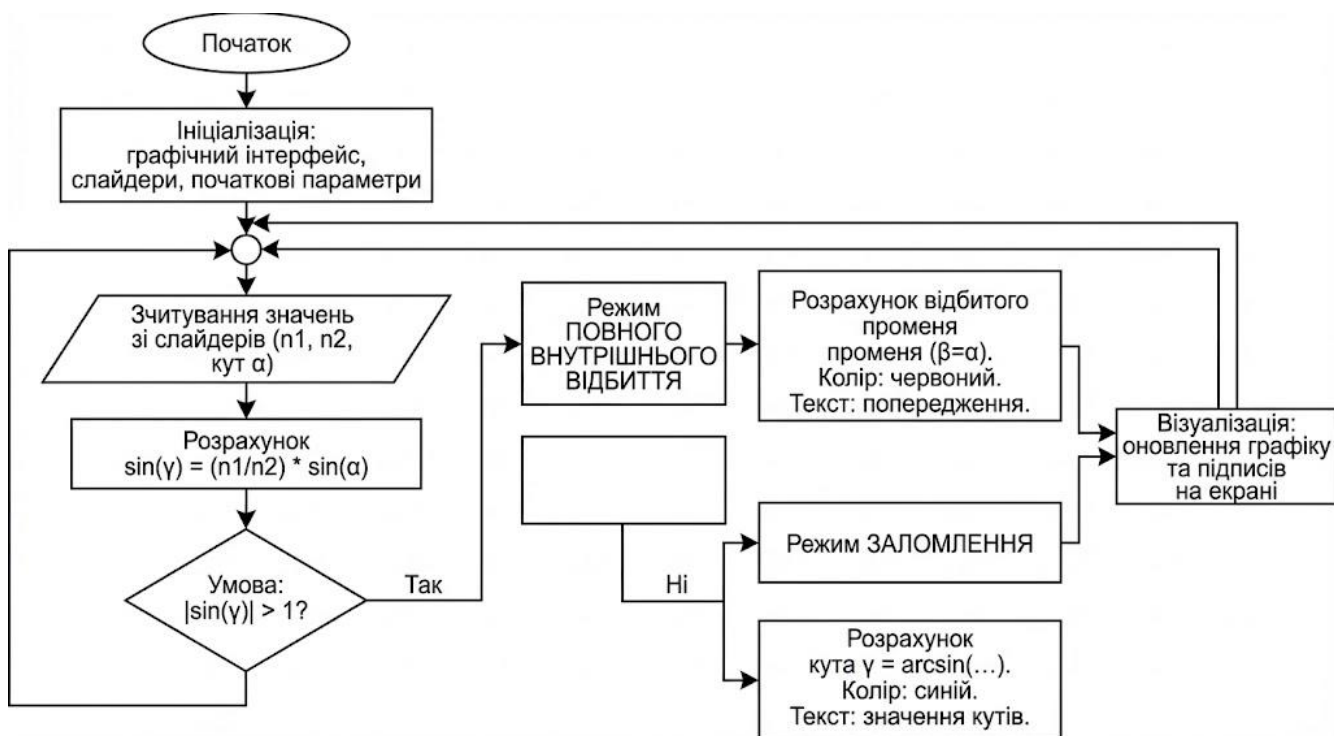


Рис. 9. Блок-схема алгоритму роботи програми

Опис структури програмного коду

Програмна реалізація виконана мовою Python і структурно поділяється на три логічні блоки: математичне ядро, налаштування інтерфейсу та функція обробки подій.

1. Функція фізичних розрахунків

Ключовим елементом коду є функція `calculate_refraction`, яка приймає вхідні параметри середовища та повертає результат взаємодії світла з межею поділу (Рис.10).

```
# --- Налаштування фізики ---
def calculate_refraction(theta1_deg, n1, n2):
    """
    Розраховує кут заломлення за законом Снеліуса.
    Повертає кут у градусах або None, якщо виникає повне внутрішнє відбиття.
    """
    theta1_rad = np.radians(theta1_deg)

    # Закон Снеліуса: n1 * sin(t1) = n2 * sin(t2)
    # sin(t2) = (n1 / n2) * sin(t1)
    sin_theta2 = (n1 / n2) * np.sin(theta1_rad)

    # Перевірка на повне внутрішнє відбиття (якщо sin > 1, розв'язку немає)
    if abs(sin_theta2) > 1.0:
        return None
```

Рис.10. Функція, яка приймає вхідні параметри середовища та повертає результат взаємодії світла з межею поділу

Особливістю реалізації є обробка виняткової ситуації повного внутрішнього відбивання: функція повертає значення `None`, що стає сигналом для графічного блоку змінити режим відображення.

2. Налаштування графічного середовища

Візуалізація реалізована через об'єкт `fig` бібліотеки `Matplotlib`. Для наочності використано метод `ax.fill_between`, який зафарбовує області середовищ різними кольорами (блакитним та фіолетовим), що допомагає учням візуально розрізняти оптичні густини речовин (Рис.11).

```

# --- Налаштування графіки ---
fig, ax = plt.subplots(figsize=(10, 7))
plt.subplots_adjust(left=0.1, bottom=0.35) # Місце для слайдерів
ax.set_title("Моделювання Заломлення Світла (Закон Снеліуса)")
ax.set_xlim(-1.5, 1.5)
ax.set_ylim(-1.5, 1.5)
ax.set_aspect('equal')
ax.axis('off') # Ховаємо осі координат для краси

# Малюємо середовища
medium1_poly = ax.fill_between([-2, 2], 0, 2, color='#e0f7fa', alpha=0.3, label='Середовище 1')
medium2_poly = ax.fill_between([-2, 2], -2, 0, color='#e1bee7', alpha=0.3, label='Середовище 2')

# Лінія розділу та нормаль
ax.axhline(0, color='black', linewidth=2) # Межа
ax.axvline(0, color='gray', linestyle='--', alpha=0.7) # Нормаль

# Початкові промені (пустишки, які будемо оновлювати)
incident_ray = ax.plot([], [], 'r-', linewidth=3, label='Падаючий промінь')
refracted_ray = ax.plot([], [], 'b-', linewidth=3, label='Заломлений промінь')
reflected_ray = ax.plot([], [], 'r--', linewidth=1.5, alpha=0.5, label='Відбитий промінь (часті')

# Текстові підписи
text_n1 = ax.text(-1.4, 0.5, '', fontsize=12, color='blue')
text_n2 = ax.text(-1.4, -0.5, '', fontsize=12, color='purple')
text_info = ax.text(0.5, 1.2, '', fontsize=10, bbox=dict(facecolor='white', alpha=0.8))

```

Рис.11. Налаштування графічного середовища

3. Обробка інтерактивності (Event Handling)

Динамічна зміна моделі забезпечується функцією `update(val)`, яка прив'язана до подій зміни слайдерів методом `.on_changed(update)`.

У цьому блоці відбувається перерахунок декартових координат кінців променів. Оскільки у фізиці кути відраховуються від нормалі (перпендикуляра), а в математиці – від додатної півосі X, у кодi використано перетворення:

- Координата X: $x = -\sin(\theta)$
- Координата Y: $y = \cos(\theta)$

Результати роботи програми

Програма коректно моделює як стандартне заломлення світла, так і явище повного внутрішнього відбивання.

На рисунку 12 продемонстровано роботу програми у режимі заломлення, коли світло переходить з повітря ($n_1=1.0$) у скло ($n_2=1.5$). Видно, що кут заломлення менший за кут падіння.

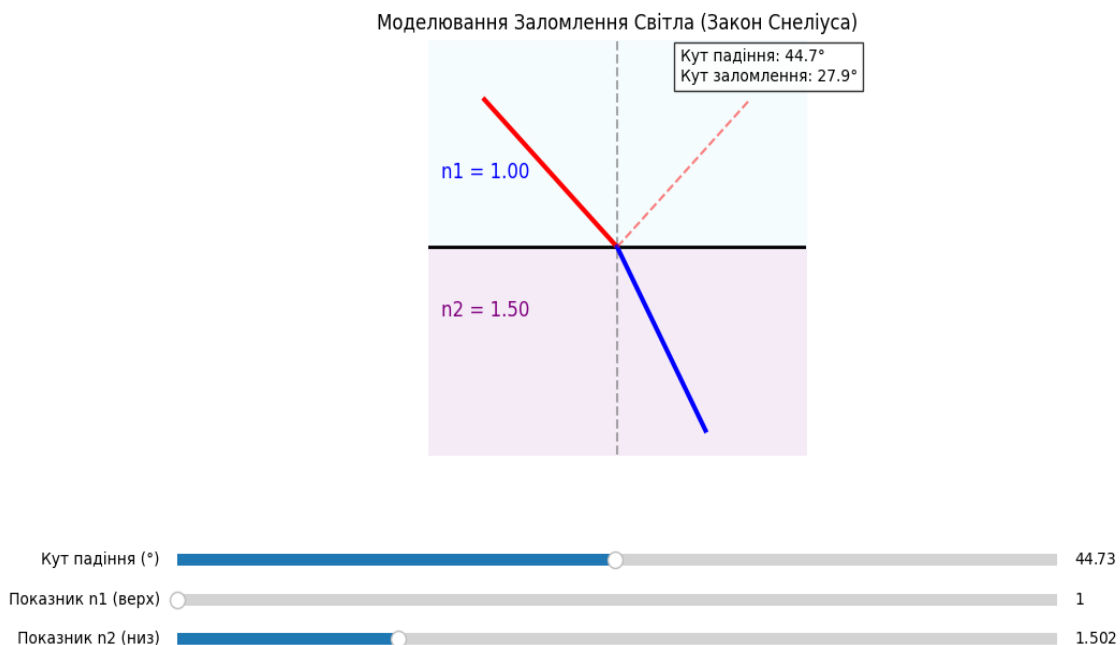


Рис.12. Інтерфейс програми: демонстрація заломлення світла

На рисунку 13 показано випадок повного внутрішнього відбивання (перехід зі скла в повітря, кут падіння великий). Програма автоматично змінює колір променя на червоний та виводить відповідне повідомлення, що є важливим методичним прийомом для привернення уваги учнів.

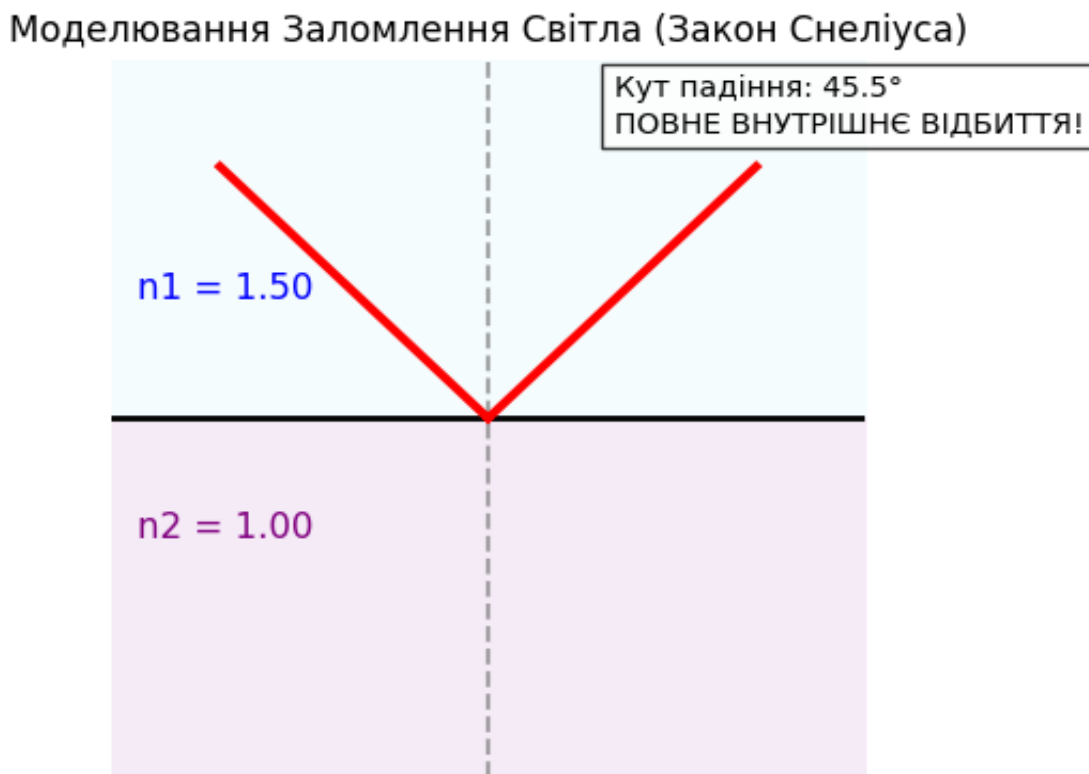


Рис.13. Демонстрація явища повного внутрішнього відбивання

Таким чином, розроблений код забезпечує стабільну роботу моделі, миттєвий відгук на дії користувача та наукову достовірність візуалізації фізичних процесів.

У другому розділі дослідження здійснено теоретичне обґрунтування, проєктування та практичну реалізацію комп'ютерної моделі оптичних явищ. За результатами виконаної роботи можна зробити такі висновки.

Обґрунтовано вибір інструментарію. На основі аналізу сучасних засобів ІКТ показано, що використання мови програмування Python у поєднанні з бібліотеками для наукових обчислень (NumPy) та візуалізації (Matplotlib) є оптимальним підходом для створення навчальних моделей у старшій школі. Такий вибір дозволяє реалізувати потужні міжпредметні зв'язки «Фізика – Інформатика» та сприяє формуванню обчислювального мислення учнів, на відміну від використання готових закритих симуляцій.

Побудовано математичну модель. Здійснено формалізацію законів геометричної оптики. Розроблено математичний апарат, що описує залежність кута заломлення від параметрів середовищ (закон Снеліуса) та визначає умови настання явища повного внутрішнього відбивання. Отримана модель адаптована для програмної реалізації в декартовій системі координат.

Реалізовано програмний продукт. Розроблено подієво-орієнтований алгоритм та написано програмний код інтерактивної моделі. Створений додаток має зручний графічний інтерфейс (слайдери, текстові поля), що дозволяє користувачу в реальному часі змінювати фізичні параметри (n_1 , n_2 , α) та миттєво отримувати візуалізацію ходу променів. Тестування підтвердило, що програма коректно відтворює як звичайне заломлення, так і граничні випадки (повне відбивання), візуально сигналізуючи про зміну фізичного режиму.

Таким чином, розроблена комп'ютерна модель є функціональним, науково достовірним та готовим до використання засобом навчання. Це створює підґрунтя для переходу до наступного етапу дослідження – розробки методичного забезпечення щодо її впровадження в освітній процес з фізики.

РОЗДІЛ 3. РОЗРОБКА МЕТОДИЧНОГО ЗАБЕЗПЕЧЕННЯ ДЛЯ ВИВЧЕННЯ ОПТИКИ ЗАСОБАМИ PYTHON

3.1. Методика використання комп'ютерної моделі під час викладання нового матеріалу (демонстраційний експеримент)

Технічне забезпечення: вчитель запускає скрипт Python на вчительському ноутбучі, який підключено до мультимедійного проєктора або інтерактивної дошки. Завдяки бібліотеці `matplotlib`, інтерфейс є контрастним та добре читається навіть із задніх парт.

Мета демонстрації: візуалізувати абстрактні поняття (заломлення, нормаль, оптична густина) та експериментально підтвердити закони геометричної оптики в режимі реального часу, що неможливо зробити так швидко з традиційними скляними призмами.

Сценарій проведення демонстрації

Етап 1: Актуалізація та знайомство з інтерфейсом

Вчитель запускає програму. На екрані з'являється схема з двома середовищами.

– **Дії вчителя:** виставляє повзунки в базове положення: Кут падіння 45° , $n_1 = 1.0$ (повітря), $n_2 = 1.5$ (скло).

– **Коментар вчителя:** «Погляньте на екран. Ми маємо модель проходження світлового променя через межу двох середовищ. Верхнє (світле) – це повітря, нижнє (темніше) – нехай це буде скло. Червона лінія – це промінь, який падає, синя – той, що пройшов у скло. Пунктирна лінія – це перпендикуляр (нормаль) до межі поділу».

Етап 2: Дослідження залежності кута заломлення від оптичної густини

Вчитель звертає увагу учнів на поведінку променя при переході в більш густе середовище.

– Питання до класу: «Як поводить себе промінь, переходячи з повітря у скло? Він продовжує йти прямо?»

– Очікувана відповідь: Ні, він зламався, наблизився до перпендикуляра.

– Проблемне питання: «А що буде, якщо ми замінимо скло на щось більш оптично густе, наприклад, на алмаз? Показник заломлення алмаза значно вищий ($n \approx 2.4$). Як зміниться хід синього променя?»

– **Дії вчителя:** повільно пересуває слайдер n_2 праворуч до значення 2.4. Учні бачать, як синій промінь різко «притискається» до нормалі.

– Висновок (формулюється разом з учнями): чим більша оптична густина другого середовища, тим сильніше світло заломлюється (кут заломлення зменшується).

Етап 3: Перевірка окремого випадку (нормальне падіння)

– **Питання до класу:** «Чи існує умова, за якої світло перейде в інше середовище і *не* заломиться?»

– **Дії вчителя:** Плавню змінює слайдер Кут падіння до 0° .

– **Демонстрація:** Промінь стає суцільною вертикальною лінією.

– **Коментар вчителя:** «Закон Снеліуса працює, але коли кут падіння нуль, то і кут заломлення нуль. Світло не змінює напрямку».

Етап 4: Оборотноість ходу променів та «Еврика» (Повне внутрішнє відбиття)

Це кульмінація демонстрації, оскільки цей ефект важко показати «на пальцях».

– **Дії вчителя:** Змінює умови на протилежні. Встановлює $n_1 = 1.5$ (скло/вода зверху), а $n_2 = 1.0$ (повітря знизу).

– **Коментар вчителя:** «Уявіть, що ми тепер «сидимо» всередині басейну і світимо ліхтариком назовні, у повітря. Показник заломлення першого середовища більший за друге».

– **Спостереження:** Учні бачать, що синій промінь тепер відхиляється від нормалі (кут заломлення більший за кут падіння).

– **Проблемне питання:** «Уважно слідкуйте за синім променем. Я буду збільшувати кут падіння. Кут заломлення зростає швидше. Що станеться, коли кут заломлення дійде до межі води?»

– **Дії вчителя:** Дуже повільно рухає слайдер Кут падіння. Синій промінь лягає на межу поділу, а потім різко зникає, перетворюючись на червоний відбитий промінь. На екрані з'являється напис: *«ПОВНЕ ВНУТРІШНЄ ВІДБИТТЯ!»*.

– **Реакція класу:** Учні фіксують момент зникнення заломленого променя.

– **Пояснення:** «Ось це і є повне внутрішнє відбиття. Світло більше не може вийти з води, воно повністю відбивається назад, як від дзеркала. Саме на цьому принципі працює оптоволоконний інтернет, який заходить у ваші домівки».

Етап 5: Експеримент «Що якщо?» (Інтерактив)

Вчитель пропонує учням самим задати параметри.

– **Вчитель:** «Хто хоче перевірити ситуацію, коли обидва середовища однакові? Наприклад, олія і спеціальне скло?»

– **Дії:** Викликаний учень або вчитель ставить $n_1 = 1.5$ та $n_2 = 1.5$.

– **Результат:** Промінь йде ідеально рівно, межа поділу оптично зникає.

3.2. Розробка інструктивно-методичних матеріалів для виконання лабораторних робіт

Однією з переваг мови Python є можливість використання створеного коду як віртуальної лабораторної установки [24]. Нами було розроблено інструкційну картку для учнів, яка дозволяє провести дослідження законів геометричної оптики навіть за відсутності реального обладнання.

Нижче наведено структуру та зміст розробленої інструкції.

ІНСТРУКЦІЙНА КАРТКА ДО ЛАБОРАТОРНОЇ РОБОТИ

Тема: Дослідження заломлення світла за допомогою комп'ютерної моделі.

Мета: Експериментально перевірити виконання закону Снеліуса та визначити умови повного внутрішнього відбивання.

Обладнання: Персональний комп'ютер із встановленим середовищем Python (IDLE, Jupyter Notebook або PyCharm), файл програми `optics_model.py`.

Хід роботи

Завдання 1. Перевірка закону заломлення світла

1. Запустіть файл програми. Переконайтеся, що вікно з графіком відображається коректно.
2. Встановіть за допомогою слайдерів наступні параметри середовищ:
 - $n_1 = 1.00$ (повітря);
 - $n_2 = 1.50$ (скло).
3. За допомогою слайдера «Кут падіння» встановіть значення кута α , вказані в таблиці.
4. Зчитайте з екрана значення кута заломлення γ (значення відображається у текстовому полі над графіком).
5. Занесіть дані в таблицю та обчисліть відношення синусів.

№ з/п	Кут падіння α , град	Кут заломлення γ , град	$\sin \alpha$	$\sin \gamma$	$n = \frac{\sin \gamma}{\sin \alpha}$
	30				
	45				
	60				

6. **Висновок до завдання 1:** Порівняйте отримане значення n із встановленим на слайдері показником n_2 . Чи виконується закон Снеліуса?

Завдання 2. Дослідження повного внутрішнього відбивання.

1. Змініть параметри середовищ, щоб світло переходило з оптично більш густого середовища в менш густе:
 - Встановіть $n_1 = 2.00$;
 - Встановіть $n_2 = 1.00$.
2. Плавню збільшуйте кут падіння α від 0° . Спостерігайте за поведінкою синього (заломленого) променя.
3. Зафіксуйте момент, коли заломлений промінь зникне, а програма видасть повідомлення про повне відбивання.
4. Запишіть значення цього граничного кута: $\alpha_{\text{гр}} =$
5. Розрахуйте теоретичне значення за формулою: $\sin \alpha = \frac{n_2}{n_1}$

6. Висновок до завдання 2: Порівняйте отриманий експериментально кут із теоретичним розрахунком.

3.3. Організація навчально-дослідницької та проєктної діяльності учнів

У контексті реформування освітньої галузі та впровадження концепції «Нової української школи», акцент зміщується з накопичення знань на формування компетентностей. Використання мови програмування Python для моделювання фізичних процесів відкриває широкі можливості для організації **STEM-проєктів** (Science, Technology, Engineering, Mathematics), що поєднують фізику, математику та інформатику в єдиний дослідницький комплекс [24, 25]. Організація проєктної діяльності на базі розробленої моделі дозволяє реалізувати рівневу диференціацію: від простих завдань на модифікацію коду (для учнів, що вивчають інформатику на рівні стандарту) до створення складних симуляцій природних явищ (для профільного рівня).

Нижче наведено методику організації трьох різнопланових дослідницьких проєктів, які базуються на програмному ядрі розробленої нами моделі.

Проєкт №1. «Фізика кольору: моделювання дисперсії світла»

Рівень складності: Середній.

Актуальність: базова модель розглядає світло як монохроматичний промінь. Однак у реальності показник заломлення залежить від довжини хвилі (кольори). Цей проєкт дозволяє змодельовати дослід Ньютона з призмою.

– **Фізична постановка задачі:** дослідити явище дисперсії. Пояснити, чому біле світло при проходженні через прозоре середовище розкладається у спектр (веселку). Врахувати, що для червоного кольору показник заломлення скла менший ($n_{\text{red}} \approx 1.51$), а для фіолетового – більший ($n_{\text{violet}} \approx 1.53$).

– Завдання з програмування (Інформатика):

1. Створити структуру даних (словник або масив), що містить параметри кольорів та їхні показники заломлення. Наприклад:

```
colors = {'red': 1.51, 'green': 1.52, 'blue': 1.53}
```

Модифікувати функцію малювання: замість одного виклику функції `calculate_refraction` використати цикл `for`, який проходить по словнику кольорів.

2. Для кожного кольору розрахувати свій унікальний кут заломлення та побудувати лінію відповідного кольору (`plt.plot(..., color='red')`).

– **Очікуваний результат:** на екрані замість одного заломленого променя з'являється «віяло» різнокольорових променів. Учень наочно бачить, що фіолетовий промінь заломлюється найсильніше, а червоний – найслабше.

Проект №2. «Ілюзії природи: моделювання міражів у неоднорідному середовищі»

Рівень складності: Високий (творчий).

Актуальність: проєкт демонструє, що реальні середовища (атмосфера) не завжди є однорідними. Це завдання розвиває алгоритмічне мислення, оскільки вимагає переходу від дискретних моделей до неперервних.

– **Фізична постановка задачі:** Змоделювати поширення світла в атмосфері над розпеченим асфальтом або піском пустелі. Відомо, що нагріте повітря має меншу густину, а отже, менший показник заломлення. Світло проходить крізь шари повітря з різною оптичною густиною і викривляється.

– **Завдання з програмування (Інформатика):**

1. Застосувати метод дискретизації: уявити атмосферу як набір із N тонких горизонтальних шарів (наприклад, 20 шарів).

2. Задати градієнт показника заломлення: у верхньому шарі $n=1.0003$, а в кожному нижчому шарі n зменшується на малу величину `step`.

3. Реалізувати ітераційний алгоритм: вихідний кут заломлення з першого шару стає вхідним кутом падіння для другого шару.

4. Зберегти координати точок заломлення у масиви та побудувати графік.

– **Очікуваний результат:** Учні отримують графік, де промінь світла є не прямою лінією, а плавною дугою, яка загинається вгору. Це є математичною моделлю нижнього міражу.

Проект №3. «Технології майбутнього: принцип роботи оптоволокна»

Рівень складності: Середній (прикладна спрямованість).

Актуальність: Демонстрація практичного застосування явища повного внутрішнього відбивання в сучасних телекомунікаціях (Інтернет).

– **Фізична постановка задачі:** Змодельовати проходження світлового імпульсу всередині скляного волокна, оточеного оболонкою з меншим показником заломлення. Визначити умови, за яких світло не покидає межі волокна.

– **Завдання з програмування (Інформатика):**

1. Налаштувати графічне вікно: намалювати дві паралельні горизонтальні лінії, що імітують стінки оптоволокна ($y=1$ та $y=-1$).

2. Реалізувати алгоритм багаторазового відбивання: коли координата y досягає стінки, змінювати знак вертикальної складової швидкості (або кута нахилу).

3. Додати умову перевірки: якщо кут падіння на стінку менший за граничний, промінь «згасає» (виходить назовні), якщо більший – продовжує рух.

– **Очікуваний результат:** Симуляція зигзагоподібного руху променя всередині світловода. Учні можуть дослідити, як вигин кабелю впливає на втрату сигналу.

Критерії оцінювання проєктної діяльності

Для оцінювання результатів роботи учнів над проєктами пропонується використовувати наступну шкалу критеріїв:

1. **Фізична коректність моделі (40%):** Чи правильно застосовані формули? Чи відповідає поведінка моделі реальним фізичним законам? Чи враховано граничні умови?

2. **Програмна реалізація (30%):** Чи є код оптимальним? Чи використано цикли та функції замість дублювання коду? Чи є код читабельним (наявність коментарів)?

3. **Візуалізація та інтерфейс (20%):** Наскільки зрозумілим є графік? Чи зручно користуватися програмою? Чи підписані осі та легенда?

4. Презентація результатів (10%): Вміння учня пояснити принцип роботи своєї програми та фізичну суть отриманого результату.

Розвиток ключових компетентностей

Впровадження описаної методики дозволяє комплексно формувати ключові компетентності випускника:

– **Інформаційно-цифрова:** навички кодування, налагодження алгоритмів, робота з бібліотеками візуалізації даних.

– **Математична:** розуміння тригонометричних функцій, поняття похідної (у проєкті з міражем), робота з системами координат.

– **Природнича:** глибоке розуміння оптичних явищ, навички моделювання експерименту.

– **Уміння вчитися впродовж життя:** здатність самостійно шукати інформацію для вирішення нестандартних технічних задач (пошук документації до бібліотек Python).

– **Спілкування іноземними мовами:** оскільки синтаксис мови Python базується на англійській лексиці, а документація бібліотек є англomовною, учні розвивають навички технічної англійської мови та навчаються інтерпретувати повідомлення програмного середовища.

Таким чином, запропонована методика використання мови Python виходить за рамки простої ілюстрації матеріалу і стає дієвим інструментом реалізації дослідницького підходу в навчанні фізики та інформатики.

У третьому розділі роботи розроблено методичне забезпечення для впровадження створеної комп'ютерної моделі в освітній процес з фізики. За результатами проведеного дослідження можна зробити такі висновки:

1. Розроблено методику проведення демонстраційного експерименту. Запропоновано сценарій уроку вивчення нового матеріалу, де комп'ютерна модель використовується як інтерактивний засіб візуалізації. Показано, що динамічна зміна параметрів моделі (показників заломлення, кута падіння) дозволяє вчителю ефективно створювати проблемні ситуації та демонструвати

явища, які складно відтворити на реальному обладнанні (наприклад, граничний кут повного внутрішнього відбивання для алмазу).

2. Створено інструктивно-методичні матеріали для лабораторних робіт. Розроблено структуру та зміст інструкції для учнів, яка перетворює програмний код на віртуальну лабораторну установку. Запропонована методика виконання роботи передбачає перевірку закону Снеліуса та дослідження умов повного внутрішнього відбивання, що забезпечує реалізацію діяльнісного підходу у навчанні.

3. Обґрунтовано методику організації проєктної діяльності. Запропоновано тематику навчальних STEM-проєктів різного рівня складності (моделювання дисперсії, міражів, оптоволокна). Доведено, що залучення учнів до модифікації програмного коду сприяє формуванню комплексу ключових компетентностей: предметних (фізичних), інформаційно-цифрових, математичних, а також компетентності спілкування іноземними мовами (через роботу з англомовним синтаксисом Python).

Таким чином, розроблене методичне забезпечення є цілісним комплексом, який дозволяє вчителю фізики ефективно використовувати сучасні інструменти програмування для підвищення якості навчання та мотивації учнів.

ВИСНОВКИ

У магістерській роботі здійснено комплексне та багатоаспектне дослідження проблеми модернізації навчання фізики в закладах загальної середньої освіти шляхом інтеграції сучасних цифрових технологій, зокрема комп'ютерного моделювання, у навчальний процес.

Узагальнення результатів теоретичного аналізу науково-методичної, психолого-педагогічної та технічної літератури дозволило констатувати, що в умовах стрімкого розвитку інформаційного суспільства та переходу до компетентнісної парадигми освіти традиційні методи викладання фізики потребують суттєвого оновлення.

Встановлено, що моделювання виступає фундаментальним методом наукового пізнання, який у навчальному процесі трансформується в потужний дидактичний засіб, що дозволяє подолати розрив між абстрактним характером фізичних теорій та емпіричним сприйняттям явищ учнями.

Особливої актуальності це набуває у старшій школі, де вивчаються складні, часто недоступні для безпосереднього спостереження процеси, а психологічні особливості учнів цього віку характеризуються переходом до гіпотетико-дедуктивного мислення, що вимагає використання засобів, здатних візуалізувати невидимі зв'язки та закономірності.

Аналіз світового та вітчизняного досвіду засвідчив, що комп'ютерне моделювання є ефективним інструментом для візуалізації фізичної реальності, проте існуючі готові програмні продукти часто функціонують за принципом «чорної скриньки», не даючи учням розуміння внутрішньої логіки процесів, що актуалізує потребу в розробці відкритих моделей.

Вагомим науково-практичним здобутком роботи стало обґрунтування та реалізація підходу, що базується на використанні мови програмування Python як інструментарію для створення фізичних моделей.

Доведено, що залучення учнів до аналізу або написання програмного коду фізичних явищ сприяє глибокій міжпредметній інтеграції фізики, математики та інформатики, реалізуючи принципи STEM-освіти. Вибір мови Python

аргументовано її лідерством у науковому світі, доступністю для початківців та наявністю спеціалізованих бібліотек NumPy та Matplotlib, які дозволяють виконувати складні математичні розрахунки та будувати якісні графічні залежності без надмірних зусиль.

У практичній частині роботи спроектовано та програмно реалізовано комп'ютерну модель явищ геометричної оптики, яка базується на строгій фізико-математичній формалізації законів заломлення світла та умов повного внутрішнього відбивання. Розроблений алгоритм та програмний код забезпечують не лише статичну ілюстрацію, а й динамічну симуляцію ходу променів у різномірних оптичних середовищах, дозволяючи користувачеві варіювати вхідні параметри в реальному часі та миттєво отримувати візуальний і числовий результат, що є критично важливим для формування дослідницьких навичок.

Ключовим результатом дослідження стала розробка цілісної методичної системи використання створеного програмного забезпечення під час вивчення розділу «Оптика». Запропонована методика охоплює різні форми організації навчальної діяльності та рівні пізнавальної активності учнів. Для етапу засвоєння нових знань розроблено сценарії демонстраційних експериментів, де вчитель використовує модель як інтерактивну дошку для перевірки висунутих учнями гіпотез («що буде, якщо змінити показник заломлення?»), що перетворює пасивне слухання лекції на активний евристичний пошук. Для практичної складової курсу створено детальні інструктивно-методичні матеріали до лабораторних робіт, які передбачають проведення комп'ютерного експерименту з подальшою математичною обробкою результатів, що дозволяє проводити лабораторні заняття навіть за відсутності дороговартісного оптичного обладнання.

Окрему увагу приділено організації навчально-дослідницької та проєктної діяльності учнів, для якої запропоновано тематику проєктів, спрямованих на самостійне розширення функціоналу моделі або дослідження складніших оптичних систем.

Підсумовуючи, можна стверджувати, що впровадження розроблених засобів комп'ютерного моделювання на базі мови Python у навчальний процес з фізики сприяє не лише підвищенню якості знань та інтересу до предмета, але й формуванню ключових компетентностей: цифрової грамотності, алгоритмічного мислення, здатності до розв'язання комплексних проблем та проведення наукових досліджень. Запропонований підхід дозволяє трансформувати урок фізики з процесу передачі інформації у процес активного конструювання знань, що повністю відповідає вимогам сучасної освітньої парадигми та готує випускників до успішної діяльності у високотехнологічному світі.

Перспективи подальших досліджень вбачаються у розширенні створеної бібліотеки моделей на інші розділи фізики, такі як термодинаміка та електродинаміка, а також у розробці адаптивних алгоритмів для перевірки знань учнів на основі результатів моделювання.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Бондаренко О., Ластовецький В., Пилипчук О. Шестопалов Є. Інформатика. Підручник для 7 класу ЗЗСО. Електронний інтерактивний додаток. URL: <https://ua.izzi.digital/DOS/750894/750899.html>
2. Фізика. 11 клас. Всеукраїнська школа онлайн. Заломлення світла. Закони заломлення. URL: <https://lms.e-school.net.ua/asset-v1:UIED+Physics-11th-grade+2020+type@asset+block@1.pdf>
3. Величко С. Психологічні основи формування абстрактного мислення старшокласників. К.: Педагогічна думка, 2019.
4. Бар'яхтар В., Довгий С. Методика навчання фізики в старшій школі. К.: Освіта, 2020.
5. Duit R., Treagust D. Conceptual change: A powerful framework for improving science teaching and learning. *International Journal of Science Education*, 2003.
6. Зайцева О. Діагностика предметних компетентностей з фізики: методичні рекомендації. Запоріжжя: ЗНУ, 2021.
7. Hestenes D. Modeling Theory for Physics Instruction. *American Journal of Physics*, 1987.
8. Сосницька Н. Сучасні технології навчання фізики: моделювання, симуляції, компетентнісний підхід. Харків: Ранок, 2022.
9. Zacharia Z., Olympiou G. Physical versus virtual labs in physics education: A comparative study. *Journal of Science Education and Technology*, 2011.
10. Мартинюк О. Метакогнітивні стратегії у формуванні наукових понять у старшокласників. Львів: ЛНУ ім. Франка, 2020.
11. Гончаренко С. У. Український педагогічний словник. К.: Либідь, 1997. 376 с.
12. Мартинюк М. Т. Методика навчання фізики в основній школі: Навч. посіб. Умань: ВПЦ «Візаві», 2017. 254 с.
13. Атаманчук П. С. Дидактичні основи формування фізичних понять: [монографія]. Кам'янець-Подільський: К-ПДПУ, 2003. 220 с.

14. Савгира С. М. Використання комп'ютерних моделей у навчанні фізики. *Фізика та астрономія в школі*. 2008. № 2. С. 15–18.
15. Blicher H. L. PhET: Interactive Simulations for Teaching and Learning Physics. *The Physics Teacher*. 2012. Vol. 50. P. 50–51.
16. National Research Council. *A Framework for K-12 Science Education: Practices, Crosscutting Concepts, and Core Ideas*. Washington, DC: The National Academies Press, 2012.
17. Wieman C. E., Perkins K. K., Adams W. K. Oersted Medal Lecture 2007: Interactive simulations for teaching physics: What works, what doesn't, and why. *American Journal of Physics*, 2008, 76(4 & 5), pp. 393–399.
18. Department for Education (UK). *National curriculum in England: computing programmes of study*. London: DfE, 2013.
19. Lavonen J. Finland: Best Practices in Science Education. In: *Science Education in Europe*, London, 2018.
20. Ministry of Education (Singapore). *ICT in Education Masterplan 4*. Singapore: MOE, 2015.
21. Punie Y., Redecker C. European Framework for the Digital Competence of Educators: DigCompEdu. *Publications Office of the European Union*, Luxembourg, 2017.
22. Державний стандарт базової середньої освіти, затверджений постановою Кабінету Міністрів України від 30 вересня 2020 р. № 898.
23. Биков В. Ю., Спірін О. М., Пінчук О. П. Сучасні засоби ІКТ підтримки навчально-дослідницької діяльності учнів. *Інформаційні технології і засоби навчання*. 2018. Том 68. № 6.
24. Семеріков С. О., Теплицький І. О. Комп'ютерне моделювання у шкільному курсі фізики: минуле, сучасне, майбутнє. *Збірник наукових праць Кам'янець-Подільського національного університету*, 2019.
25. Морзе Н. В., Барна О. В. Формування навичок комп'ютерного моделювання як складової обчислювального мислення. *Комп'ютер у школі та сім'ї*. 2021. № 3.

ДОДАТКИ

Код моделі

```
import numpy as np
import matplotlib.pyplot as plt
from matplotlib.widgets import Slider, TextBox

# --- Налаштування фізики ---
def calculate_refraction(theta1_deg, n1, n2):
    ««««
    Розраховує кут заломлення за законом Снеліуса.
    Повертає кут у градусах або None, якщо виникає повне внутрішнє
    відбиття.
    ««««

    theta1_rad = np.radians(theta1_deg)

    # Закон Снеліуса:  $n1 \cdot \sin(t1) = n2 \cdot \sin(t2)$ 
    #  $\sin(t2) = (n1 / n2) \cdot \sin(t1)$ 
    sin_theta2 = (n1 / n2) * np.sin(theta1_rad)

    # Перевірка на повне внутрішнє відбиття (якщо  $\sin > 1$ , розв'язку немає)
    if abs(sin_theta2) > 1.0:
        return None

    theta2_rad = np.arcsin(sin_theta2)
    return np.degrees(theta2_rad)

# --- Налаштування графіки ---
fig, ax = plt.subplots(figsize=(10, 7))
plt.subplots_adjust(left=0.1, bottom=0.35) # Місце для слайдерів
ax.set_title(«Моделювання Заломлення Світла (Закон Снеліуса)»)
```

```

ax.set_xlim(-1.5, 1.5)
ax.set_ylim(-1.5, 1.5)
ax.set_aspect('equal')
ax.axis('off') # Ховаємо осі координат для краси

# Малюємо середовища
medium1_poly = ax.fill_between([-2, 2], 0, 2, color='#e0f7fa', alpha=0.3,
label='Середовище 1')
medium2_poly = ax.fill_between([-2, 2], -2, 0, color='#e1bee7', alpha=0.3,
label='Середовище 2')

# Лінія розділу та нормаль
ax.axhline(0, color='black', linewidth=2) # Межа
ax.axvline(0, color='gray', linestyle='--', alpha=0.7) # Нормаль

# Початкові промені (пустишки, які будемо оновлювати)
incident_ray, = ax.plot([], [], 'r-', linewidth=3, label='Падаючий промінь')
refracted_ray, = ax.plot([], [], 'b-', linewidth=3, label='Заломлений промінь')
reflected_ray, = ax.plot([], [], 'r--', linewidth=1.5, alpha=0.5, label='Відбитий
промінь (частково)')

# Текстові підписи
text_n1 = ax.text(-1.4, 0.5, "n1", fontsize=12, color='blue')
text_n2 = ax.text(-1.4, -0.5, "n2", fontsize=12, color='purple')
text_info = ax.text(0.5, 1.2, "n1 > n2", fontsize=10, bbox=dict(facecolor='white',
alpha=0.8))

# --- Слайдери ---
ax_theta = plt.axes([0.2, 0.2, 0.6, 0.03])
ax_n1 = plt.axes([0.2, 0.15, 0.6, 0.03])

```

```
ax_n2 = plt.axes([0.2, 0.1, 0.6, 0.03])
```

```
slider_theta = Slider(ax_theta, 'Кут падіння (°)', 0, 89.9, valinit=45)
```

```
slider_n1 = Slider(ax_n1, 'Показник n1 (верх)', 1.0, 3.0, valinit=1.0)
```

```
slider_n2 = Slider(ax_n2, 'Показник n2 (низ)', 1.0, 3.0, valinit=1.5)
```

```
# --- Функція оновлення ---
```

```
def update(val):
```

```
    theta1 = slider_theta.val
```

```
    n1 = slider_n1.val
```

```
    n2 = slider_n2.val
```

```
# 1. Малюємо падаючий промінь
```

```
# Він приходить з 2-го квадранта ( $x < 0$ ,  $y > 0$ ) в точку (0,0)
```

```
# Кут рахується від нормалі (ось Y).
```

```
# Тому  $x = -\sin(\theta)$ ,  $y = \cos(\theta)$ 
```

```
t1_rad = np.radians(theta1)
```

```
x_inc = -np.sin(t1_rad) * 1.5
```

```
y_inc = np.cos(t1_rad) * 1.5
```

```
incident_ray.set_data([x_inc, 0], [y_inc, 0])
```

```
# 2. Розрахунок заломлення
```

```
theta2 = calculate_refraction(theta1, n1, n2)
```

```
# 3. Малюємо заломлений промінь або відбиття
```

```
if theta2 is not None:
```

```
    # Світло проходить
```

```
    t2_rad = np.radians(theta2)
```

```
    x_refr = np.sin(t2_rad) * 1.5
```

```
    y_refr = -np.cos(t2_rad) * 1.5 # Йде вниз
```

```

refracted_ray.set_data([0, x_refr], [0, y_refr])
refracted_ray.set_color('blue')
refracted_ray.set_label('Заломлений промінь')

text_info.set_text(f'Кут падіння: {theta1:.1f}°\nКут заломлення: {theta2:.1f}°')

# Слабке відбиття завжди є (Френель), покажемо його пунктиром
reflected_ray.set_data([0, -x_inc], [0, y_inc]) # Симетрично відносно Y

else:
    # Повне внутрішнє відбиття (Total Internal Reflection)
    # Промінь заломлення зникає (або стає відбитим)
    refracted_ray.set_data([0, -x_inc], [0, y_inc]) # Стає повністю відбитим
    refracted_ray.set_color('red') # Змінюємо колір, щоб показати відбиття

    reflected_ray.set_data([], []) # Ховаємо допоміжний пунктир

    text_info.set_text(f'Кут падіння: {theta1:.1f}°\nПОВНЕ ВНУТРІШНЄ ВІДБИТТЯ!')

# Оновлення підписів середовищ
text_n1.set_text(f'n1 = {n1:.2f}»)
text_n2.set_text(f'n2 = {n2:.2f}»)

fig.canvas.draw_idle()

# Підключаємо функцію до слайдерів
slider_theta.on_changed(update)

```

```
slider_n1.on_changed(update)
```

```
slider_n2.on_changed(update)
```

```
# Запуск першого кадру
```

```
update(0)
```

```
plt.show()
```