

д. т. н., професор **Вірченко Г. А.**,
kpivir@gmail.com, ORCID: 0000-0001-9586-4538

к. т. н., доцент **Яблонський П. М.**,
ypn@ukr.net, ORCID: 0000-0002-1971-5140

асистент **Грубич М. В.**,
mariya.grubich@gmail.com, ORCID: 0009-0003-9056-3826

Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»

ДЕЯКІ ПИТАННЯ КОМП'ЮТЕРНОЇ ВІЗУАЛІЗАЦІЇ ФІЗИЧНИХ ЯВИЩ І ПРОЦЕСІВ ЗАСОБАМИ МОВИ ПРОГРАМУВАННЯ PYTHON

Нині в більшості сфер життєдіяльності людей набули стрімкого розповсюдження різноманітні комп'ютерні засоби. Їхнє продуктивне використання суттєвим чином залежить від технічного і програмного забезпечення. У тому числі це також стосується володіння відповідними мовами, серед яких досить популярною в наш час є Python. Останнє обумовлено її здатністю розв'язувати широке коло задач, працювати в середовищі декількох операційних систем, простою синтаксису, гнучкістю та ефективністю генерованого коду, можливістю застосування значного числа потужних спеціалізованих бібліотек, дружнім інтуїтивно зрозумілим інтерфейсом і т. д.

На додаток до перерахованих позитивних властивостей мови Python головна її перевага полягає у відсутності потреби придбання платних ліцензій для забезпечення функціонування вказаного програмного продукту. Тому він отримав велику популярність у багатьох галузях, що обумовило актуальність зазначеного питання також і для сфери освіти. На практиці це стосується, зокрема, впровадження в навчальний процес фізико-математичного факультету Національного технічного університету України «Київський політехнічний інститут імені Ігоря Сікорського». У даній публікації подано деякі приклади використання мови Python на першому році навчання здобувачів бакалаврського рівня вищої освіти під час викладання дисциплін з інформатики та програмування. Основне завдання цих компонентів становить забезпечення належної підготовки студентів до ефективного застосування отриманих знань, вмінь і навичок для виконання необхідних розрахунків, формування графічних матеріалів, проведення комп'ютерних експериментів тощо. Важливим для спеціальностей фізико-математичного профілю є наукове дослідження різноманітних природних явищ та технічних процесів. Тому при вивченні основ програмування особлива увага приділяється акцентованим моментам. У статті запропоновану методику проілюстровано на

прикладі падіння тіла певної маси під дією сил тяжіння та опору, коли остання пропорційна швидкості руху. Наведено відповідну математичну модель, розглянуто шляхи її аналізу з використанням графічних засобів комп'ютерної візуалізації мови програмування Python. Окреслено подальші перспективні напрямки застосування напрацьованого підходу, які стосуються побудови реалістичних динамічних, тобто змінюваних у часі, геометричних моделей.

Ключові слова: інформатика; комп'ютерна візуалізація; мова програмування Python; освіта; фізичні явища і процеси.

Постановка проблеми. Сучасне навчання в закладах вищої освіти вимагає широкого і ґрунтовного впровадження комп'ютерних інформаційних технологій для підвищення якості фахової підготовки студентів. Зазначене питання складно вирішити без освоювання здобувачами мов програмування. Серед останніх доволі популярною є Python завдяки своїм перевагам щодо можливості функціонування в різних операційних системах, простоті, гнучкості та ефективності отримуваних програм, наявності багатьох спеціалізованих бібліотек, дружньому інтерфейсу тощо. Тому належна підготовка студентів становить актуальне завдання.

Аналіз досліджень і публікацій. У літературних джерелах [1-7] подано інформацію стосовно мови Python. У підручнику [1] викладено відомості з апаратного і програмного забезпечення сучасних комп'ютерів, проілюстровано багатьма прикладами синтаксис мови Python. Видання [2] розглядає не тільки такі базові поняття, як змінні, списки, цикли, класи тощо, а й ілюструє проєктно-орієнтований підхід до програмування. У книзі [3] зазначається, що Python, завдяки своїй простоті та ефективності, нині є однією з популярних мов. Крім фундаментальних програмних конструкцій особливу увагу приділено також реалізації об'єктно-орієнтованого підходу. Навчальний посібник [4] містить матеріал із теоретичних засад і практичного використання мови Python, призначений для студентів, аспірантів та інших користувачів. У публікації [5] описано належні засоби програмування, орієнтовані на студентів вищих навчальних закладів, інженерів і науковців. Це також стосується праці [6], де крім відповідних основ та стандартної бібліотеки проаналізовано велике число сторонніх пакетів, що забезпечує успішне їхнє застосування під час виконання курсових, бакалаврських і магістерських робіт різних спеціальностей. Видання [7] знайомить із Python англійською мовою.

Цілі та завдання статті полягають у поданні запропонованої методики комп'ютерного моделювання фізичних явищ та процесів графічними засобами мови програмування Python при навчанні студентів фізико-математичного факультету КПІ ім. Ігоря Сікорського. Зазначений підхід забезпечує отримання базових навичок стосовно використання відповідних математичних моделей, проведення їхнього ефективного

аналізу за допомогою сучасних інформаційних технологій.

Основна частина. Розглянемо застосування деяких графічних можливостей мови програмування Python на прикладі поданого в публікації [8] падіння зі швидкістю v тіла T під дією сили тяжіння F та сили опору R . Належну математичну модель ілюструє рис. 1.

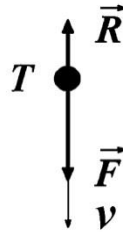


Рис. 1. Сили, що діють на падаюче зі швидкістю v тіло T :
 $\vec{F}$ – тяжіння; $\vec{R}$ – опору

Наведені сили мають протилежний напрям та величини

$$F = mg \geq |R|, \quad (1)$$

де $m > 0$ – маса тіла,

g – прискорення вільного падіння.

Проаналізуємо випадок, коли опір пропорційний швидкості v руху тіла, тобто

$$R = -k \cdot v, \quad (2)$$

де $k > 0$ – певний коефіцієнт.

Для нульової початкової швидкості тіла в момент часу $t=0$, згідно з виданням [8], маємо

$$v(t) = \frac{mg}{k} (1 - e^{-\frac{k}{m}t}). \quad (3)$$

З точки зору дослідження моделі, яка визначається виразами (1) ... (3), цікаве опрацювання певних проміжків для величин m , k та t . На підставі формули (3) гранична максимальна швидкість

$$v_{\max} = v(\infty) = \frac{mg}{k}. \quad (4)$$

Відповідно до співвідношення (4) вона зростає для більшої маси тіла і спадає при підвищенні діючого опору.

Відомо, що головна перевага геометричних моделей полягає в їхній наочності, яка суттєво спрощує практичне використання багатьох математичних описів, зокрема в різноманітній інженерній діяльності. Зручними та продуктивними в зазначеному плані є сучасні засоби комп'ютерної графіки. Викладемо декотрі з них, притаманні мові Python,

на прикладі наступних даних. Нехай необхідно дослідити поведінку тіл масою

$$m \in [m_n, m_k] = [5 \text{ кг}, 10 \text{ кг}], \quad (5)$$

силу опору з коефіцієнтом

$$k \in [k_n, k_k] = [5 \frac{\text{кг}}{\text{с}}, 10 \frac{\text{кг}}{\text{с}}] \quad (6)$$

та проміжок часу

$$t \in [t_n, t_k] = [0 \text{ с}, 10 \text{ с}]. \quad (7)$$

Зауважимо, що коефіцієнт опору зазвичай пропорційний площі поперечного перерізу тіла, яке рухається.

З виразів (5) і (6) випливає, що

$$\frac{m}{k} \in \left[\frac{m_n}{k_k}, \frac{m_k}{k_n} \right] = [0,5 \text{ с}; 2 \text{ с}], \quad (8)$$

але при дотриманні також умов (1) ... (3), (5) та (6).

Перші виконуються обов'язково, оскільки, див. формули (1) ... (3),

$$mg \geq kv,$$

$$mg \geq k \frac{mg}{k} (1 - e^{-\frac{k}{m}t}),$$

$$1 \geq (1 - e^{-\frac{k}{m}t}),$$

де остання нерівність правильна для довільних значень m , k , t , які розглядаються.

Залежності (5), (6), (8) свідчать, що певній величині m відповідає проміжок

$$\frac{m}{k} \in \left[\frac{m}{k_k}, \frac{m}{k_n} \right], \quad (9)$$

а деякому значенню k

$$\frac{m}{k} \in \left[\frac{m_n}{k}, \frac{m_k}{k} \right]. \quad (10)$$

Графічну комп'ютерну візуалізацію формул (5), (6), (8) ... (10) , реалізовано програмою на мові Python, приведеною на рис. 2.

```
# підключення бібліотек математичних і графічних функцій
import numpy as np
import matplotlib.pyplot as plt; from mpl_toolkits import mplot3d

# початкові та кінцеві значення (натуральні числа):
mp=5; mk=10 # маси m тіла, кг
kp=5; kk=10 # коефіцієнта k опору, кг/с

k=1
def f1(m,k): # рівняння поверхні
    return m/k

def f2(): # назви осей
    ax.set_xlabel("m, кг",fontsize=fnts)
    ax.set_ylabel("k, кг/с",fontsize=fnts)
    ax.set_zlabel("m/k, с",fontsize=fnts)

# проміжки на осях m та k
mval=np.linspace(mp,mk,mk-mp+1); kval=np.linspace(kp,kk,kk-kp+1)
m,k=np.meshgrid(mval,kval) # масив для величин m/k
fmk=f1(m,k) # присвоювання значень m/k

figs=5; fnts=12
# побудова поверхні
fig=plt.figure(figsize=(figs,figs)); ax=plt.axes(projection='3d'); f2()
ax.set_title('Візуалізація обмежень для m та k'); ax.elev,ax.azim,ax.roll=30,45,0
ax.plot_wireframe(m,k,fmk,color='red',rstride=1,cstride=1); plt.show()

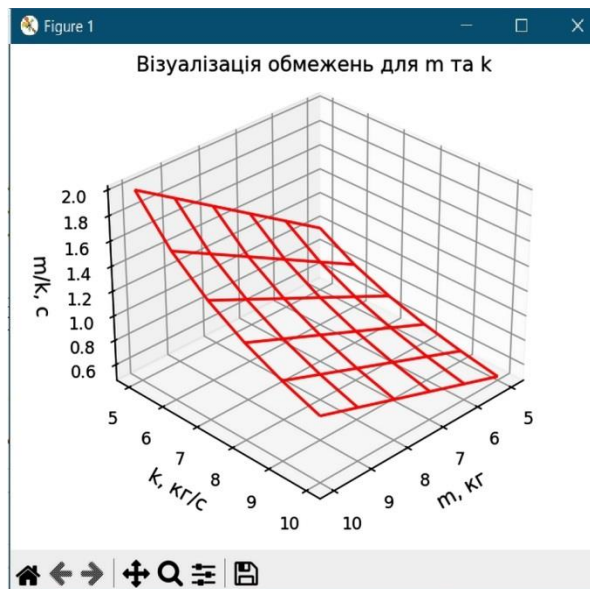
# вид 1
fig=plt.figure(figsize=(figs,figs)); ax=plt.axes(projection='3d'); f2()
ax.set_title('Залежність m/k від m'); ax.elev,ax.azim,ax.roll=0.001,-90.001,0
ax.plot_wireframe(m,k,fmk,color='red',rstride=1,cstride=1); plt.show()

# вид 2
fig=plt.figure(figsize=(figs,figs)); ax=plt.axes(projection='3d'); f2()
ax.set_title('Залежність m/k від k'); ax.elev,ax.azim,ax.roll=1.5,0,0.1
ax.plot_wireframe(m,k,fmk,color='red',rstride=1,cstride=1); plt.show()
```

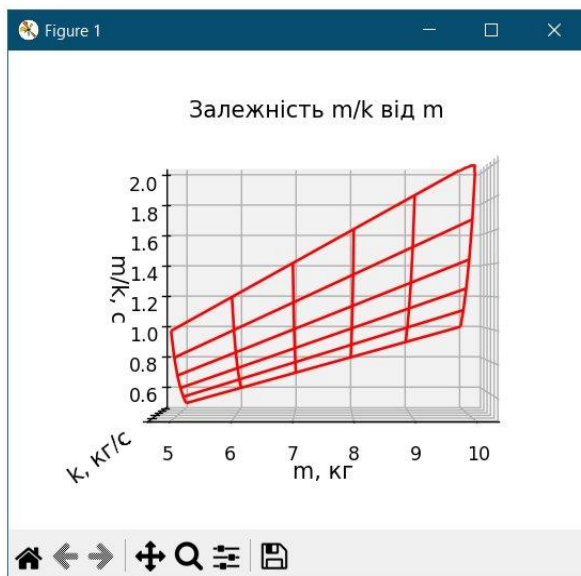
Рис. 2. Програма візуалізації опрацьовуваної області вихідних даних

Побудовані за допомогою неї динамічні зображення показано на рис. 3. При цьому наявні засоби для бажаного переміщення, обертання й масштабування наведених фігур, що забезпечує зручне виконання необхідного аналізу, сприяє формуванню цілісного системного уявлення про досліджуваний фізичний процес, спрощує його вивчення. Так, наприклад, рис. 3, а та рис. 3, б ілюструють характер змінювання проміжків m/k , які відповідають конкретним значенням m і k . Зокрема, величині $m=5$ кг корелює діапазон $m/k \in [\sim 0,5 \text{ с}, \sim 1 \text{ с}]$, $m=8$ кг – $m/k \in [\sim 0,8 \text{ с}, \sim 1,6 \text{ с}]$, $m=10$ кг – $m/k \in [\sim 1 \text{ с}, \sim 2 \text{ с}]$. Як бачимо, зі зростанням m належний проміжок m/k теж збільшується. Зауважимо, що суто графічні моделі, забезпечуючи високу наочність, мають певні обмеження стосовно своєї

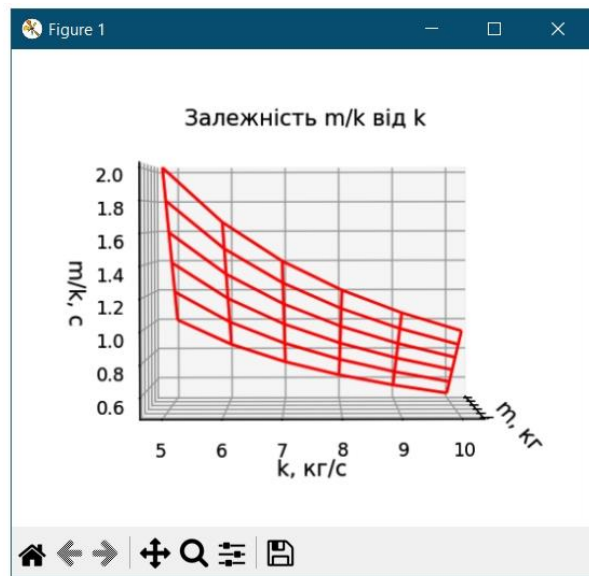
точності. Але цей недолік нівелюється застосуванням комп'ютерних графоаналітичних моделей, які розглядаються. За аналогією з рис. 3, б маємо для $k=5$ кг/с – $m/k \in [\sim 0,5$ с, ~ 1 с], $k=8$ кг/с – $m/k \in [\sim 0,6$ с, $\sim 1,2$ с], $k=10$ кг/с – $m/k \in [\sim 0,5$ с, ~ 1 с]. У даному разі зі зростанням k належний проміжок m/k зменшується.



a



б



в

Рис. 3. Приклади отриманих динамічних зображень:
a – загальний вигляд; *б, в* – спеціалізовані види

Отже, нами виконано аналіз вихідних проміжків (5) і (6) можливого змінювання маси m падаючого тіла та коефіцієнтів k сили опору його руху в аспекті їхнього співвідношення m/k , яке важливе для моделювання швидкості v , див. формули (3) та (4).

На рис. 4 приведено програму, написану на мові Python, що на

підставі множини

$$m/k = (0,5 \text{ с}; 0,8 \text{ с}; 1 \text{ с}; 1,5 \text{ с}; 2 \text{ с}) \quad (11)$$

будує графіки змінювання швидкості (3) для діапазону часу (7).

```
# підключення бібліотек математичних і графічних функцій
import numpy as np
import matplotlib.pyplot as plt

data=[0.5,0.8,1,1.5,2] # множина опрацьовуваних відношень m/k, с
tp=0; tk=10 # початкові та кінцеві значення часу t, с
g=9.81 # прискорення вільного падіння, м/с2

t=np.linspace(tp,tk,100) # проміжки на осі t
amk=np.array(data); n=len(amk)

def f(mk,t):
    return mk*g*(1-np.exp(-1/mk*t)) # формула швидкості v

fnts=12; fg=plt.figure(figsize=(7,7))
plt.title("Залежність швидкості v від m/k та часу t")
plt.xlabel("t, с",fontsize=fnts); plt.ylabel("v, м/с",fontsize=fnts)
plt.grid(True)

for i in range(n):
    s='m/k='+str(amk[i])+' с'; plt.plot(t,f(amk[i],t),label=s)
plt.legend(fontsize=fnts-1); plt.show()
```

Рис. 4. Програма створення графіків швидкості v падіння тіла

Результати її роботи показано на рис. 5. З них, зокрема, видно динаміку досягнення максимальної швидкості падіння тіла в залежності від визначених кортежем (11) співвідношень маси m та коефіцієнта k опору руху для діапазону часу (7).

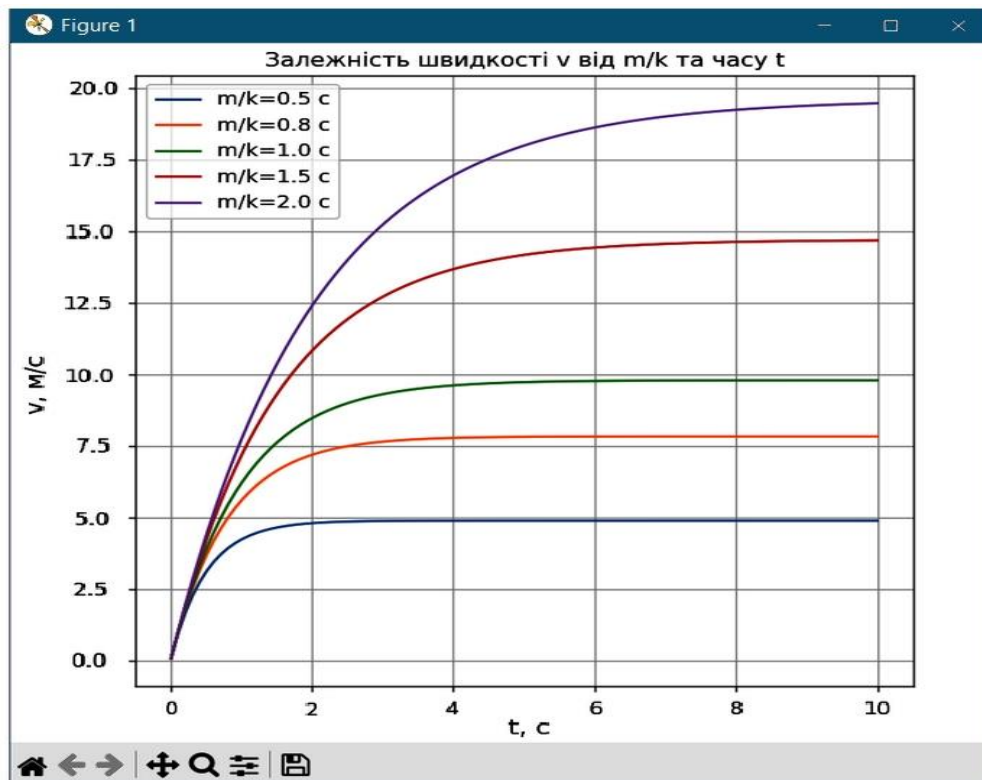


Рис. 5. Графіки змінювання швидкості v падіння тіла в залежності від співвідношення його маси m та коефіцієнта k опору руху

На завершення викладу матеріалу подамо програму на мові Python, яка буде поверхню (4) максимальних швидкостей $v_{\max}$ тіла для опрацьовуваних проміжків (5) і (6) змінювання його маси m та коефіцієнта k опору руху, див. рис. 6. Отримане динамічне зображення ілюструє рис. 7.

```
# підключення бібліотек математичних і графічних функцій
import numpy as np
import matplotlib.pyplot as plt; from mpl_toolkits import mplot3d

# початкові та кінцеві значення (натуральні числа):
mp=5; mk=10 # маси m тіла, кг
kp=5; kk=10 # коефіцієнта k опору, кг/с
g=9.81 # прискорення вільного падіння, м/с2

k=1
def f1(m,k): # рівняння поверхні vmax
    return m*g/k

def f2(): # назви осей
    ax.set_xlabel("m, кг",fontsize=fnts)
    ax.set_ylabel("k, кг/с",fontsize=fnts)
    ax.set_zlabel("vmax, м/с",fontsize=fnts)

# проміжки на осях m та k
mval=np.linspace(mp,mk,mk-mp+1); kval=np.linspace(kp,kk,kk-kp+1)
m,k=np.meshgrid(mval,kval) # масив для величин vmax
vmax=f1(m,k) # присвоювання значень vmax

figs=5; fnts=12
# побудова поверхні
fig=plt.figure(figsize=(figs,figs)); ax=plt.axes(projection='3d'); f2()
ax.set_title('Візуалізація поверхні vmax для m та k');
ax.elev,ax.azim,ax.roll=30,45,0
ax.plot_wireframe(m,k,vmax,color='red',rstride=1,cstride=1); plt.show()
```

Рис. 6. Програма формування поверхні максимальних швидкостей $v_{\max}$ падіння тіла

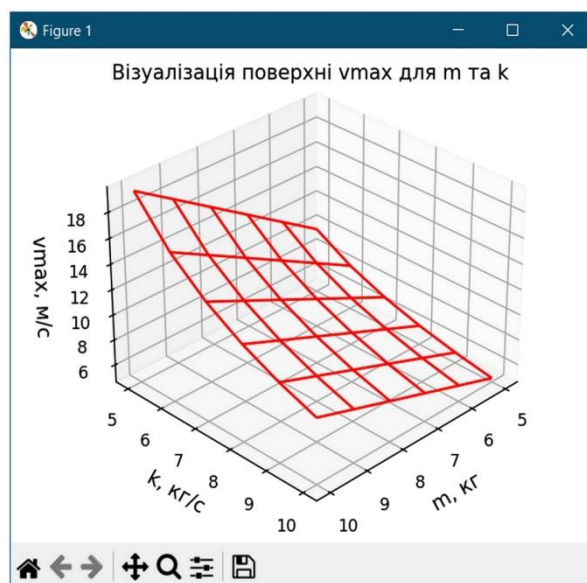


Рис. 7. Змінювання максимальної швидкості $v_{\max}$ падіння тіла в залежності від співвідношення його маси m та коефіцієнта k опору руху

Прогнозовано його геометрія, тобто величина $v_{\max}$, подібна з коефіцієнтом у вигляді прискорення g вільного падіння докладно проаналізованому вище характеру змінювання значень відношення m/k маси тіла до коефіцієнта сили опору.

Таким чином, нами розглянуто деякі питання щодо запропонованої методики комп'ютерної візуалізації фізичних явищ і процесів засобами мови програмування Python на прикладі опрацювання математичної моделі падіння тіла під дією сили тяжіння та опору, яка пропорційна швидкості руху. При цьому було комплексно поєднано такі складові, як створення та аналіз належних аналітичних залежностей, написання згідно з наявними потребами необхідної програми, графічна візуалізація її роботи, оцінка отриманих результатів.

Висновки та перспективи досліджень. Дану публікацію присвячено вирішенню важливого для якісної підготовки здобувачів вищої технічної освіти завдання стосовно ефективного впровадження в навчальний процес сучасних комп'ютерних інформаційних технологій. Напрацьований підхід подано на прикладі дослідження фізичних явищ засобами мови програмування Python, яка викладається студентам фізико-математичного факультету Національного технічного університету України «Київський політехнічний інститут імені Ігоря Сікорського». Один із перспективних напрямків подальшого вдосконалення викладеного матеріалу полягає в розроблянні програм, які створюють реалістичні динамічні зображення різноманітних процесів.

Література

1. Яковенко А. В. Основи програмування. Python. Частина 1: підручник. Київ: КПІ ім. Ігоря Сікорського, 2018. 195 с.
2. Маттес Е. Пришвидшений курс Python: практичний, проєктно-орієнтований вступ до програмування. Львів: Видавництво Старого Лева, 2021. 556 с.
3. Васильєв О. М. Програмування мовою Python. Тернопіль: Видавництво «Навчальна книга-Богдан», 2021. 503 с.
4. Висоцька В. А., Оборська О. В. Python: Алгоритмізація та програмування: навчальний посібник. Львів: Видавництво «Новий Світ-2000», 2021. 514 с.
5. Анісімов А. В., Дорошенко А. Ю., Погорілий С. Д., Дорогий Я. Ю. Програмування числових методів мовою Python: підручник. Київ: ВПЦ «Київський університет», 2015. 639 с.
6. Коней В. Б. Мова програмування Python для інженерів і науковців: навчальний. Івано-Франківськ: ІФНТУНГ, 2019. 267 с.

7. *Heisler F., Amos D., Bader D., Jablonski J.* Python basics. A practical introduction to Python 3. Real Python, 2020. 642 p.
8. *Тацій Р., Стасюк М., Пазен О.* Елементи математичного моделювання та прикладної математики: навчальний посібник. Львів: ЛДУ БЖД, 2021. 182 с.

References

1. *Yakovenko A. V.* Osnovy prohramuvannia. Python. Chastyna 1: pidruchnyk. Kyiv: KPI im. Ihoria Sikorskoho, 2018. 195 s. {in Ukrainian}
2. *Mattes E.* Pryshvydshenyi kurs Python: praktychnyi, proiektno-orientovanyi vstup do prohramuvannia. Lviv: Vydavnytstvo Staroho Leva, 2021. 556 s. {in Ukrainian}
3. *Vasyliiev O. M.* Prohramuvannia movoiu Python. Ternopil: Vydavnytstvo «Navchalna knyha-Bohdan», 2021. 503 s. {in Ukrainian}
4. *Vysotska V. A., Oborska O. V.* Python: Alhorytmizatsiia ta prohramuvannia: navchalnyi posibnyk. Lviv: Vydavnytstvo «Novyi Svit-2000», 2021. 514 s. {in Ukrainian}
5. *Anisimov A. V., Doroshenko A. Yu., Pohorilyi S. D., Dorohyi Ya. Yu.* Prohramuvannia chyslovykh metodiv movoiu Python: pidruchnyk. Kyiv: VPTs «Kyivskyi universytet», 2015. 639 s. {in Ukrainian}
6. *Kopei V. B.* Mova prohramuvannia Python dlia inzheneriv i naukovtsiv: navchalnyi. Ivano-Frankivsk: IFNTUNH, 2019. 267 s. {in Ukrainian}
7. *Heisler F., Amos D., Bader D., Jablonski J.* Python basics. A practical introduction to Python 3. Real Python, 2020. 642 p. {in English}
8. *Tatsii R., Stasiuk M., Pazen O.* Elementy matematychnoho modeliuvannia ta prykladnoi matematyky: navchalnyi posibnyk. Lviv: LDU BZhD, 2021. 182 s. {in Ukrainian}

Doctor of Technical Sciences, Professor **Gennadii Virchenko**,
kpivir@gmail.com, ORCID: 0000-0001-9586-4538

Doctor of Philosophy, Associate Professor **Petro Yablonskyi**,
ypn@ukr.net, ORCID: 0000-0002-1971-5140

Assistant **Mariya Hrubych**,
mariya.grubich@gmail.com, ORCID: 0009-0003-9056-3826

National Technical University of Ukraine
«Igor Sikorsky Kyiv Polytechnic Institute»

SOME ISSUES OF COMPUTER VISUALIZATION OF PHYSICAL PHENOMENA AND PROCESSES BY MEANS OF THE PYTHON PROGRAMMING LANGUAGE

Nowadays, various computer tools have become rapidly widespread in most areas of human activity. Their productive use depends significantly on the hardware and software. This also includes the knowledge of the relevant languages, among which Python is quite popular these days. The latter is due to its ability to solve a wide range of problems, work in the environment of several operating systems, simplicity of syntax, flexibility and efficiency of the generated code, the ability to use a significant number of powerful specialized libraries, a friendly intuitive interface, etc.

In addition to the listed positive properties of the Python language, its main advantage is the lack of need to purchase paid licenses to ensure the functioning of this software product. Therefore, it has gained great popularity in many industries, which has made this issue relevant also for the field of education. In practice, this applies, in particular, to the introduction into the educational process of the Faculty of Physics and Mathematics of the National Technical University of Ukraine «Igor Sikorsky Kyiv Polytechnic Institute». This publication presents some examples of using Python in the first year of undergraduate studies when teaching computer science and programming. The main purpose of these components is to ensure proper preparation of students for the effective application of the acquired knowledge, skills and abilities to perform the necessary calculations, generate graphic materials, conduct computer experiments, etc. Scientific research of various natural phenomena and technical processes is important for specialties of the physical and mathematical profile. Therefore, when studying the basics of programming, special attention is paid to the emphasized points. In this article, the proposed methodology is illustrated using the example of the fall of a certain mass body under the action of gravity and resistance forces, when the latter is proportional to the speed of movement. The corresponding mathematical model is presented, and ways of its analysis using graphical tools of computer visualization of the Python programming language are considered. Further promising directions for the application of the developed approach are outlined, which concern the construction of realistic dynamic, i.e. time-varying, geometric models.

Key words: informatics; computer visualization; Python programming language; education; physical phenomena and processes.