

УДК 378.016:53-047.58

DOI <https://doi.org/10.32782/pet-2021-2-1>

Ніна ГОЛОВІНА

кандидат фізико-математичних наук, доцент кафедри експериментальної фізики, інформаційних та освітніх технологій, Волинський національний університет імені Лесі Українки, просп. Волі, 13, м. Луцьк, Україна, 43025

Микола ГОЛОВІН

кандидат фізико-математичних наук, доцент кафедри комп'ютерних наук та кібербезпеки, Волинський національний університет імені Лесі Українки, просп. Волі, 13, м. Луцьк, Україна, 43025

Бібліографічний опис статті: Головіна, Н., Головін, М. (2021) Методичні особливості моделювання фізичних явищ на прикладі взаємодіючих коливань. *Фізика та освітні технології*, 2, 3–10, doi: <https://doi.org/10.32782/pet-2021-2-1>

**МЕТОДИЧНІ ОСОБЛИВОСТІ МОДЕЛЮВАННЯ ФІЗИЧНИХ ЯВИЩ
НА ПРИКЛАДІ ВЗАЄМОДІЮЧИХ КОЛИВАНЬ**

Розглядаються методичні особливості навчального моделювання фізичних явищ на прикладі реалізації моделі коливання системи зв'язаних осциляторів та моделі частотної взаємодії двох осциляторів. Для реалізації моделей використовується мова програмування Python із застосуванням спеціалізованої бібліотеки Visual. Надається програмний код. Проведений модельний експеримент. Представлені траєкторії рухів відповідних частинок. Проведений аналіз результатів моделювання. Проаналізовані методичні особливості моделювання фізичних явищ в контексті лабораторних практикумів з фізики та програмування. Особливістю цієї роботи є те, що навчальне моделювання фізичних явищ тут розглядається в контексті концепції когнітивної (пізнавальної) психології. Поєднання доктрин когнітивної психології і методичних підходів навчання в галузі інформатики і фізики дало можливість розглядати процеси навчання через призму структури знань, її еволюції, ієрархічності організації та порційності модифікуючих цю структуру доз матеріалу. Окремі алгоритмічні задачі на моделювання фізичних процесів розглядаються в роботі, як компоненти цілісної системи задач, що входять в лабораторний практикум. У роботі відстоюється думка, що лабораторні практикуми з фізики і з програмування в окремих темах треба представляти, як систему зв'язаних між собою задач. Відмічається, що з точки зору методики навчання програмування, бажано щоб кожна поточна задача практикуму, з одного боку, була б результатом еволюції попередніх задач і їх розв'язків, а з іншого боку, кожна задача мала б бути базисом для наступної задачі. У роботі акцентується увага на тому, що існують ключові задачі, які є базисом для цілого пакету задач. У роботі зазначається, що аналогічна ситуація, насправді, з практикумами з фізики, однак у практикумах з програмування це особливо відчувається, як студентами, так і викладачами, що проводять лабораторні заняття. Відмічається, що щойно описана структура задач для практикумів в окремих темах породжує уявлення, як про дерево задач (проблем), які треба подолати в процесі лабораторного практикуму з окремої теми. В роботі відстоюється думка про подібність навчальних дій в галузі практичного програмування і в галузі експериментальної фізики. Адже, як перша, так і друга галузь діяльності вимагає вишуканого абстрактно-логічного та причинно-наслідкового мислення; строгої структури професійних знань. В практичній навчальній діяльності з програмування, як і в практичній діяльності з фізики проводяться чисельні експерименти – перевірки на коректність роботи програми в процесі її створення. Перевірка на коректність роботи програм, що моделюють фізичні явища, ще більше зближують специфіку діяльностей.

Ключові слова: моделювання в навчанні, комп'ютерне моделювання, програмування, Visual Python, VPython, модель коливальних рухів, варіація параметрів, модельний експеримент, масштабування.

Nina HOLOVINA

Candidate of Physical and Mathematical Sciences, Associate Professor of the Department of Experimental Physics, Information and Educational Technologies, Lesya Ukrainka Volyn National University, 13 Volya Ave., Lutsk, Ukraine, 43025

Mykola HOLOVIN

Candidate of Physical and Mathematical Sciences, Associate Professor of the Department of Computer Science and Cybersecurity, Lesya Ukrainka Volyn National University, 13 Volya Ave., Lutsk, Ukraine, 43025

To cite this article: Holovina, N. & Holovin, M. (2021) *Metodychni osoblyvosti modeliuвання fizychnykh yavlyshch na prykladi vzaiemodiiuchykh kolyvan [Methodological features of physical phenomena on the example of interacting oscillations]. Physics and educational technology, 2, 3–10, doi: <https://doi.org/10.32782/pet-2021-2-1>*

METHODOLOGICAL FEATURES OF PHYSICAL PHENOMENA ON THE EXAMPLE OF INTERACTING OSCILLATIONS

Methodological specifics of educational modeling of physical phenomena on the example of realization of the model of oscillations of the system of connected oscillators and the model of frequency interaction of two oscillators. The programming language Python and the specialized library Visual are used for the realization of models. The programming code is provided. The model experiment is held. The trajectories of movement of corresponding particles are presented. The analysis of the results of modeling is held. The methodological specifics of modeling of physical phenomena in the context of laboratory physical practicum of physics and programming is analyzed.

The peculiarity of this work is that educational modeling of physical phenomena here is considered in the context of conceptual cognitive psychology. Combination of the doctrines of cognitive psychology and methodological approaches to education in the domain of informatics and physics gave the possibility to consider the educational processes through the prism of the structural knowledge, its evolution, organizational hierarchy and portionality of the doses of materials that modify this structure. Individual algorithmic problems related to modeling of the physical processes are regarded in this research as the components of the holistic system that are the part of the laboratory practicum. The study argues that laboratory workshops in physics and programming in certain themes should be presented as a system of problems that are connected to each other.

It is noted that from the point of view of the methodology of educational programming, it is desired that each task of the practicum is from one side the result of evolution of previous problems and their solutions, while from the other side, it should be the basis for the following task.

The research focuses its attention on the key problems that serve as a basis for the whole set of problems. The study accentuates that analogical situation is with physical practicums, however, it is especially felt in the programming practicums by the students and educators that held the laboratory works. It is noted that the recently described structure of the problem for the practicums in the separate themes creates the understanding about the tree of problems, that should be tackled in the course of laboratory practicum in each theme. The research claims that educational activities in the domain of practical programming and in the experimental physics are similar since both require abstract logical and causal thinking, as well as strict structure of professional knowledge. In the practical educational programming activity and practical activity in physics, a lot of experiments are held. They check the correctness of work of the program in the process of its creation. The check on the correctness of the work of the program that model physical phenomena bring closer, even more, the closeness of these activities.

Key words: modeling in education, computer modeling, programming, Visual Python, VPython, model of oscillatory movements, variation of parameters, model experiment, scaling.

Вступ. Комп'ютерне моделювання фізичних процесів є цікавою галуззю навчального програмування. Створення комп'ютерної моделі фізичного явища і подальше його використання наближає процес навчання до наукового пошуку, сприяє розвитку дослідницьких умінь і навичок, накладає на програму, що моделює, суворі рамки реального процесу. Важливі компетенції в галузі програмування, математики та фізики вимагаються від суб'єкта навчання в процесі створення ним моделей фізичних процесів.

Навчальні дії в галузі практичного програмування і в галузі експериментальної фізики мають багато спільного. Як перша, так і друга галузь діяльності вимагає вишуканого абстрактно-логічного та причинно-наслідкового мислення; строгої структури професійних знань. В обох випадках відбувається щільне сплетіння

витончених ментальних і матеріалізованих дій.

Програми, як об'єкти ментальної пізнавальної й матеріалізованої діяльності, є складно організованими, добре структурованими. Як на локальному рівні, так і на рівні окремих укрупнених функціональних блоків, програми мають численні логічні внутрішні та зовнішні зв'язки. У процесі навчальних дій в напрямку моделювання фізичних процесів і явищ контрастно проявляються особливості ментальних процесів людини, способи мислення, формування знань. Цьому сприяє висока формалізація знань та одночасно висока структурність і зв'язність в усіх трьох згаданих вище напрямках (інформатика, математика, фізика).

Дослідження в галузі методики навчання **актуальними** робить висока контрастність структури предметів вивчення, а саме: модельних програм та відповідних фізичних процесів.

Саме висока контрастність, витончена структура та зв'язність компонентів об'єкту вивчення (математичної моделі фізичного об'єкту або явища) дає можливість максимально формалізувати сам процес навчання та оптимізувати його. У цьому сенсі навчальне програмування в галузі створення фізичних моделей може бути хорошим дослідницьким полігоном для відпрацювання нових методик навчання в тому числі і в інших більш гуманітарних галузях знань, де навчальний матеріал має низький контраст, гірше формалізований і є менш структурним. Еволюцію пізнавальних схем в галузі програмування було детально розглянуто в роботах [1, 2].

Метою цієї роботи є дослідження особливостей застосування моделювання фізичних явищ до вивчення фізики та програмування на прикладі коливання системи зв'язаних осциляторів та частотної взаємодії двох осциляторів.

Завданням цієї роботи є аналіз еволюції структури знань на прикладі задач моделювання коливання системи зв'язаних осциляторів та частотної взаємодії двох осциляторів. Цікавими, в навчальному сенсі, є не тільки створення моделей та їх відлагодження але і послідовне дослідження відповідних фізичних процесів в різних проявах за допомогою готових моделюючих програм. Це є особливо цінним у тих випадках, коли модель демонструє явища мікро або макро світу, які до того ж відбуваються дуже швидко або надповільно. Адже ці явища не можливо досліджувати безпосередньо, тим більше в умовах шкільного експерименту.

В якості інструменту моделювання в цій роботі використовується мова Visual Python. Ця мова широко використовується міжнародним співтовариством для моделювання фізичних явищ. Мова Visual Python або VPython є мовою програмування Python плюс 3D графічний модуль, який називається Visual. VPython дозволяє користувачам створювати 3d об'єкти, такі як сфери, конуси, прямокутні бруски в 3d просторі і рухати їх там. Процес моделювання відбувається у відповідному вікні. Простота візуалізації процесу дозволяє програмістам зосередитися на обчислювальному аспекті модельної програми. Простота VPython зробила цей інструмент дуже зручним для ілюстрації «простої» фізики, особливо в освітньому середовищі [3]. Адже моделювання фізичних

процесів шляхом програмування 3d – графіки є досить складним. Поява мови VPython дозволила подолати цю складність. Іншою зручною обставиною мови є те, що транслятор Python працює в режимі інтерпретатора, а синтаксис мови не є складним. Це дуже зручно для навчального програмування, адже код програми, що інтерпретується, може запуститись з помилками і виконається до першої помилки.

Модель коливання системи зв'язаних осциляторів. Розглянемо задачу про коливання двох маятників, пов'язаних між собою пружним зв'язком жорсткістю q . Це можуть бути фізичні, математичні маятники або пружинні маятники (рис. 1). Ця задача і її реалізація мовою Pascal представлена в роботі [4].

Для створення комп'ютерної моделі взаємодії зв'язаних осциляторів з екранною візуалізацією процесу необхідно під'єднати до транслятора Python бібліотеку Visual та описати візуальні компоненти, що будуть задіяні в процесі.

```
from vpython import *
scene.title="Зв'язані коливання";
scene.background=vector(1,1,1);
ball1=sphere(pos=vector(0,4,0), raduis=0.01,
color=color.yellow)
ball2 = sphere(pos=vector(0,0,0), raduis=0.01,
color=color.green)
for a in [ball1, ball2] : a.orbit=curve(color=a.
color, radius = 0.1)
```

Далі необхідно ввести початкові параметри експерименту.

```
m=1; r=0.2; k=100; q=8; dt=0.01; v1=0; v2=0;
x1=1; x2=0; t=0
```

Система із зв'язаних пружинних маятників має два ступені вільності і описується рівняннями:

$$m\ddot{\xi}_1 + r\dot{\xi}_1 + k\xi_1 + q(\xi_1 - \xi_2) = 0,$$

$$m\ddot{\xi}_2 + r\dot{\xi}_2 + k\xi_2 + q(\xi_2 - \xi_1) = 0.$$

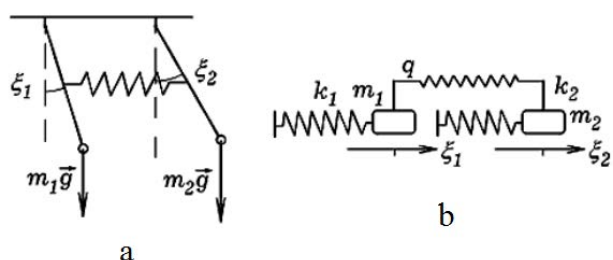


Рис. 1. Рух двох зв'язаних осциляторів:
a) Фізичні, математичні маятники
та b) пружинні маятники

Для їх розв'язання необхідно створити цикл за часом, в якому обчислюються прискорення, швидкості і координати маятників за формулами:

$$\theta_1^{t+1} = (-F^t - r\eta_1^t - k\xi_1^t) / m_1,$$

$$\eta_1^{t+1} = \eta_1^t + \theta_1^{t+1}\Delta\tau, \quad \xi_1^{t+1} = \xi_1^t + \eta_1^{t+1}\Delta\tau.$$

$$\theta_2^{t+1} = (F^t - r\eta_2^t - k\xi_2^t) / m_2,$$

$$\eta_2^{t+1} = \eta_2^t + \theta_2^{t+1}\Delta\tau, \quad \xi_2^{t+1} = \xi_2^t + \eta_2^{t+1}\Delta\tau,$$

де $F = q(\xi_1 - \xi_2)$.

Результати виводяться на екран.

Відповідний цикл за часом t виглядає наступним чином.

```
while t<20 :
rate(50); t=t+dt;
F=-q*(x1-x2);
a1=(-F-r*v1-k*x1)/m; v1=v1+a1*dt;
x1=x1+v1*dt;
a2=(F-r*v2-k*x2)/m; v2=v2+a2*dt;
x2=x2+v2*dt;
for a in [ball1]:
a.pos.x=-10+t
a.pos.y=3+x1*4
a.orbit.append(pos=a.pos)
for a in [ball2]:
a.pos.x= -10+t;
a.pos.y=-3+x2*4
a.orbit.append(pos=a.pos)
```

Модельний експеримент та результати дослідження. У математичних виразах параметри ζ_1 та ζ_2 представлені в програмі через значення y .

При запуску програми перший осцилятор (на рис. 2 зверху) починає здійснювати

коливання, в процесі яких його енергія поступово передається другому осцилятору (на рис. 2 знизу). Це відбувається на першому етапі роботи моделі (рис. 2a). З часом майже вся енергія концентрується на другому осциляторі (рис. 2b). Настає момент, коли коливання першого осцилятора практично припиняються, а другий осцилятор коливається з максимальною амплітудою. Після цього, через деякий час, енергія знову передається від другого маятника до першого і т. д. Видно, що енергія повертається в перший осцилятор. Такі взаємні транзити енергії відбуваються кілька разів (рис. 2c). Коливання затухають, тому їх амплітуда поступово падає і коливання припиняються.

Для вивчення поширення пружних хвиль в безперервних середовищах може бути використана модель одновимірного пружного середовища, що складається із сукупності осциляторів, розташованих уздовж прямої і пов'язаних між собою пружними зв'язками. Таке середовище, незважаючи на свою простоту, дозволяє досліджувати цілий ряд хвильових явищ: поширення збурення в пружному середовищі, відбиття і проходження імпульсу через границю розділу двох середовищ, інтерференції хвиль і т. д.

Модель частотної взаємодії двох осциляторів. У програмі, як і в попередньому випадку, нам знадобиться бібліотека Visual мови Python. Підключимо її. Крім того, започаткуємо роботу у вікні моделювання заголовком та задамо колір фону на якому буде відбуватись моделювання.

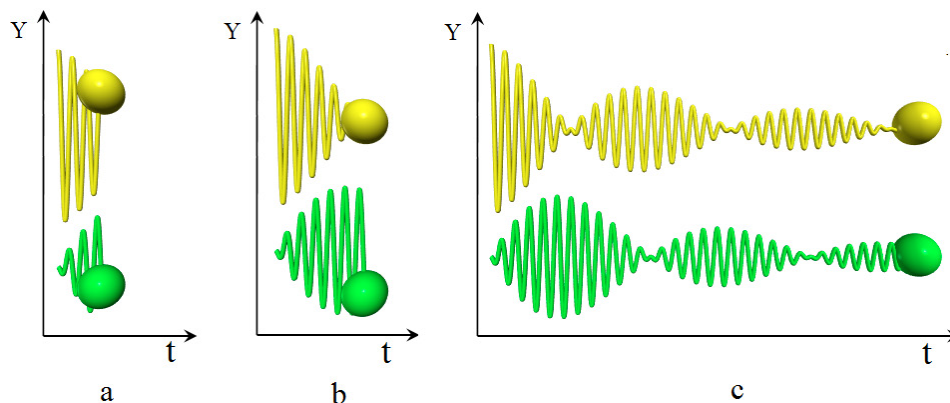


Рис. 2. Розвиток процесу взаємодії осциляторів в часовій розгортці. Перетікання енергії впродовж їх взаємодії та поступове затухання обох коливань


```

from vpython import *
scene.background=vector(1,1,1); scene.
title="Взаємодіючі коливання"
Забезпечимо малювання координатних осей
графіку:
L=10; curve(pos=[vector(-L,0,0),vector
(L,0,0)], color=vector(0.5,0.5,0.5))
xaxis=curve(pos=[vector(-L,0,0), vector
(L,0,0)], color=vector(0.5,0.5,0.5))
yaxis=curve(pos=[vector(0,0,0), vector(0,L,0)],
color=vector(0.5,0.5,0.5))
yaxis1=curve(pos=[vector(0,0,0),
vector(0,-L,0)], color=vector(0.5,0.5,0.5))

```

Екранна імітація модельних процесів вимагає описів графічних компонентів. Такий опис представлений в програмному фрагменті нижче.

```

ball = sphere(pos=vector(0,4,0), radius=0.01,
color=color.yellow)
for a in [ball1, ball2, ball3] : a.orbit =
curve(color=a.color, radius = 0.1)

```

У програмі відповідні вхідні дані задаються наступним чином.

$W=4$; $rW=0.7$; $A1=2$; $A2=2$; $t=0$, де W – це частота базового гармонічного коливання; $W+rW$ – частота гармонічного коливання, що додається до базового; а rW – різниця частот; $A1$ та $A2$ – амплітуди відповідних коливань; t – час в моделі.

Цикл `while t<20`: реалізує багатократне повторення обчислення поточних координат кульок, що виконують поперечні коливання розгорнуті в шкалі часу. Мова йде про два взаємодіючих коливання з частотами W та $W+rW$, а також третє складне коливання, яке є сумою перших двох. У цьому ж циклі відбувається і вивід на екран.

```

while t<20 :
rate(40); t += 0.1
for a in [ball1]:
a.pos.x=-10+t; a.pos.y=7+A1*cos(W*t);
a.orbit.append(pos=a.pos)
for a in [ball2]:
a.pos.x=-10+t; a.pos.y=2+A2*cos((W+rW)*t);
a.orbit.append(pos=a.pos)
for a in [ball3]:
a.pos.x=-10+t; a.pos.y=-4+A1*cos(W*t)+A2*cos((W+rW)*t);
a.orbit.append(pos=a.pos)

```

Змінна t в циклі `while t<20`: відповідає за час в моделі.

Модельний експеримент та результати дослідження. Був проведений модельний експеримент. Додавались коливальні рухи, які мали різні співвідношення частот. На рис.3 представлені коливання, що мають співвідношення частот як 4 до 4.7, а на рис.4 – як 4 до 5. На сумарному коливанні видно його високу чутливість, до частот коливань доданків.

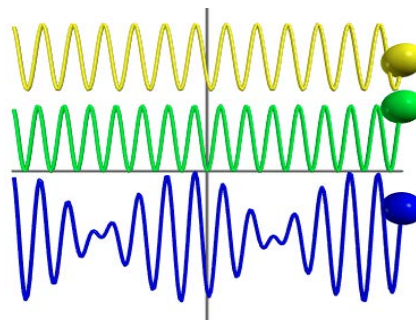


Рис. 3. Гармонічні коливання та їх сума.
Параметри коливань:
 $W=4$; $rW=0.7$; $A1=2$; $A2=2$

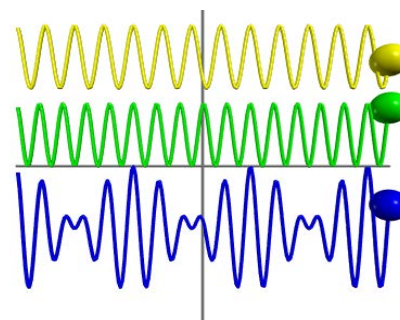


Рис. 4. Гармонічні коливання та їх сума.
Параметри коливань:
 $W=4$; $rW=1$; $A1=2$; $A2=2$

Фізика насичена моделями, що мають в своїй будові складні взаємозв'язані коливальні процеси. Це стосується, як класичної фізики так і її сучасних розділів, таких наприклад, як квантова фізика. Розгляд модельних експериментальних траєкторій може дозволити за їх виглядом розпізнати ту чи іншу конфігурацію фізичних параметрів, що визначають вигляд коливань. Повсякденне життя такого досвіду не дає. Якісне порівняння експериментальних і модельних траєкторій може розкрити механізми тих взаємодій, що детермінують результуючі експериментальні результати. Останнє дозволяє використовувати комп'ютерні моделі для пояснення сутності фізичних явищ.

Методика навчання фізики в контексті моделювання фізичних явищ. Практикум з фізики і з програмування в окремих темах можна представити, як системи зв'язаних між собою задач. З точки зору методики навчання програмування, бажано щоб кожна поточна задача практикуму, з одного боку, була б результатом еволюції попередніх задач і їх розв'язків, а з іншого боку, кожна задача мала б бути базисом для наступної задачі. Інколи існують ключові задачі, що є базисом для цілого пакету задач. Аналогічна ситуація, насправді, з практикумами із фізики, однак в практикумах з програмування це особливо відчувається, як студентами, так і викладачами, що проводять лабораторні заняття.

Зрозуміло, що щойно описана ситуація породжує уявлення про дерево задач (проблем), які треба подолати в процесі лабораторного практикуму з окремої теми. Це інтуїтивне відчуття співпадає з уявленням когнітивних психологів (пізнавальна психологія) стосовно еволюції структури знань. Там відстоюється думка, що ні одна пізнавальна схема (в тому числі і програмна) не є абсолютно новою. Кожна є трансформацією або модифікацією вже існуючої [5, 6]. Доза нової модифікуючої порції інформації, що поточно обмірковується в полі уваги не повинна перевищувати 7 ± 2 нових зв'язаних між собою компонентів [7]. У випадку програмування, це 7 ± 2 нових операторів, що зв'язані між собою в логічно завершену програмну конструкцію. У контексті статті, об'єднуюча логіка програмного фрагменту, що моделює фізичне явище, може мати не тільки алгоритмічний сенс, а і охоплювати сутність фізичного процесу чи його складової. Еволюція пізнавальних структур, що формується в процесі розв'язування алгоритмічних задач розглядалась в роботах [1, 2]. Пізнавальні структури в процесі їх еволюції модифікуються в напрямку їх перманентної диференціації [8]. Пізнавальна конструкція, що утворюється в межах однієї алгоритмічної задачі має ієрархічний характер [1, 2].

У контексті викладеного зрозуміло, що модель коливання системи зв'язаних осциляторів та модель частотної взаємодії двох осциляторів може бути впроваджена в практикум з моделювання фізичних явищ після наступного ланцюга задач: моделювання статичної частинки на екрані; модель прямолінійного рівномірного руху частинки вздовж однієї

з осей координат (X або Y); модель прямолінійного рівноприскореного руху тіла вздовж однієї з осей координат (наприклад, падіння тіла в полі тяжіння Землі); модуль прямолінійного рівносповільненого руху тіла під кутом до осі координат. Представлені в роботі задачі можуть бути базисом для розв'язку цілого пакету задач, зокрема, різноманітних силових взаємодій між тілами [9]. Базис, що слугував стартовою платформою для розглянутих в роботі моделей може бути також стартовою площадкою для таких моделей, як: модель прямолінійного руху тіла під кутом до перешкоди та його відбивання від цієї перешкоди або заломлення траєкторії тіла, якщо тіло це фотон на межі двох середовищ.

Важливим моментом в цій еволюції структури знань є порційність і дозованість добавок при трансформації та модифікації коду від однієї модельної програми до іншої в процесі її ускладнення. У цьому контексті необхідно пам'ятати про згаданий вище об'єм поля уваги 7 ± 2 .

Використання програмування в навчанні фізики завжди є складним завданням, як для учня, так і для викладача. Такий процес вивчення має справу з багатьма технічними та міждисциплінарними питаннями, а також вимагає синхронізації знань фізики, математики з інформатикою.

Комплексний підхід до викладання, коли інформатика та математика синхронізується з фізикою, демонструє методологію, яка ґрунтується на конструктивістському модельному підході до навчання. Це дає, з одного боку, можливість поглибити вивчення фізики, наповнити його математикою і інформатикою. З іншого боку, учні можуть провести серії експериментів з комп'ютерними моделями, які в умовах школи неможливо провести інакше.

При успішному поєднанні вивчення фізики, інформатики та математики модельний підхід допомагає глибше розуміти суть фізичних процесів. Цей підхід сприяє взаємопоглибленню в кожній із згаданих дисциплін, а також сприяє цілісному баченню фізики, математики та інформатики, як однієї тісно зв'язаної структури знань, в якій кожна з компонентів несе своє змістовне навантаження. З педагогічної точки зору, інтегрований модельний підхід може стимулювати учня відразу до вивчення декількох цивілізаційно важливих природничих дисциплін. Важливим аспектом модельного

підходу в навчанні є його потужний вплив на учня в сенсі формування причинно-наслідкового, абстрактно-логічного, матеріалістичного мислення [10].

Висновки. Створено програмну модель, що реалізує коливання двох маятників, пов'язаних між собою пружним зв'язком жорсткістю q , а також взаємодію двох коливальних рухів близьких за частотою. Використано мову Python, яка демонструє свою дієвість при моделюванні фізичних явищ та процесів.

Проведено модельний експеримент. Отримані модельні траєкторії рухів. Розглянуті ситуації з різними конфігураціями частот взаємодіючих осциляторів.

Зазначена важливість таких модельних експериментів в шкільному курсі в тому сенсі,

що повсякденне життя не дає досвіду взаємодії коливальних процесів. Однак існує велика кількість експериментальних даних, що дають досліди в мікро і макро світі. Якісне порівняння експериментальних і модельних траєкторій може розкрити механізм взаємодії. Останнє дозволяє використовувати комп'ютерні моделі для пояснення сутності фізичних явищ.

Відмічений потужний вплив модельного підходу на якість природничої ланки освіти. Зазначено, що цей підхід сприяє формуванню причинно-наслідкового, абстрактно-логічного, матеріалістичного мислення. Сприяє цілісному баченню фізики математики та інформатики, як однієї тісно зв'язаної структури знань, в якій кожна з компонентів несе своє змістовне навантаження.

ЛІТЕРАТУРА:

1. Зміст підготовки висококваліфікованого фахівця з інформаційних комп'ютерних технологій у контексті когнітивних процесів (на прикладі програмування). *Інформаційні технології в освіті*. 2008. Вип. 2. С. 66–73. URL: <http://ite.kspu.edu/issue-2/p-66-73>.
2. Головін М.Б., Головіна Н.А., Головіна Н.М. Модельний розгляд пізнавальних процесів, супутніх навчальному програмуванню. *Психологічні перспективи*. 2018. Вип. 31. С. 57–70. URL: http://nbuv.gov.ua/UJRN/Ppst_2018_31_7.
3. Гетманова Е.Е. Использование Visual Python для моделирования физических процессов. *Компьютерные инструменты в образовании*. 2005. № 4. С. 43–47.
4. Майер Р.В. Компьютерное моделирование. Учебно-методическое пособие для студентов педагогических вузов. Глазов : ГГПИ, 2015. 620 с.
5. Холодная М.А. Психология интеллекта: парадоксы исследования. Санкт-Петербург : Питер, 2002. 272 с.
6. Найссер У. Познание и реальность. Смысл и принципы когнитивной психологии. М. : Прогрес, 1981. 225 с.
7. Miller George A. The Magical Number Seven, Plus or Minus Two. *The Psychological Review*. 1956. Vol. 63. Issue 2. P. 81–97.
8. Чуприкова Н.И. Психология умственного развития: Принцип дифференциации. М. : Столетие, 1997. 478 с.
9. Головін М.Б., Головіна Н.А., Федонюк А.А. Аплікації з комп'ютерної фізики мовою Visual Python на прикладі моделювання силової взаємодії. *Комп'ютерно-інтегровані технології: освіта, наука, виробництво*. 2020. Вип. № 40. С. 16–22. URL: <http://cit-journal.com.ua/index.php/cit/article/view/151>.
10. Головін М.Б., Головіна Н.А. Специфіка навчальних дій, що містять комп'ютерне моделювання фізичних процесів. *Комп'ютерно-інтегровані технології: освіта, наука, виробництво*. 2018. Вип. 32. С. 10–18. URL: <http://ki.lutsk-ntu.com.ua/node/139/section/4>.

REFERENCES:

1. The meaning of training of the highly qualified specialists in the domain of information technologies in the context of cognitive processes (on the example of programming) (2008) *Information technologies in education*. 2, 66–73. URL: <http://ite.kspu.edu/issue-2/p-66-73> [in Ukrainian].
2. Holovin, M.B., Holovina, N.A., Holovina, N.M. (2018) Model considerations of cognitive processes associated with educational programming. *Psychological perspectives*. 31, 57–70. http://nbuv.gov.ua/UJRN/Ppst_2018_31_7 [in Ukrainian].
3. Hetmanova, E.E. (2005) Utilization of Visual Python for modeling physical processes. *Computer instruments in education*. 4, 43–47 [in Russian].
4. Mayer, R.V. (2015) *Computer modeling. Educational methodological textbook for the students of pedagogical educational establishments*. Glazov: GGPI [in Russian].
5. Holodnaya, M.A. (2002) *Psychology of intellect: paradoxes of research*. Saint Petersburg: Piter [in Russian].

6. Naysser, U. (1981) *Cognition and reality. The meaning and principles of cognitive psychology*. М.: Progress [in Russian].
7. Miller George, A. (1956) The Magical Number Seven, Plus or Minus Two. *The Psychological Review*. 63, 2, 81–97 [in English].
8. Chuprikova, N.I. (1997) *Psychology of Cognitive Development: Principle of Differentiation*. М.: Stoletie [in Russian].
9. Holovin, M.B., Holovina N.A., Fedoniuk A.A. (2020) Applications of computer physics with Visual Python on the example of modeling of power interaction. *Computer integrated technologies: education, science, production*. 40, 16–22. URL: <http://cit-journal.com.ua/index.php/cit/article/view/151> [in Ukrainian].
10. Holovin M.B. Holovina N.A. (2018) Specifics of educational actions that comprise of computer modeling of physical processes. *Computer integrated technologies: education, science, production*. 32, 10–18. URL: <http://ki.lutsk-ntu.com.ua/node/139/section/4> [in Ukrainian].