

УДК 53.072.8

DOI: 10.31499/2307-4906.3.2022.265913

ОСОБЛИВОСТІ ОРГАНІЗАЦІЇ ЛАБОРАТОРНИХ РОБІТ ПРИ ВИВЧЕННІ ФІЗИКИ СТУДЕНТАМИ ІТ-СПЕЦІАЛЬНОСТЕЙ

Діна Варганова, старший викладач кафедри інженерії програмного забезпечення, Державний університет «Житомирська політехніка».

ORCID: 0000-0002-8712-027X

E-mail: kipz_vdo@ztu.edu.ua

Роман Коломієць, кандидат технічних наук, доцент, доцент кафедри біомедичної інженерії та телекомунікацій, Державний університет «Житомирська політехніка».

ORCID: 0000-0002-9020-938X

E-mail: krt_kro@ztu.edu.ua, r.o.kolomiets@gmail.com

У статті показана можливість зручного використання мови програмування JavaScript при виконанні лабораторних робіт з фізики. Розглянуто два приклади: перший – з детермінованим рухом (прямолінійним рівномірним та прискореним), другий – з хаотичним рухом (броунівський рух). Описано структуру типового мініпроєкту (лабораторної роботи) та показано, що такий підхід в організації навчання глибоко пов'язаний зі специфікою ІТ-спеціальності. У результаті студенти набувають навички програмування моделей фізичних процесів із наочною візуалізацією отриманих результатів.

Ключові слова: освітні технології дистанційного навчання; моделювання процесів руху у фізиці; моделювання броунівського руху; розробка симуляторів фізичних процесів; JavaScript; WEB-технології у навчанні; лабораторні роботи з фізики; віртуальне навчальне середовище.

FEATURES OF THE ORGANIZATION OF LABORATORY WORKS IN THE STUDY OF PHYSICS BY STUDENTS OF IT SPECIALTIES

Dina Varganova, Senior Lecturer at the Department of Software Systems, Zhytomyr Polytechnic State University.

ORCID: 0000-0002-8712-027X

E-mail: kipz_vdo@ztu.edu.ua

Roman Kolomiets, Candidate of Technical Sciences (Ph.D.), Department of Biomedical Engineering and Telecommunication, Zhytomyr Polytechnic State University.

ORCID: 0000-0002-9020-938X

E-mail: krt_kro@ztu.edu.ua, r.o.kolomiets@gmail.com

In this article, we analyze the problems of organizing laboratory work in physics for students of IT specialties in terms of distance learning. We show the possibility of convenient use of WEB technologies and JavaScript programming language for this purpose. The choice of JavaScript is due to the true platform-independence of this programming language: it is possible to write code in any text editor and run it in any web browser. Unlike specialized environments (Scratch), JavaScript is one of the most common languages of true programming, but the simplicity of syntax is also quite accessible to high school and first-year students. We give two examples: the first with deterministic motion (rectilinear

uniform and accelerated) and the second with chaotic motion (Brownian motion). The article describes the structure of a typical mini-project (laboratory work) and shows that this approach in the organization of training is related to the specifics of the IT specialty. Now we consider only "static" cases – that is, without the use of animation. However, JavaScript tools also allow you to implement "dynamics", i.e. with animation. According to such a scheme, it is possible to implement almost any of the known physical processes. As a result, students acquire skills in programming models of physical processes with visual visualization of the results. Such models make it possible to change the initial conditions for the development of the physical process and see the process of its course, to compare the program-experimental results with the calculated, deeper and more complete understanding of the physical process during its programming. Thus, the foundation of knowledge of the future engineer – CAD designer (computer-aided design systems, which are widely used in real production and especially in its preparation).

Keywords: *educational technologies of distance learning; modeling of motion processes in physics; modeling of Brownian motion; development of simulators of physical processes; JavaScript; WEB technologies in education; laboratory works in physics; virtual learning environment.*

Розвиток інформаційного суспільства передбачає певні зміни у сфері освіти, її реформування. Проте поряд із подібними змінами як викладачам навчальних дисциплін, так і учням / студентам в останні роки доводиться стикатися із новими викликами (пандемія COVID-19, війна російської федерації проти України). Це призводить до потреби у пошуку нових ефективних методик викладання багатьох навчальних дисциплін. Особливо складно організувати змішане (дистанційне) навчання у випадку вивчення дисциплін природничого циклу (фізика, хімія, біологія, астрономія).

Традиційно вивчення таких дисциплін має дві складові: теоретичну і практичну. Причому «практична» складова не може обмежуватися лише розв'язуванням задач «на папері», оскільки навчальними планами цих дисциплін передбачені лабораторні заняття. В умовах дистанційного навчання сильно зменшується або навіть взагалі втрачається головна перевага такого виду занять – наочність. Лабораторне заняття дає можливість побачити предмет вивчення «наживо», «відчути його руками», провести реальний дослід та побачити реальний перебіг природного процесу.

Так, на сьогодні ми маємо значний прогрес у технологіях дистанційного навчання – за останні роки з'явилися різноманітні симулятори фізичних та хімічних процесів, віртуальні моделі біологічних об'єктів тощо. Але, на думку авторів, такому підходу дещо бракує «інтерактивності» – тобто якщо при виконанні «традиційної» лабораторної роботи учні (студенти) насамперед збирають експериментальну установку, то тут вони отримують «готовий піддослідний об'єкт». У першому випадку вони повинні проаналізувати хід збирання, та від того, наскільки вони ретельно проведуть дослід, залежить точність кінцевого результату. А ще слід додати те, що результат реального вимірювання – це випадкова величина із певними стохастичними характеристиками, і при реальних фізичних дослідах є потреба в оцінці похибки кінцевого результату – і порівняти з детермінованими результатами у випадку використання віртуальних моделей (симуляторів). Тому ми вважаємо, що симулятори – це добре, але якщо учні (студенти) будуть не просто користуватися подібними симуляторами, а і створювати їх – це буде ще краще.

Таким чином, основною проблемою, яка піднімається у цій роботі, є активізація залучення студентів до створення симуляторів фізичних процесів під час вивчення предмета.

Метою статті є демонстрація одного нестандартного підходу до організації лабораторних робіт з фізики при дистанційній формі навчання. Підхід полягає у тому, що студенти можуть написати програмний код, який симулює деякий фізичний процес, і поспостерігати за перебігом (поведінкою) віртуального фізичного процесу. При цьому є можливість змінити початкові дані (фізичних величин), і таким чином виконується практично справжня лабораторна робота з фізики. Отримані результати завжди можна перевірити шляхом теоретичних розрахунків.

Публікацій про особливості вивчення фізики в умовах дистанційного навчання вдалося знайти відносно небагато, і переважно вони присвячені особливостям шкільної освіти, а не професійно-технічної або вищої. Наприклад, у [1] наводиться ґрунтовний аналіз нормативно-правової бази для технологій дистанційного навчання та акцентується на STEM-підході до вивчення фізики (тобто вона повинна вивчатися у тісному зв'язку з іншими навчальними дисциплінами, зокрема хімією, астрономією, математикою, технологіями).

Непогано написані рекомендації щодо проведення окремих лабораторних робіт із фізики (зокрема, з вимірювання довжини хвилі світла, використання давачів смартфона, огляд симуляторів фізичних процесів) у [2], але це ресурс адресований насамперед учителям шкіл. Ми ж орієнтуємося переважно на вищу освіту.

У роботі [3] пропонується слухне рішення максимального використання особистих мобільних пристроїв учнів при постановці фізичних експериментів та удосконалено навчально-методичне забезпечення навчального фізичного експерименту, що передбачає використання мобільних додатків та дистанційних технологій.

Важливість проектного підходу до вивчення фізики розкривається в [4]. І тут неможливо не навести влучну цитату з цієї роботи: «...найціннішим є те, що в процесі теоретичної і практичної діяльності, в процесі експериментальної і самостійної роботи учень здобуває “власні знання”, які перестають бути абстрактними».

Також неможливо залишити поза увагою програмні симулятори фізичних процесів, зокрема міжнародний проект РНЕТ [5]. Цінність цього проекту полягає насамперед у тому, що він легкодоступний (не потрібно нічого інстальовати на комп'ютер) – все працює у браузері. Лабораторні роботи супроводжуються детальним описом (більшість інструкцій також перекладені і на українську мову). Станом на травень 2022 року в наявності 43 лабораторних роботи з фізики за темами «Рух», «Звук і хвилі», «Робота, енергія, сила», «Теплота», «Квантові явища», «Світло, випромінення», «Електрика, магнетизм, електричне коло». Зокрема, саме проект РНЕТ надихав авторів на власні розробки.

Головною ідеєю нашого підходу до проведення практичної (лабораторної) роботи з фізики є написання простого симулятора конкретного фізичного процесу. При цьому акцент робиться не на дизайні, а на математичній стороні – як, за якими математичними формулами відбувається фізичний процес. При цьому паралельно розв'язується ще одна задача – студент розбирається з алгоритмом розвитку процесу, може проекспериментувати із вхідними та проміжними даними і подивитися, як змінюється результат.

Звичайно, такий підхід виправданий насамперед при навчанні студентів саме на ІТ-спеціальностях, але він може бути використаний також для навчання студентів, які вже знають основи програмування. Ми пропонуємо використовувати мову

програмування JavaScript. Такий вибір обумовлений насамперед її доступністю та справжньою крос-платформеністю, оскільки вона виконується прямо в браузері, незалежно від операційної системи. Для написання коду можна використовувати будь-який текстовий редактор, але перевагу слід надати тому, який вміє підсвічувати синтаксис JavaScript (Notepad++, Sublime Text тощо). Хоча JavaScript – мова web-розробки, але її синтаксис настільки схожий на класичні C++/C#, що у студентів, які раніше вивчали ці мови, проблем з переходом не виникає. Єдина різниця, до якої треба звикнути: JavaScript – мова зі слабкою динамічною типізацією (C++ та C# – мови із сильною статичною типізацією). З одного боку, слабка динамічна типізація часто вважається недоліком мови, але з іншого – вона забезпечує більшу «гнучкість» коду. Хоча JavaScript за останні п'ять років стабільно тримається у п'ятірці найпопулярніших мов програмування, але з друкованою україномовною літературою все якось сумно. Є книга [6], яка орієнтована на дітей віком 9–14 років. Але, якщо не обмежуватися друкованою літературою, то можна знайти достатньо багато україномовних інтернет-ресурсів для вивчення цієї мови.

Проект подібних лабораторних робіт може виглядати у найпростішому випадку як html-файл, до якого динамічно лінкується js-файл з виконуваним кодом. Для візуалізації результатів зручно використовувати елемент HTML5 canvas, який, по суті, є растровим зображенням.

Наведемо приклад такого файлу:

```
<!DOCTYPE html>
<html>
  <head>
    <meta charset="utf-8">
    <title>Лабораторна робота ....</title>
  </head>
  <body>
    <canvas id="LAB" width="800" height="600">
      <script type="text/javascript" src="lab.js">
        </script>
      </canvas>
    </body>
  </html>
```

Тут для canvas обрана мнемонічна назва LAB і виконуваний js-файл названий lab.js, звичайно ці назви можна замінити на щось більш осмислене. Розміри canvas (зображення) становлять 800×600 пікселів. Їх теж можна змінити – залежно від розмірів екрану та зручності візуалізації.

Такий html-файл може бути просто відкритий у браузері. Важливо: js-файл повинен знаходитися у тій же папці, що і html-файл, бо інакше потрібно буде прописати точний шлях до нього в лапках після src=.

Далі розглянемо приклади наповнення файлу lab.js.

Приклад 1. Кінематика. Прямолінійний рух. Вивчення фізики часто починають саме з кінематики. Кінематика вивчає геометричні аспекти руху і не займається силами, які є причинами руху. Основною практичною задачею кінематики є

побудова траєкторії руху тіла (точніше – матеріальної точки, яка знаходиться у центрі мас тіла). У декартовій системі координат на площині $\{x, y\}$ у загальному випадку (тобто рух тіла з довільної точки $(x_0; y_0)$ з початковою швидкістю v_0 і прискоренням a , і під довільним кутом α) рівняння руху є параметричними рівняннями від параметра t , який має фізичний зміст часу:

$$x(t) = x_0 + v_{0x}t + \frac{a_x t^2}{2}; \quad y(t) = y_0 + v_{0y}t + \frac{a_y t^2}{2},$$

де $v_{0x} = v_0 \cos \alpha$, $v_{0y} = v_0 \sin \alpha$, $a_x = a \cos \alpha$, $a_y = a \sin \alpha$ – проєкції векторів швидкості та прискорення на відповідні осі координат. Наведемо JavaScript-код, який реалізує моделювання такого руху (напівжирним шрифтом виділено місце в коді, де задаються початкові дані для фізичного процесу):

```
// Налаштовування області для малювання:
let canvas = document.getElementById('LAB');
let ctx = canvas.getContext('2d');
// Встановлення границь області малювання:
const X = canvas.width;
const Y = canvas.height;
// Встановлення початку координат:
const X0 = Math.floor(X/2);
const Y0 = Math.floor(Y/2);
// Крок координатної сітки:
const grid = 100;
// Малювання координатної сітки:
drawGrid(grid);
// Встановлення значень змінних для моделювання траєкторії руху:
let x0 = -200, y0 = -200; // координати початкової точки
let angle = 30; // початковий кут у градусах
let v0 = 2; // початкова швидкість
let a = 5; // прискорення
// Малювання траєкторії руху:
drawTrajectory(x0, y0, angle, v0, a);

// Далі ідуть всі інші функції, які потрібні для роботи
// Малювання сітки системи координат – для наочності
function drawGrid(grid) {
  ctx.strokeStyle = "#00FF00";
  ctx.lineWidth = 1;
  let x, y;
  for (x=0; x<X; x+=grid) {
    ctx.beginPath();
    ctx.moveTo(x, 0);
    ctx.lineTo(x, Y);
    ctx.closePath();
    ctx.stroke();
  }
}
```

```
for (y=0; y<Y; y+=grid) {
  ctx.beginPath();
  ctx.moveTo(0, y);
  ctx.lineTo(X, y);
  ctx.closePath();
  ctx.stroke();
}
ctx.strokeStyle = "#000000";
ctx.beginPath();
ctx.moveTo(X0, 0);
ctx.lineTo(X0, Y);
ctx.closePath();
ctx.stroke();
ctx.beginPath();
ctx.moveTo(0, Y0);
ctx.lineTo(X, Y0);
ctx.closePath();
ctx.stroke();
}
// Малювання траєкторії руху
function drawTrajectory(x0, y0, angle, v0, a) {
  // Перевірка того, чи рух взагалі може відбутися:
  if ((v0==0)&&(a==0)) {
    alert("Швидкість та прискорення дорівнюють нулю – руху не буде!");
    return;
  }
  // Якщо рух можливий - то будуюмо його траєкторію:
  let t = 0;
  let x = X0+x0, y = Y0-y0;
  // Переводимо кут в радіани:
  let alpha = angle * Math.PI / 180;
  // Малюємо траєкторію руху:
  while ((x>0)&&(x<X)&&(y>0)&&(y<Y)) {
    x = Math.floor(X0+x0+v0*Math.cos(alpha)*t
      +(a*Math.cos(alpha)*t*t)/2);
    y = Math.floor(Y0-y0-v0*Math.sin(alpha)*t-(a*Math.sin(alpha)*t*t)/2);
    ctx.beginPath();
    ctx.arc(x, y, 3, 0, 2*Math.PI);
    ctx.closePath();
    ctx.fill();
    ctx.stroke();
    t++;
  }
}
```

На рисунку 1 наведені приклади моделювання прямолінійного руху, починаючи з точки з координатами (-200; -200). Чорними крапками (під кутом 60°) показаний приклад рівномірного руху (прискорення $a = 0 \text{ м/с}^2$), а сірими (під кутом 30°) – приклад рівноприскореного руху.

Особливо цікаво розглянути випадок, коли швидкість та прискорення мають різні знаки (наприклад, швидкість додатна, а прискорення від'ємне, або навпаки). Тоді рух виходить рівноуповільненим. Також на цій моделі студентам можна наочно показати той факт, що рух буде прямолінійним лише тоді, коли швидкість і прискорення є колінеарними векторами, а якщо між вектором швидкості та прискорення буде якийсь кут (не 0° або не 180°) – тоді рух прямолінійним не буде.

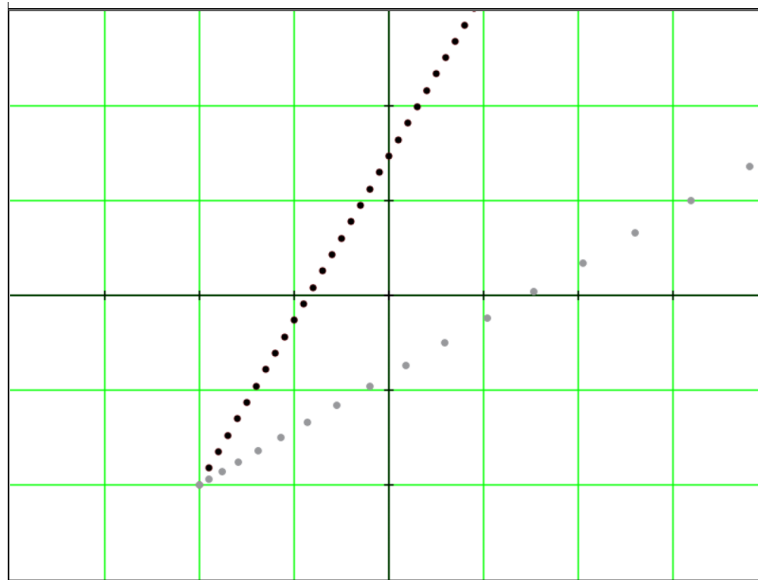


Рис. 1. Приклади моделювання рівномірного та рівноприскореного рухів

Приклад 2. Броунівський рух. Броунівський рух – невпорядкований, хаотичний рух частинки під дією нерівномірних ударів молекул речовини з різних боків у розчинах. Названий на честь ботаніка Роберта Броуна, який спостерігав це явище під мікроскопом у 1827 р. Повністю теорію броунівського руху та дифузії сформулював у 1905 р. Альберт Ейнштейн. Відкриття й пояснення броунівського руху мало велике значення для фізики, оскільки було свідченням теплового руху молекул.

У математиці броунівський рух розглядається як один із прикладів вінерівських процесів. Вінерівський процес в теорії випадкових процесів – це стохастичний процес з неперервним часом, що математично виражає випадкові блукання. Названий на честь Норберта Вінера – американського математика, творця основ кібернетики, які пов'язані із теорією інформації та теорією керування. Випадковий процес називається вінерівським, якщо:

1. Починається в нулі.
2. Має незалежні прирости, тобто величина кожного наступного зміщення (стрибка) ніяк не залежить від попереднього.
3. Усі прирости є стаціонарними, тобто мають однаковий закон розподілу (як правило – нормальний).

Вінерівський процес часто зустрічається в чистій, прикладній та фінансовій математиці, економіці, фізиці.

Наведемо JavaScript-код для моделювання траєкторії броунівського руху:

```
// Налаштовування області для малювання:
let canvas = document.getElementById('BrownMove');
let ctx = canvas.getContext('2d');
// Встановлення границь області малювання:
const X = canvas.width;
const Y = canvas.height;
// Встановлення початку координат:
const x0 = Math.floor(X/2);
const y0 = Math.floor(Y/2);
// Радіус броунівської частинки:
const R = 15;
// Діапазон можливих зміщень броунівської частинки:
const minStep = -20;
const maxStep = 20;
// Встановлення кольору броунівської частинки:
ctx.fillStyle = "#00FF00";
// Встановлення початкових значень змінних для моделювання:
let dx = 0, dy = 0;
let x = x0, y = y0;
let i = 0;
do {
  ctx.beginPath();
  ctx.arc(x, y, R, 0, 2*Math.PI);
  ctx.closePath();
  ctx.fill();
  ctx.stroke();
  dx = getIntRandomQNormal(minStep, maxStep);
  dy = getIntRandomQNormal(minStep, maxStep);
  x += dx;
  y += dy;
  i++;
} while (i<500); // Кількість кроків для візуалізації траєкторії
// Функція для реалізації рівномірного закону розподілу в діапазоні
function getIntRandomInRange(min, max) {
  return Math.floor(Math.random() * (max - min + 1) + min);
}
// Функція для реалізації квазінормального закону розподілу
function getIntRandomQNormal(min, max) {
  return Math.floor((getIntRandomInRange(min, max) +
    getIntRandomInRange(min, max) +
    getIntRandomInRange(min, max) +
    getIntRandomInRange(min, max))/4); }

```

На рисунку 2 наведені приклади хаотичних траєкторій броунівського руху для 1000 та 500 кроків (зміщень). Важливо відзначити, що вбудована функція генерації випадкового числа майже у всіх мовах програмування (і JavaScript тут не виняток) працює за квазірівномірним законом розподілу, тоді як у броунівському русі характер

зміни положення частини має нормальний закон розподілу. Реалізація справжнього нормального закону розподілу випадкової величини програмними засобами – доволі нетривіальна задача, але відомо, що нормальний закон розподілу випадкової величини можливо реалізувати як суму рівномірних [7]. Саме такий підхід і використовується в цьому коді (функція `getIntRandomQNormal`).

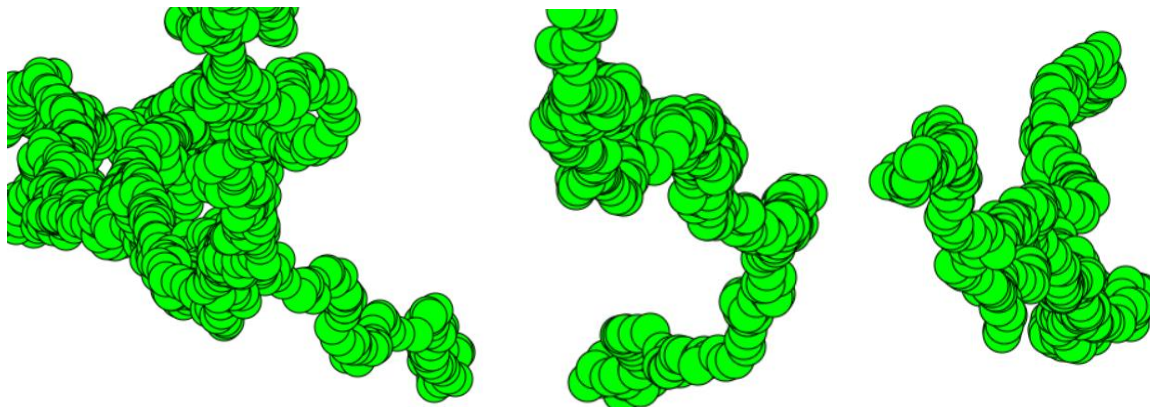


Рис. 2. Приклади моделювання траєкторій броунівського руху

Реалізація розглянутого підходу відкриває надзвичайно широкі можливості моделювання різноманітних фізичних процесів доступними засобами в умовах дистанційного навчання. Подібний підхід до лабораторних робіт можна назвати мультипредметним – оскільки він допомагає поглибити знання не лише фізики, а й програмування та комп'ютерної графіки. Показано, що подібним способом можливо моделювати не лише детерміновані фізичні процеси, а й стохастичні.

Експеримент з подібними лабораторними роботами з фізики було проведено в Державному університеті «Житомирська політехніка» у другому семестрі 2021/2022 навчального року в групах студентів, які навчаються за спеціальністю 121 – «Інженерія програмного забезпечення». У результаті такого підходу до організації лабораторних робіт з дисципліни «Фізика» зростає зацікавленість студентів (порівняно з попереднім роком), і лабораторні роботи з фізики стали органічно пов'язаними зі специфікою спеціальності.

Перспективи подальшого розвитку такого підходу доволі широкі. У статті ми розглядали лише «статичні» випадки – тобто без використання анімації. Проте засоби JavaScript дають можливість також реалізувати і «динамічні», тобто з анімацією. За подібною схемою можливо реалізувати майже будь-який з відомих фізичних процесів. У результаті студенти набувають навички програмування моделей фізичних процесів із наочною візуалізацією отриманих результатів. Такі моделі дають можливість змінити початкові умови розвитку фізичного процесу та побачити процес його перебігу, порівняти програмно-експериментальні результати із розрахунковими, глибше та повніше зрозуміти фізичний процес під час його програмування. Таким чином закладається фундамент знань майбутнього інженера – проєктувальника САПР (системи автоматизованого проєктування, які дуже широко використовуються на реальному виробництві та особливо при його підготовці).

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Блащук А. П. Інструктивно-методичні рекомендації щодо викладання фізики та астрономії у 2020/2021 навчальному році. *На допомогу методисту*. 2020. №6. URL: http://metodcluster.vn.ua/page/na_dop_met/1_2020/kaf_ekolog/lab_fizuku/fizuka.pdf (дата звернення: 03.05.2022).
2. На допомогу вчителів: 5 практичних порад, як провести дослідну роботу в умовах дистанційного навчання. URL: <https://b-pro.com.ua/statti/poglyad-eksperta.-laboratorni-roboti-v-umovah-distancijnogo-navchannya> (дата звернення: 03.05.2022).
3. Колесникова О. А. Діяльнісний підхід до формування в учнів експериментаторських умінь засобами мобільних та дистанційних технологій в навчанні фізики: автореф. дис. ... канд. пед. наук: 13.00.02. Київ, 2021. 25 с. URL: https://npu.edu.ua/images/file/vidil_aspirant/avtoref/%D0%94_26.053.06/Kolesnukova.pdf (дата звернення: 03.05.2022).
4. Мацюк В. М. Використання методу проєктів в умовах дистанційного навчання фізики. *Сучасні інформаційні технології та інноваційні методики навчання: досвід, тенденції, перспективи*: матеріали Міжнар. конф. (Тернопіль, 30 квітня 2020). Тернопіль: ТНПУ ім. В. Гнатюка, 2020. С. 57–58. URL: http://dspace.tnpu.edu.ua/jspui/bitstream/123456789/15358/1/18_Matsiuk.pdf (дата звернення: 03.05.2022).
5. Інтерактивні симуляції для природничих наук і математики. URL: <https://phet.colorado.edu/uk/> (дата звернення: 03.05.2022).
6. Морган Н. JavaScript для дітей (пер. з англ. Н. Агаджанян). Львів: Видавництво Старого Лева, 2018. 408 с.
7. Огірко О. І., Галайко Н. В. Теорія ймовірностей та математична статистика: навч. посіб. Львів: ЛьвДУВС, 2017. 292 с.

REFERENCES

1. Blashchuk, A. P. (2020). Instruktivno-metodychni rekomendatsii shchodo vykladannia fizyky ta astronomii u 2020/2021 navchalnomu rotsi. Na dopomohu metodystu. Issue 6. URL: http://metodcluster.vn.ua/page/na_dop_met/1_2020/kaf_ekolog/lab_fizuku/fizuka.pdf [in Ukrainian].
2. Na dopomohu vchyteliv: 5 praktychnykh porad, yak provesty doslidnu robotu v umovakh dystantsiinoho navchannia. URL: [https://b-pro.com.ua/statti/poglyad-eksperta.-laboratorni-roboti-v-umovakh-distancijnogo-navchannya](https://b-pro.com.ua/statti/poglyad-eksperta.-laboratorni-roboti-v-umovah-distancijnogo-navchannya) [in Ukrainian].
3. Kolesnykova, O. A. (2021). Diialnisnyi pidkhid do formuvannia v uchniv eksperymentatorskykh umin zasobamy mobilnykh ta dystantsiinykh tekhnolohii v navchanni fizyky. *Extended abstract of candidate's thesis*. Kyiv, NPU im. M. P. Drahomanova. URL: https://npu.edu.ua/images/file/vidil_aspirant/avtoref/%D0%94_26.053.06/Kolesnukova.pdf [in Ukrainian].
4. Matsiuk, V. M. (2020). Vykorystannia metodu proektiv v umovakh dystantsiinoho navchannia fizyky. *Suchasni informatsiini tekhnolohii ta innovatsiini metodyky navchannia: dosvid, tendentsii, perspektyvy: proceedings of the International Conference*. Ternopil: Ternopil Volodymyr Hnatiuk National Pedagogical University, 57–58. URL: http://dspace.tnpu.edu.ua/jspui/bitstream/123456789/15358/1/18_Matsiuk.pdf [in Ukrainian].
5. Interaktyvni symuliatsii dlia pryrodnychykh nauk i matematyky. URL: <https://phet.colorado.edu/uk/>
6. Morgan, N. (2018). JavaScript for Kids: A Playful Introduction to Programming. Lviv: Vydavnytstvo Staroho Leva [in Ukrainian].
7. Ohirko, O. I., Halaiko, N. V. (2017). Teoriiia ymovirnostei ta matematychna statystyka. Lviv: LvDUVS [in Ukrainian].