

Л. Міцкан

ORCID ID 0000-0002-7896-1739

Т. Вербицька

ORCID ID 0000-0002-0405-2735

В. М. Базурін

ORCID ID 0000-0002-6614-4889

Глухівський національний педагогічний університет  
імені Олександра Довженка

## ПОРІВНЯЛЬНИЙ АНАЛІЗ МОВ PYTHON I FREE PASCAL ЯК ПЕРШИХ МОВ ПРОГРАМУВАННЯ ДЛЯ УЧНІВ 8 КЛАСУ

У статті розкриваються проблеми вивчення Python у якості першої мови програмування. На основі емпіричних даних зроблено висновок про те, що основними проблемами є відсутність візуальних середовищ програмування для Python, специфічний синтаксис цієї мови і одночасне вивчення Python і Free Pascal. Автори переконливо доводять, що дані проблеми частково вирішуються шляхом добору середовища програмування та розробки відповідного методичного забезпечення. Синтаксис мови Python, незважаючи на свою специфіку, лаконічний і краще зрозумілий учням. Вивчення мови програмування Python доцільно здійснювати окремо від вивчення Free Pascal.

**Ключові слова:** програмування, мова програмування, Python, Free Pascal, Lazarus, інформатика.

**Постановка проблеми.** Державним стандартом середньої освіти з інформатики одним із основних завдань шкільного курсу інформатики визначено формування в учнів здатності будувати і використовувати інформаційні моделі, а також засоби опису та моделювання явищ та процесів.

Державним стандартом встановлено такі вимоги до загальноосвітньої підготовки учнів:

– знати поняття алгоритму, етапи процесу описання алгоритмів, у тому числі за допомогою програмних засобів, способи описання алгоритмів, призначення базових алгоритмічних структур, програмного коду,

– розуміти поняття формалізації задачі, поняття комп'ютерного моделювання, процес перетворення даних за формальними правилами, призначення комп'ютерної програми і пояснювати етапи її виконання на комп'ютері, уміти визначати властивості об'єктів та значення таких властивостей, здійснювати класифікацію і розпізнавати об'єкти, виконувати формалізоване описання об'єкта, здійснювати виокремлення підзадач в описанні задачі, уміти складати алгоритми для різних виконавців [1].

Отже, основи алгоритмізації і програмування визнано необхідними складовими шкільного курсу інформатики. Саме тому після тривалого ухилу у бік інформаційних технологій увага науковців знову звернулася до навчання основ програмування.

У якості першого середовища програмування виступає Scratch. Першими мовами програмування є Python і Free Pascal (середовище програмування Lazarus).

Мова Python останнім часом набуває все більшої популярності, як серед програмістів, так і серед учнів. Саме цим фактом зумовлене введення елементів програмування на мові Python у шкільний курс інформатики.

До переваг Python у якості першої мови програмування слід віднести досить простий синтаксис. Текст програм, написаних на мові Python, відрізняється від програм, написаних на подібних мовах (C++, C#, Java та ін.) лаконічністю, зрозумілістю. Проте вивчення Python у якості першої мови програмування викликає певні труднощі. Це пов'язано з низкою як об'єктивних, так і суб'єктивних чинників:

– незначною кількістю україномовного навчально-методичного забезпечення з

проблеми навчання програмування на мові Python;

- відсутністю широко вживаних візуальних середовищ програмування на мові Python;
- неможливістю створення екранних форм у візуальному режимі;
- деяка своєрідність синтаксису мови Python;
- одночасним вивченням мов Python і Free Pascal (Lazarus).

Практика показує, що частина наведених чинників поступово нівелюється (наприклад, навчально-методичне забезпечення з навчання програмування на мові Python інтенсивно розробляється). Для усунення негативного впливу інших чинників необхідна цілеспрямована робота викладачів і вчителів.

**Аналіз актуальних досліджень.** Проблема навчання учнів основ програмування знаходиться в центрі уваги багатьох відомих учених.

Проблемі навчання основ програмування присвячені дослідження В.В. Лапінського [6], В.Д. Руденка, О.М. Кривоноса [4], С.С. Жуковського [2], О.М. Спіріна, П.Г. Шевчука [10] та інших дослідників.

Важливість програмування як важливої складової шкільного курсу інформатики підкреслюють у своїх дослідженнях М.І. Жалдак, В.Д. Руденко.

Проблемам вибору першої мови програмування приділена увага у дослідженнях О.М. Кривоноса, П.Г. Шевчука [5] та С.С. Жуковського [2].

У своїх дослідженнях С.С. Жуковський переконливо доводить доцільність вибору мови C++ у якості першої мови програмування [2].

Дослідження О.М. Спіріна і П.Г. Шевчука розкривають особливості навчання учнів основ програмування на мові C# [11].

Проте недостатньо вивченими залишаються питання вивчення мови Python у якості першої мови програмування, а також проблеми, які виникають в учнів і вчителів під час вивчення Python.

**Мета статті** – визначити основні проблеми, які виникають під час навчання основ програмування на мові Python.

**Виклад основного матеріалу.** Як ми вже зазначали на початку статті, основи алгоритмізації та програмування є важливою складовою шкільного курсу інформатики. Саме цей компонент шкільної програми найбільше сприяє розвитку в учнів логічного та аналітичного мислення, формуванню у них інтересу до професій, пов'язаних з ІТ-індустрією. За допомогою мови програмування учень має можливість розробити власну програму, причому функціонал програми не обмежується рамками шкільної програми. Значна кількість Інтернет-ресурсів [9] присвячена програмуванню на різних мовах. Причому одні ресурси обмежуються описом синтаксису конкретної мови програмування, інші надають відомості з алгоритміки тощо.

Першим середовищем програмування, як це вже зазначалося, є Scratch. Це зручний засіб навчання основам програмування, проте його функціональні можливості обмежені. Саме тому на наступному етапі навчання програмування пропонуються для вивчення повноцінні мови програмування – Scratch і Free Pascal (Lazarus).

У 8 класі за програмою тема програмування вивчається 28 годин. Вона містить розділи: «Основи подійно- та об'єктно-орієнтованого програмування» та «Алгоритми роботи з об'єктами та величинами». Програмою передбачено 6 практичних робіт з основ алгоритмізації та програмування.

Автори підручника [8] пропонують розглядати вивчення програмування використовуючи середовища Lazarus та мову програмування Python (розглядаються 3 середовища програмування мовою Python).

У першому розділі розглядають створення проектів, хоча саму мову програмування та основи синтаксису починають вивчати у другому розділі, що є суперечить логіці навчання програмування. У самих підтемах виклад матеріалу є досить логічним і представлений для вивчення двох мов програмування. Під час викладу теоретичного матеріалу автори покроково пропонують вправи для закріплення. У вправах подається не тільки завдання, а й готовий програмний код.

Проте проводити дослідження, спираючись лише на аналіз змісту навчальної програми і підручників, нам вважається недоцільним. Тому в рамках даного дослідження було проведено опитування серед вчителів інформатики районних центрів Сумської області, які вивчають програмування з учнями 8 класу. З них 46,7 % мають стаж роботи більше 5 років, 26,7% – від 2 до 5 років та 26,7% – менше 2 років. Отже, більшість вчителів встигли набути певного досвіду викладання інформатики, і їх висновки про Python можна вважати такими, що заслуговують на увагу.

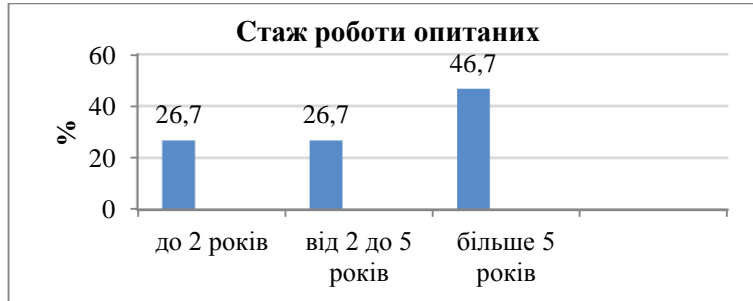


Рис. 1. Стаж роботи респондентів

Серед найбільш поширених для вивчення інформатики у 8 класі є підручник авторів Н.В. Морзе, О.В. Барна, В.П. Вембер [8] та альтернативний йому підручник авторів Й.Я. Ривкінда, Т.І. Лисенко, Л.А. Чернікової, В.В. Шакотько [3].

У першому підручнику у розділі програмування автори пропонують для вивчення мови програмування Object Pascal (середовище програмування Lazarus) та Python.

За результатами опитування логікою пояснення матеріалу за підручником [8] повністю задоволені 33,3 % вчителів. Переважна більшість (53,3%) вбачають можливість деяких змін у межах існуючих тем, 13,3% висловили бажання змінити послідовність викладу тем. Як виявилось, відчутною проблемою при вивченні програмування за вищезазначеним підручником є вивчення відразу двох мов програмування. Так вважають 53,3% опитуваних. Тому 33,3% вчителів не вивчали з учнями мову програмування Python, а обмежилися мовою Free Pascal.

Серед складнощів, пов'язаних із програмуванням на Python, найбільш поширеною є проблема відсутності середовища візуальної розробки програм, подібного до Scratch, Lazarus, Microsoft Visual Studio тощо. На це вказують 63,6% опитаних (рис.2). Пропоновані у підручнику середовища програмування не мають можливості створювати Windows-форму у режимі конструктора. В учнів це викликає додаткові труднощі, а для середнього учня це призводить до зниження мотивації до вивчення програмування, і як підсумок – відсутність успіхів у вивченні цього важливого розділу шкільної програми.



Рис. 2. Основні складнощі, які виникали під час вивчення мови Python:

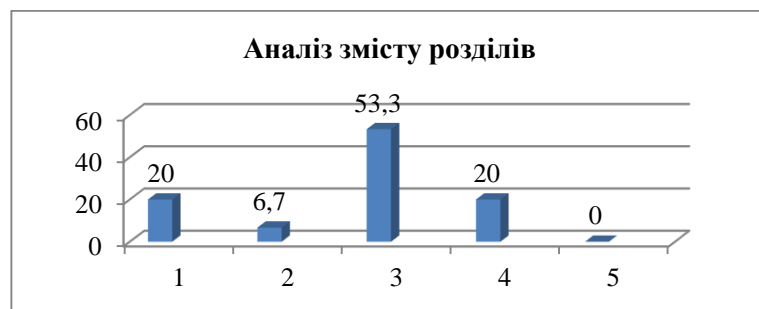
- 1 – робота з середовищем програмування;
- 2 – складності з синтаксисом мови Python;
- 3 – алгоритмічні помилки (важко реалізувати алгоритм розв'язання задачі на мові Python);
- 4 – складності, пов'язані з відсутністю середовища візуальної розробки програм

Вивчення різних середовищ програмування для однієї мови програмування ми вважаємо виправданим для студентів, які вже мають навички програмування, мають певний рівень самостійності. Але, на нашу думку, для учнів-початківців варто зупинитися на одному

середовищі програмування і вивчати саме його.

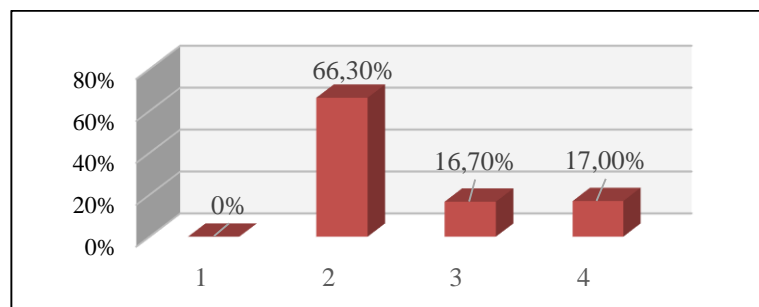
Значно менша кількість опитаних (27,3%) вважає основною проблемою складнощі синтаксису. Справді, синтаксис Python подібний до синтаксису C++, однак має й суттєві відмінності. Проте, якщо учень вивчає Python в якості першої мови програмування, йому легше звикнути до такого синтаксису. У подальшому, під час написання програм, дана проблема поступово нівелюється. У випадку паралельного вивчення Python і Lazarus проблема синтаксису постає особливо гостро. Не встигнувши звикнути до однієї мови програмування, учні ознайомлюються з іншою, потім знову повертаються до Python і т.д.

Наступне питання стосувалося змісту навчального матеріалу з основ програмування, який пропонується до вивчення (рис.3). Більшість опитаних (53,3%) вважають доцільним вивчати одну мову і одне середовище програмування. Значно менша кількість учителів висловлюють думку, що варто змінити тематику завдань на такі, що пов'язані з життям. Найменша кількість опитаних (6,7%) вважають, що треба змінити і мови, і середовища програмування. Для 20% опитаних зміст навчального матеріалу видається цілком оптимальним.



**Рис. 3. Думка про раціональність змісту розділів з основ програмування:**  
 1 – повністю задовольняє; 2 – доцільно змінити мови і середовище програмування, які вивчаються; 3 – доцільно залишити одну мову і одне середовище програмування; 4 – доцільно змінити тематику завдань на такі, що більше пов'язані із життям; 5 – доцільно підібрати більш абстрактні завдання.

Розділилася й думка про доцільність вивчення мови Python у школі. Найбільша кількість опитаних (66,3%) вважають, що програмування на мові Python слід вивчати з учнями у вигляді окремого розділу. Значно менша кількість респондентів висловлює думку про необхідність поглибленого вивчення Python, а 17% негативно ставляться до вивчення цієї мови у школі.



**Рис.4. Думка про доцільність вивчення мови Python у шкільному курсі інформатики:**  
 1 – вивчення мови Python є цілком виправданим у нинішньому вигляді; 2 – вивчення мови Python доцільно здійснювати у вигляді окремого розділу програми; 3 – вивчення мови Python доцільно поглибити; 4 – вивчати недоцільно

Проведене опитування частково підтверджує нашу думку про існуючі складнощі з вивченням першої мови програмування і мови Python, зокрема. Отже, основними проблемами є: проблема середовища програмування і проблема синтаксису.

Приступаючи до розгляду вказаних проблем, доцільно спочатку коротко охарактеризувати Python як мову програмування.

Розглянемо коротко особливості мови Python. Її було розроблено у 80-ті рр. XX ст. На даний час актуальною є версія 3.7.

Серед особливостей Python є мінімальний синтаксис, динамічна типізація даних, автоматичне управління пам'яттю. Python належить до інтерпретованих мов, тому програми, написані на Python, мають порівняно низьку швидкість виконання і потребують наявності на комп'ютері встановленого середовища програмування [12].

Популярність даної мови останнім часом зростає, тому для Python розроблено близько десяти середовищ програмування. Розглянемо лише ті, які пропонуються для вивчення у школі.

Одним із найпростіших є інтегроване середовище програмування IDLE. IDLE встановлюється на комп'ютер одночасно із інтерпретатором мови Python. Під час роботи з ним використовуються два вікна: вікно інтерпретатора (PythonShell) та вікно редактора коду (рис.5).

Системні вимоги Python версії 3.5 такі:

- операційна система Microsoft Windows 8/7/Vista/XP (64-bit включно);
- обсяг оперативної пам'яті: 256 Мб і більше;
- мінімальна тактова частота процесора – 800 МГц;
- вільний простір на диску: 50 МБ і більше;
- мінімальна роздільна здатність екрану – 1024x768 [12].

Отже, для комфортної роботи у середовищі IDLE достатньо мати комп'ютер з мінімальними характеристиками. Це значно розширює можливості середовища для навчання.

Інше середовище, пропоноване для вивчення у школі PyCharm (рис.6). PyCharm підтримує Python, Javascript, Coffeescript, Typescript, HTML/CSS, AngularJS, Node.js та інші мови програмування. PyCharm має широкі можливості для написання та редагування програм, створення проекту, навігації по проектах тощо [13].

Недолік: для роботи в даному середовищі з Python треба додатково встановлювати та підключати інтерпретатор Python.

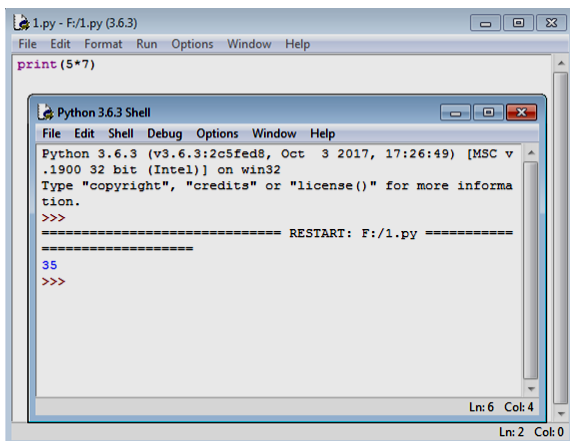


Рис. 5. Середовище IDLE

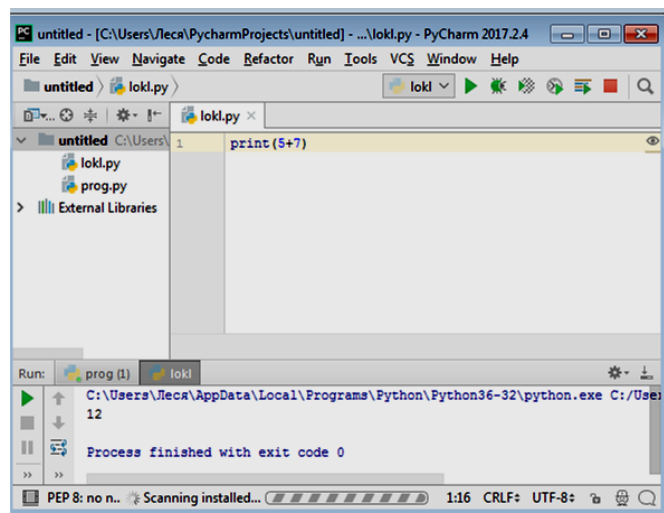


Рис. 6. Середовище PyCharm

Системні вимоги PyCharm такі:

- операційна система Microsoft Windows 10/8/7/Vista/2003/XP;
- оперативна пам'ять: 1 Гб мінімум, 2 Гб рекомендовано;
- вільний простір на диску: 300 МБ + не менше ніж 1 Гб для кешу;
- мінімальна роздільна здатність екрану — 1024x768;
- віртуальна машина JDK версії 1.6 і вище.

Існують версії PyCharm для операційних систем Mac та Linux.

Третє з пропонованих у підручнику середовищ програмування – он-лайн середовище CodingGround (рис.7).

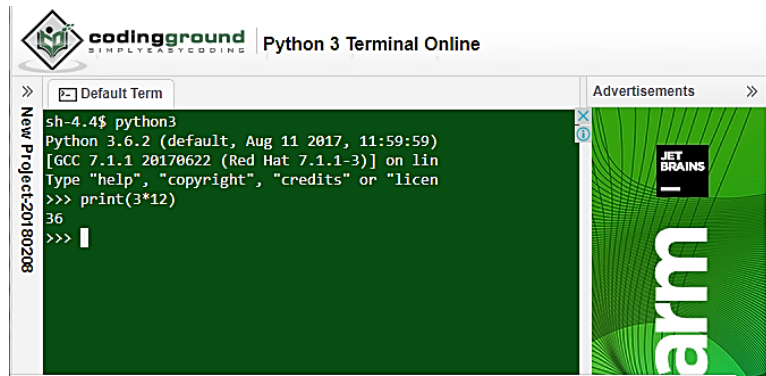


Рис. 7. Он-лайн-середовище CodingGround

Його інтерфейс подібний до IDLE, проте для роботи у цьому середовищі достатньо браузера [14].

Отже, середовищ програмування на мові Python існує значна кількість, необхідно лише дослідити їх дидактичні можливості і вибрати те середовище, застосування якого матиме найбільший ефект.

На відміну від Python, для Free Pascal існує візуальне середовище програмування (Lazarus), за допомогою якого можна створювати екранну форму у режимі конструктора. Тому в питанні середовища програмування однозначно виграє Free Pascal.

Наступною проблемою, пов'язаною з вивченням Python, вчителі визнають специфічний синтаксис цієї мови програмування. Не будемо розглядати синтаксис Python детально, а лише проілюструємо його на прикладах розв'язання алгоритмічних задач. Паралельно розглянемо розв'язки задач на мові Free Pascal (середовище Lazarus).

Задача 1. Написати програму, яка підраховує суму цифр трицифрового числа, введеного користувачем.

Приклад програми на мові Python:

```
chislo=int(input('Введіть трицифрове число  ')) #введення даних
dig1=chislo//100 #обчислення
chislo=chislo-dig1*100
dig2=chislo//10
dig3=chislo-dig2*10
s=dig1+dig2+dig3
print(s) #виведення даних
```

Приклад програми на мові Free Pascal

```
unit Unit1;
{$mode objfpc}{$H+}
interface
uses//використані модулі
  Classes, SysUtils, FileUtil, Forms, Controls, Graphics, Dialogs, StdCtrls;
type
{ TForm1 }
  TForm1 = class(TForm)
    Button1: TButton;
  procedure Button1Click(Sender: TObject);
  private
    { private declarations }
  public
    { public declarations }
  end;
var
  Form1: TForm1;
```

```

implementation
{$R *.lfm}
{ TForm1 }
procedure TForm1.Button1Click(Sender: TObject);
var chislo, dig1, dig2, dig3, s: integer; // оголошення змінних
begin
// виклик вікна введення даних
chislo := StrToInt(InputBox('Введення числа', 'Введіть трицифрове число', ''));
dig1 := chislo div 100; // обчислення
chislo := chislo - dig1 * 100;
dig2 := chislo div 10;
dig3 := chislo - dig2 * 10;
s := dig1 + dig2 + dig3;
// виведення результату у вікно повідомлення
MessageDlg(IntToStr(s), mtInformation, [mbOk], 0);
end;
end.

```

Задача 2. Написати програму, яка реалізує гру «Відгадай число» між комп'ютером та користувачем. Користувач за 6 спроб має відгадати «задумане» число з діапазону від 1 до 20. Якщо користувач відгадає число за 6 спроб, то на екран виводиться повідомлення про кількість спроб, якщо ні — виводиться «задумане» число.

Приклад програми на мові Python

```

import random # підключення модуля для генерації випадкових чисел
guesses Taken = 0 # задання початкового значення для циклу
print('Привіт! Як тебе звати?')
myName = input ()
number = random.randint(1, 20) # генерація випадкового числа від 1 до 20
print('Отже, ' + myName + ', Я задумав число від 1 до 20.')
# задання циклу з передумовою для реалізації багаторазового виконання команд
while guesses Taken < 6:
print('Спробуй відгадати.')
guess = input ()
guess = int(guess)
guessesTaken = guessesTaken + 1
if guess < number: # перевірка відповідності введеного та задуманого чисел
print('Твоє число менше від задуманого.')
if guess > number:
print('Твоє число більше за задумане.')
if guess == number:
break # вихід із циклу
if guess == number:
guessesTaken = str(guessesTaken)
print('Дуже добре, ' + myName + '! Для відгадування тобі знадобилося ' +
guessesTaken + ' разів!')
# виведення вікна повідомлення із задуманим числом у разі програшу
if guess != number:
number = str(number)
print('Ні. Число, яке я задумав - ' + number)

```

Приклад програми на мові Free Pascal

```

unit Unit1;
{$modeobjfpc}{$H+}
interface
uses Classes, SysUtils, FileUtil, Forms, Controls, Graphics, Dialogs, StdCtrls;

```

```

type
  { TForm1 }
  TForm1 = class(TForm)
    Button1: TButton;
    Edit1: TEdit;
    Label1: TLabel;
    Label2: TLabel;
  procedure Button1Click(Sender: TObject);
  private
    { privatedeclarations }
  public
    { publicdeclarations }
  end;
var
  Form1: TForm1;
implementation
  {$R *.lfm}
  { TForm1 }
  procedure TForm1.Button1Click(Sender: TObject);
  //оголошення змінних
  varquessestaken, quess, number:integer;
  myName:string;
  // початок процедури
  begin
    myName:=Edit1.Text;//присвоювання значення текстового поля змінній
    Randomize;//генерація випадкового числа від 1 до 20
    number:=random(20)+1;
    //виведення вікна повідомлення
    MessageDlg('Отже '+ myName + ', я задумав число від 1 до 20. Вгадай
    його!Продовжити?', MtConfirmation, [mbYes, mbNo],0);
    // задання початкового значення для циклу
    quessestaken:=0;
    // задання циклу з передумовою дляреалізації багаторазового виконання команд
    while quessesTaken<6 do
    begin
      quessestaken:=quessestaken+1;
      quess:=StrToInt(InputBox('Введення числа','Спроба '+IntToStr(quessestaken),"));
      // перевірка відповідності введеного та задуманого чисел
      if quess<number then
        MessageDlg('Твоє число менше від задуманого', MtInformation, [mbOk],0);
      ifquess>numberthen
        MessageDlg('Твоє число більше від задуманого', MtInformation, [mbOk],0);
      ifquess=numberthen
        begin
          MessageDlg('Молодець! Для відгадування тобі знадобилось ' +
          IntToStr(quessestaken)+' спроб', MtInformation, [mbOk],0);
          break// вихідіз циклу
        end;
      end;
    // виведення вікна повідомлення із задуманим числому разі програшу
    if quess<>number then
      MessageDlg('Число, яке я задумав - ' + IntToStr(number), MtInformation, [mbOk],0);
    end;
  end.

```



Проаналізувавши текст програм, можна зробити висновок про те, що програма, написана на Python, має значно менший код, а отже, учням легше його зрозуміти. Тому для учнів, для яких Python є першою мовою програмування, специфічний синтаксис є швидше перевагою, ніж недоліком.

Питання змісту і послідовності вивчення основ програмування на мові Python, а також розробка відповідного навчально-методичного забезпечення виходять за межі даної статті і вимагають окремого дослідження.

**Висновки та перспективи подальших наукових розвідок.** На основі проведеного дослідження можна зробити такі висновки:

1. Вивчення мови Python у якості першої мови програмування має певні перспективи. У цьому випадку учням доводиться відразу вивчати синтаксис Python, і проблема синтаксису мови програмування певною мірою нівелюється.

2. Вивчення мови Python доцільно здійснювати окремо від мови Free Pascal. Це запобігає плутанині з синтаксисом мови програмування під час написання програм.

3. Значні проблеми у вивченні Python викликає відсутність візуального середовища розробки програм. У середовищах, описаних у підручнику, неможливо створити екранну форму у режимі конструктора. Це якраз той момент у навчанні основ програмування, який здатний зацікавити учнів.

4. Проблема синтаксису мови Python, який відрізняється від C/C++ - подібних мов програмування, не є визначальною. Код програми, написаної на Python, значно лаконічніший, ніж код програми, написаний на Free Pascal, а отже, й зрозумілішим для учнів.

Перспективами подальших досліджень у даному напрямі є:

1. Розробка моделі навчання основ програмування на мові Python, обґрунтування.
2. Розробка методичної системи навчання програмуванню на мові Python.
3. Добір оптимального середовища програмування для учнів.
4. Розробка змістового компоненту навчання програмування на мові Python.

#### **СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ**

1. Державний стандарт базової і повної загальної середньої освіти / затверджено постановою Кабінету Міністрів України від 23 листопада 2011 р. №1392. // Комп'ютер у школі та сім'ї. – 2012. – №1. – С.33-38.
2. Жуковський С.С., Коротун О.В. Про перспективу введення мови програмування C++ в навчальний процес загальноосвітніх навчальних закладів // Комп'ютер у школі та сім'ї. – 2014. – №1. – С. 23-25.
3. Інформатика : підруч. для 8-го кл. загальноосвіт. навч. закл. / Й.Я.Ривкінд [та ін.]. — Київ : Генеза, 2016. — 288 с.
4. Кривонос О.М. Використання задачного підходу в процесі навчання програмування майбутніх учителів інформатики // Інформаційні технології і засоби навчання. – 2014. – Т. 40, вип. 2. – С. 83-91. – Режим доступу: [http://nbuv.gov.ua/UJRN/ITZN\\_2014\\_40\\_2\\_10](http://nbuv.gov.ua/UJRN/ITZN_2014_40_2_10)
5. Кривонос О.М., Шевчук П.Г. Комп'ютерно-інтегровані технології: освіта, наука, виробництво. – Луцьк, 2011. – Вип.№5. – С.148-150.
6. Лапінський В. В. Проблема вибору першої мови програмування – сьогоднішнє бачення / В. В. Лапінський // Комп'ютер у школі та сім'ї. – 2014. – № 1. – С. 14-17. – Режим доступу: [http://nbuv.gov.ua/UJRN/komp\\_2014\\_1\\_4](http://nbuv.gov.ua/UJRN/komp_2014_1_4).
7. Морзе Н.В. Методика навчання інформатики. Частина 4. Методика навчання основ алгоритмізації та програмування. Навч. посібник. У 4 ч. За ред. М.І. Жалдака. – Київ: Навчальна книга, 2004. – 368 с.
8. Морзе Н.В., Барна О.В., Вембер В.П. Інформатика : підруч. для 8 кл. загальноосвіт. навч. закладів. – К.: УОВЦ «Оріон», 2016. – 240 с.
9. Програмування мовою Python [Електронний ресурс]. Режим доступу: <https://www.youtube.com/watch?v=iRLBV4ZG3qA>
10. Шевчук П. Г. Програмно-технологічні умови використання мови C# для навчання програмування в загальноосвітніх навчальних закладах // Науковий часопис НПУ імені

- М. П. Драгоманова. Серія 2 : Комп'ютерно-орієнтовані системи навчання. 2011. – № 10. – С. 80-84. – Режим доступу: [http://nbuv.gov.ua/UJRN/Nchnpu\\_2\\_2011\\_10\\_11](http://nbuv.gov.ua/UJRN/Nchnpu_2_2011_10_11)
11. Шевчук П.Г. Методика навчання програмування учнів класів технологічного профілю на основі використання мови С# : дис. ... кандидата педагогічних наук : 10.02.19 / Шевчук Петро Георгійович. – К., 2013. – 319 с.
12. About Python | Python.org [online]. Access mode: <https://www.python.org/about/>
13. PyCharm: Python IDE for Professional Developers by JetBrains [online]. Access mode: <https://www.jetbrains.com/pycharm/?fromMenu>. Available: Feb.3, 2018.
14. Python 3 Terminal Online [online]. Access mode: [https://www.tutorialspoint.com/python3\\_terminal\\_online.php](https://www.tutorialspoint.com/python3_terminal_online.php)

**Мицкан Л., Вербицкая Т., Базурин В.Н. Сравнительный анализ языков программирования Python и Free Pascal как первых языков программирования для учеников 8 класса.**

*В статье раскрываются проблемы изучения языка Python в качестве первого языка программирования. На основании эмпирических данных сделан вывод о том, что основными проблемами есть отсутствие визуальных сред программирования для Python, специфический синтаксис этого языка и одновременное изучение Python и Free Pascal. Авторы доказывают, что данные проблемы частично решаются путем подбора среды программирования и разработки соответствующего методического обеспечения. Синтаксис языка Python, несмотря на свою специфику, лаконичный и лучше понятен ученикам. Изучение языка программирования Python целесообразно осуществлять отдельно от изучения Free Pascal.*

**Ключевые слова:** программирование, язык программирования, Python, Free Pascal, Lazarus, информатика.

**Mitskan L., Verbytska T., Bazurin V.M. Comparative analysis of programming languages Python and Free Pascal as the first programming languages for pupils of the 8th grade.**

*The article reveals the problems of learning the Python language as the first programming language. On the basis of empirical data, a paper is made that the main problems are the lack of visual programming environments for Python, the specific syntax of this language and the simultaneous study of Python and Free Pascal. The authors argue that these problems are partially solved by selecting a programming environment and developing appropriate methodological support. The syntax of the Python language, despite its specifics, is concise and better understood by the students. It is advisable to study the Python programming language separately from the study of Free Pascal.*

**Keywords:** programming, programming language, Python, Free Pascal, Lazarus, computer science.