TRƯỜNG ĐẠI HỌC VINH

**VIỆN KỸ THUẬT VÀ CÔNG NGHỆ**



BÁO CÁO THỰC HÀNH

**KỸ THUẬT LẬP TRÌNH**

MÃ HỌC PHẦN: ELE22004

SVTH: NGUYỄN BÁ ĐẠT

MSSV: 245752021610160

GVHD: MAI THẾ ANH

NGHỆ AN, 2025

MỤC LỤC

[**Bài 1. Thực hiện các thuật toán bằng phần mềm Flowgorithm** 3](#_Toc215530584)

[**1. Mục đích:** 3](#_Toc215530585)

[**2. Các bước thực hiện và kết quả:** 3](#_Toc215530586)

[**Bài 2. Các cú pháp, kiểu dữ liệu, lệnh điều khiển trong lập trình Python** 6](#_Toc215530587)

[**1.** **Mục đích:** 6](#_Toc215530588)

[**2.** **Các bước thực hiện và kết quả:** 6](#_Toc215530589)

[**Bài 3. Lập trình hàm trong Python** 13](#_Toc215530590)

[**1.** **Mục đích:** 13](#_Toc215530591)

[**2.** **Các bước thực hiện và kết quả:** 13](#_Toc215530592)

[**Bài 4. Các kiểu dữ liệu có cấu trúc trong Python** 18](#_Toc215530593)

[**1. Mục đích:** 18](#_Toc215530594)

[**2. Các bước thực hiện và kết quả:** 18](#_Toc215530595)

[**Bài 5. Thiết kế module trong Python** 31](#_Toc215530596)

[**1.** **Mục đích:** 31](#_Toc215530597)

[**2.** **Các bước thực hiện và kết quả:** 31](#_Toc215530598)

[**Bài 6: Lập trình hướng đối tượng trong Python** 37](#_Toc215530599)

[**1.** **Mục đích:** 37](#_Toc215530600)

[**2.** **Các bước thực hiện và kết quả:** 37](#_Toc215530601)

[**Bài 7: Thao tác trên tập tin và thư mục trong Python** 41](#_Toc215530602)

[**1.** **Mục đích:** 41](#_Toc215530603)

[**2.** **Các bước thực hiện và kết quả:** 41](#_Toc215530604)

[**Bài 8: Lập trình giao diện trong Python** 46](#_Toc215530605)

[**1.** **Mục đích:** 46](#_Toc215530606)

[**2.** **Các bước thực hiện và kết quả:** 46](#_Toc215530607)

[**TÀI LIỆU THAM KHẢO** 50](#_Toc215530608)

**Bài 1. Thực hiện các thuật toán bằng phần mềm Flowgorithm**

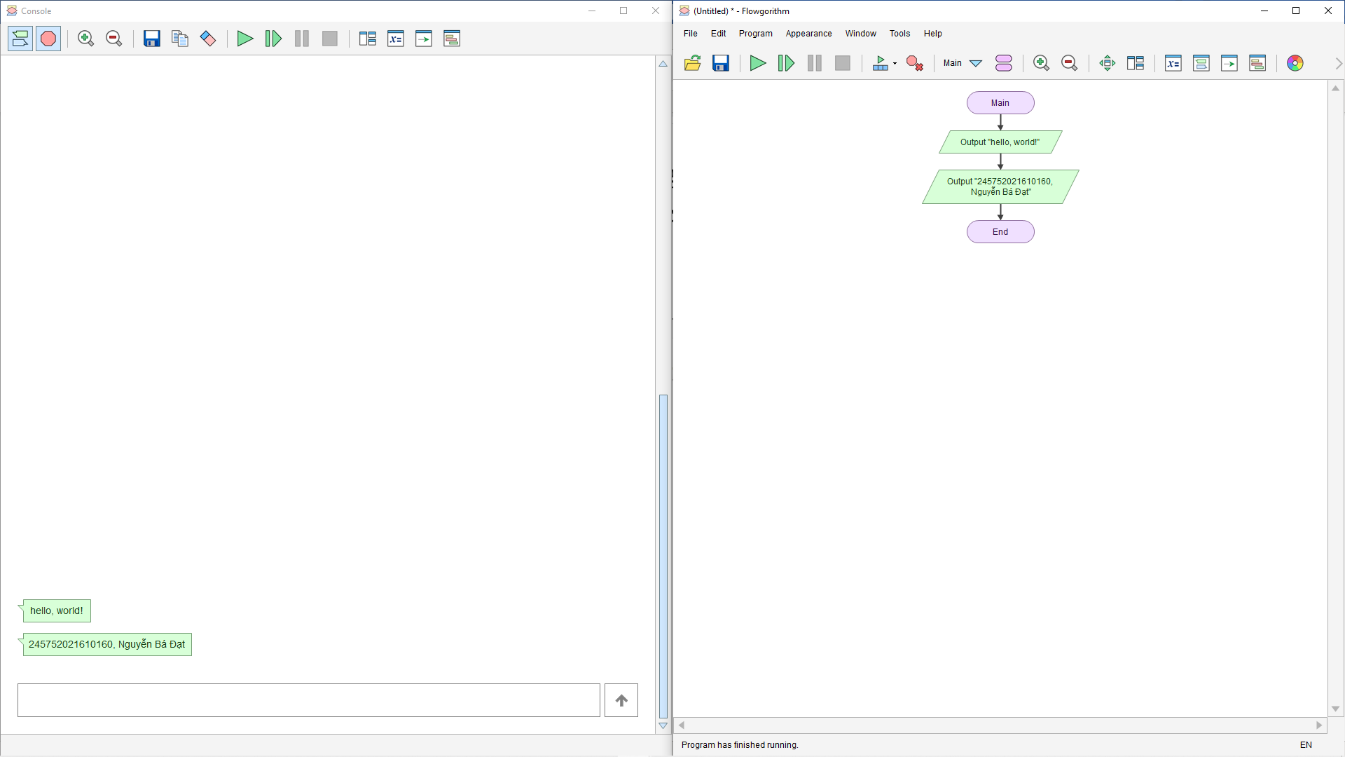
**1. Mục đích:**

- Sử dụng phần mềm Flowgorithm trong thiết kế và biểu diễn thuật toán;

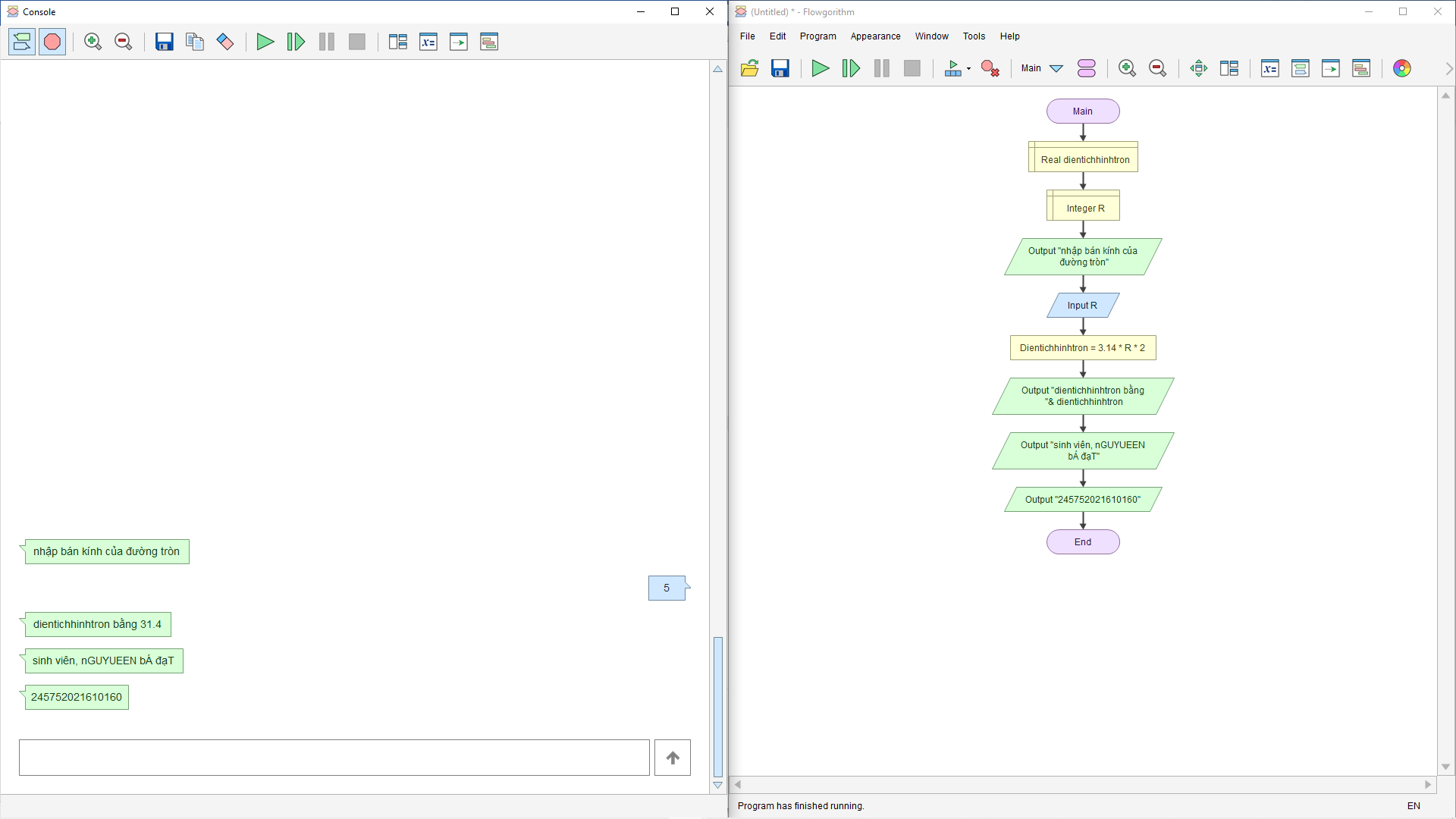
- Xây dựng thuật toán cho các bài toán cụ thể trên Flowgorithm

**2. Các bước thực hiện và kết quả:**

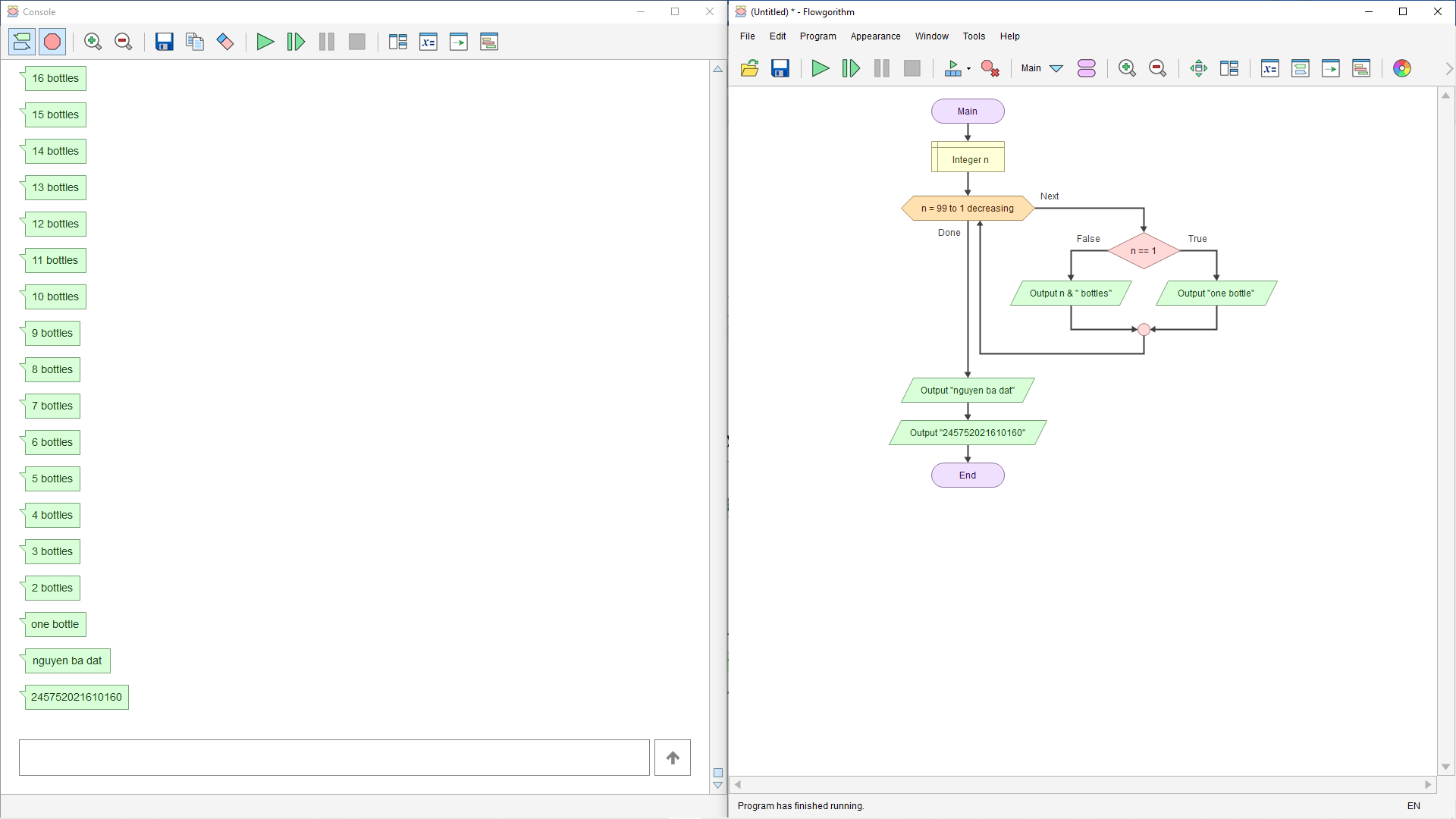
1.1. Câu chào hello:



1.2. Tính diện tích hình tròn:



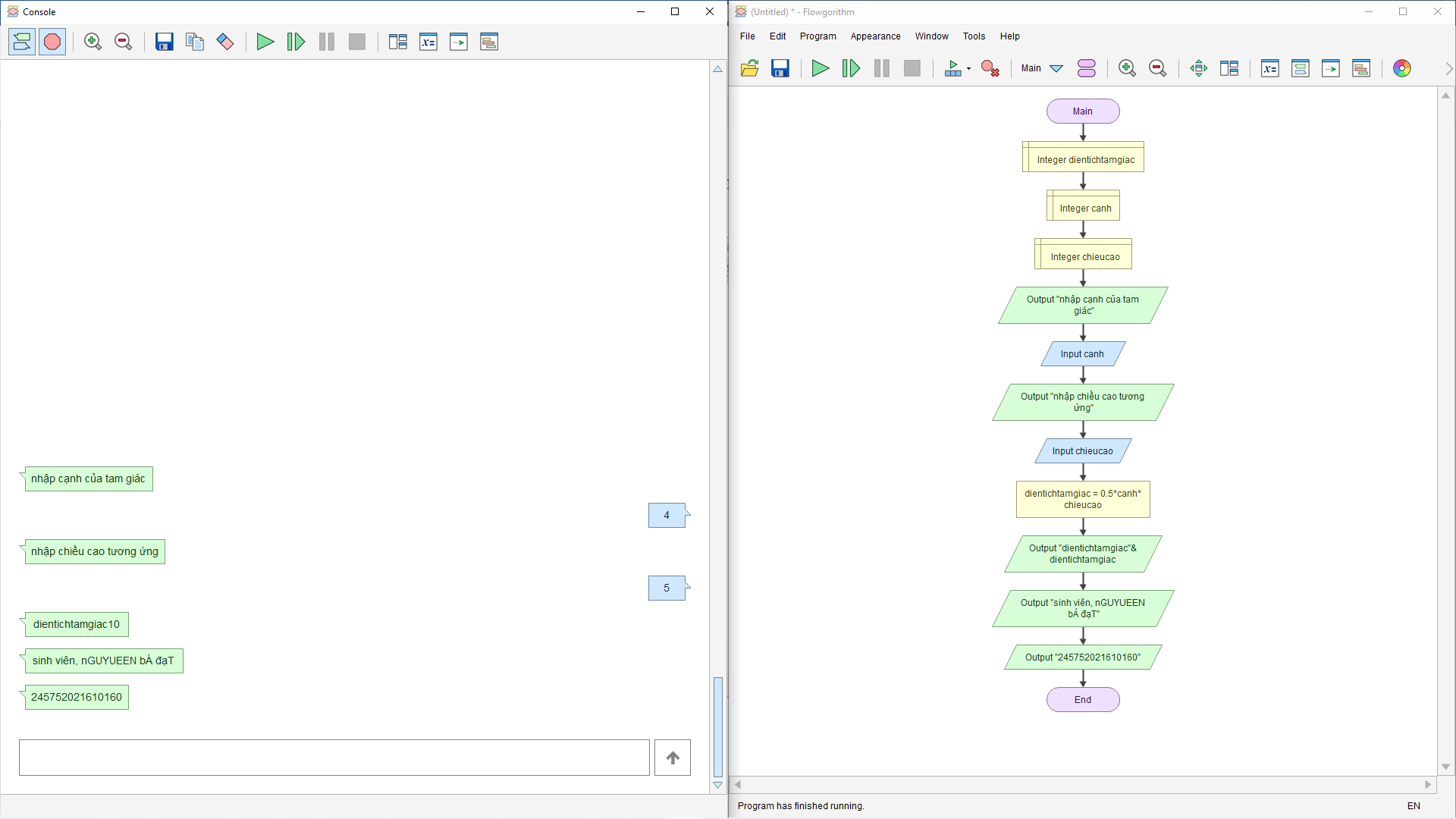
1.3. Nhập và in số lượng chai sử dụng vòng lặp for:



1.4. Tính diện tích hình tròn sử dụng hàm (chương trình con):



1.5. Viết chương trình nhập và cạnh và chiều cao tương ứng của một tam giác và in ra màn hình diện tích tam giác



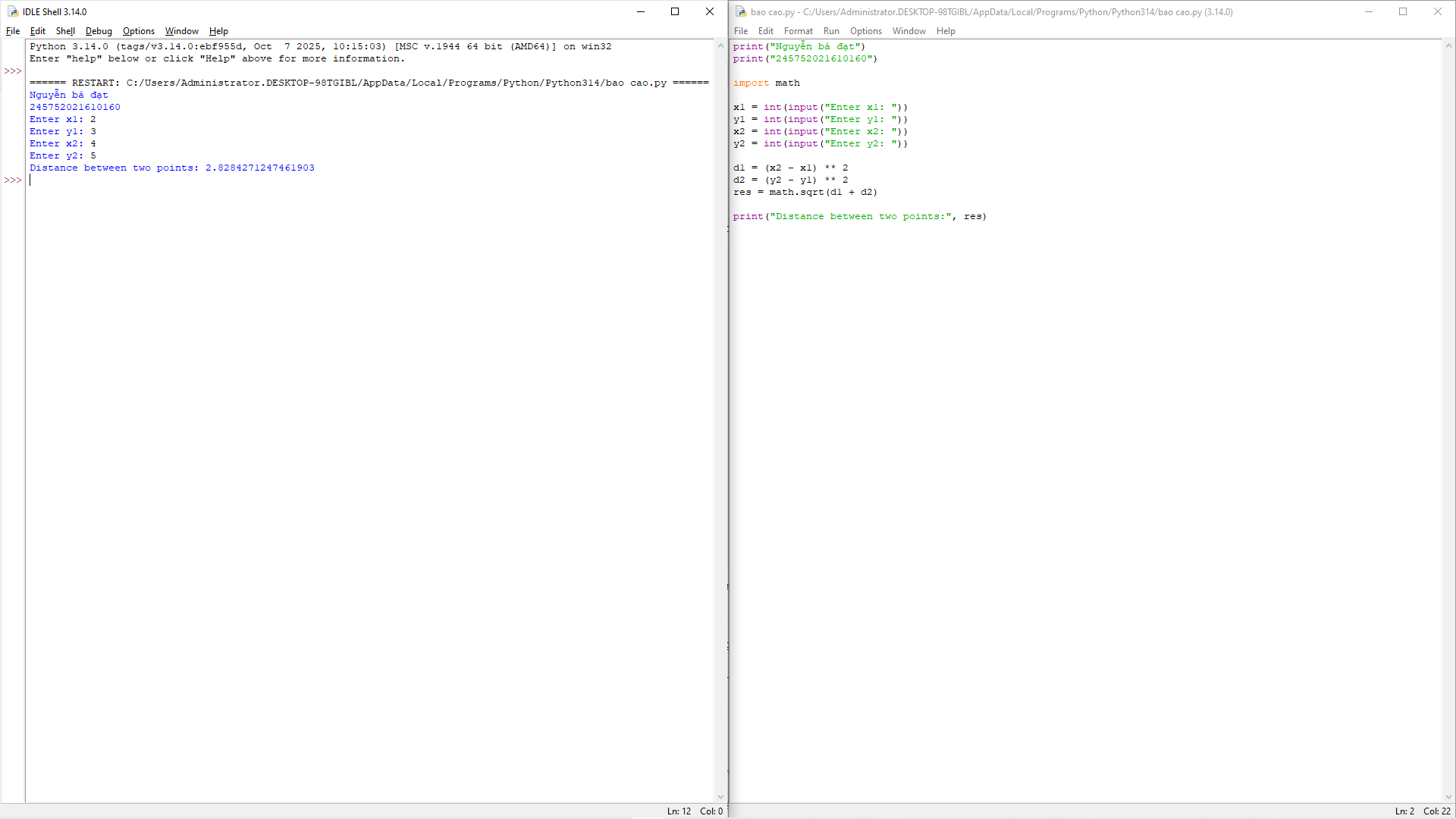
**Bài 2. Các cú pháp, kiểu dữ liệu, lệnh điều khiển trong lập trình Python**

1. **Mục đích:**

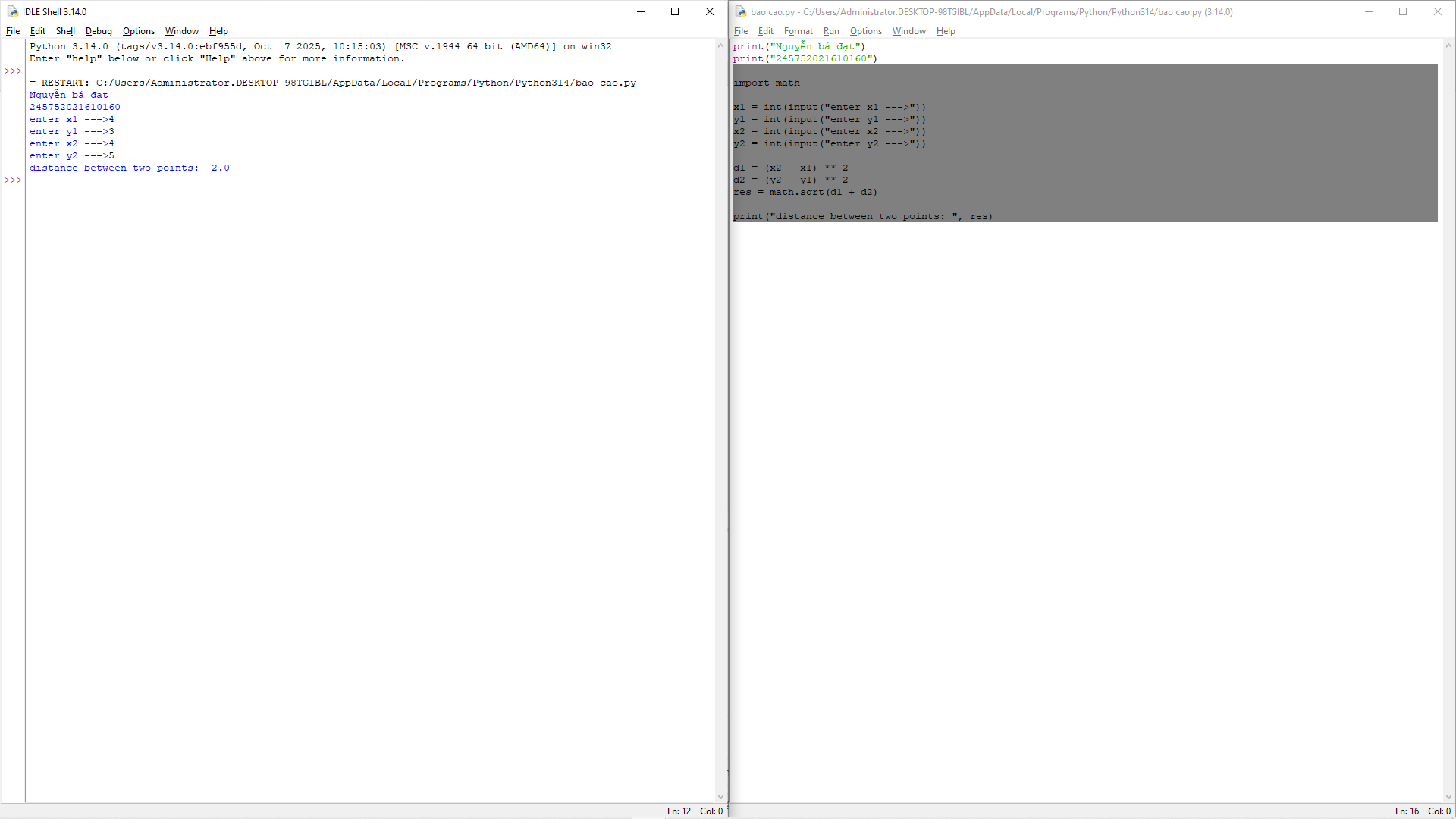
* Giúp sinh viên hiểu được Các cú pháp, kiểu dữ liệu, lệnh điều khiển trong lập trình Python

1. **Các bước thực hiện và kết quả:**
2. Viết đoạn chương trình sau và sửa lỗi:

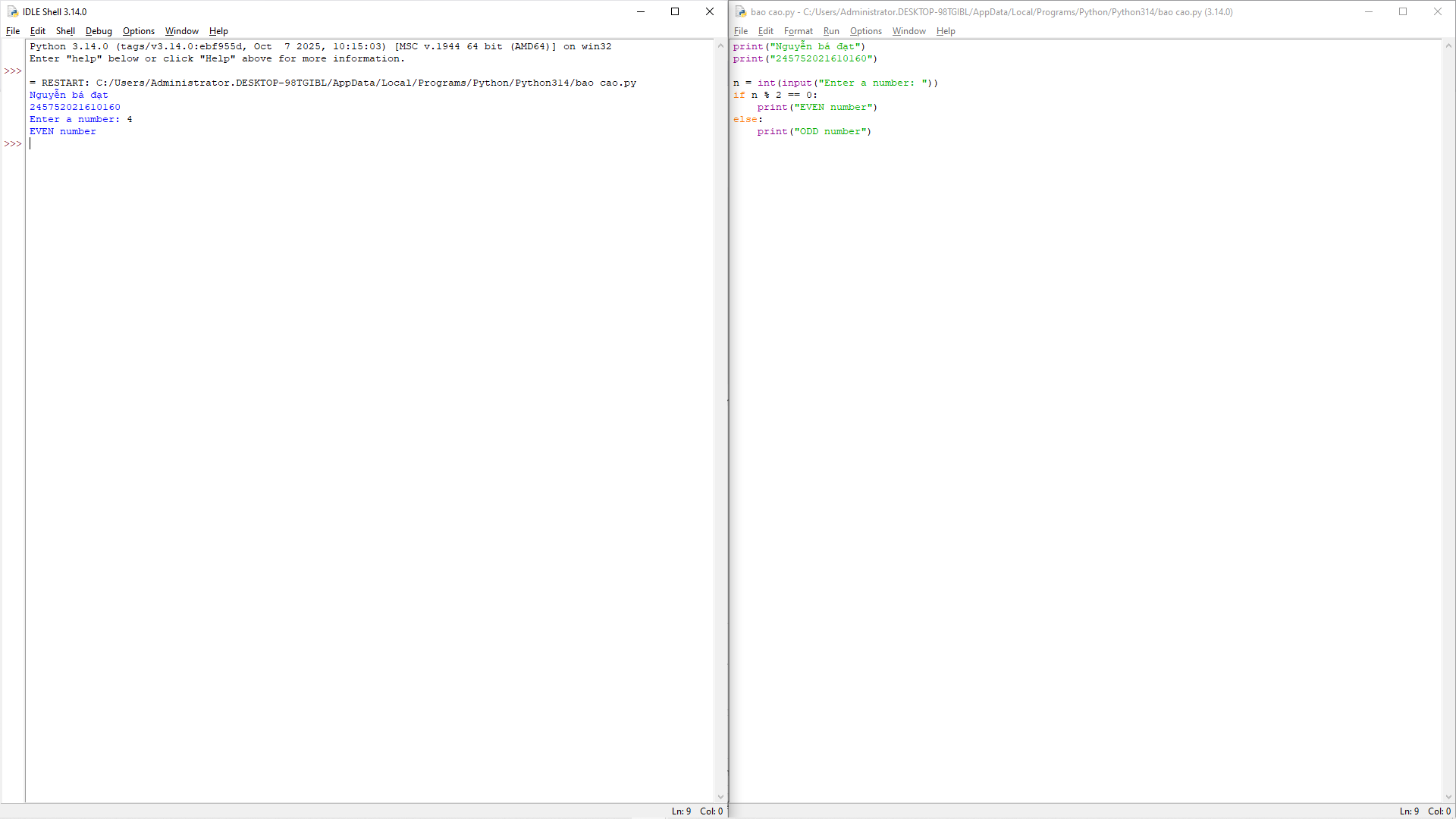
Đoạn code sau khi sửa lỗi và kết quả là:



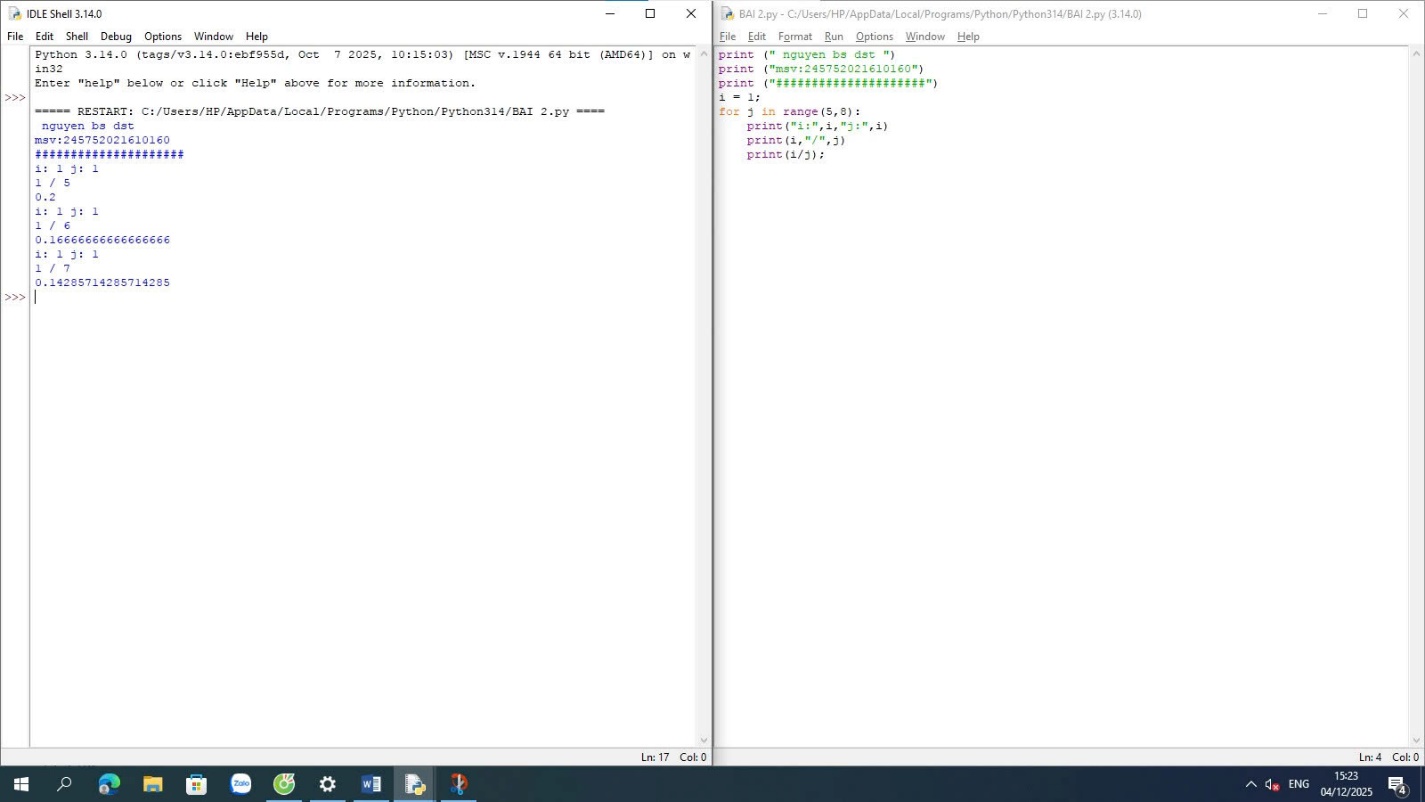
2. Viết chương trình nhập hai điểm và tính khoảng cách:



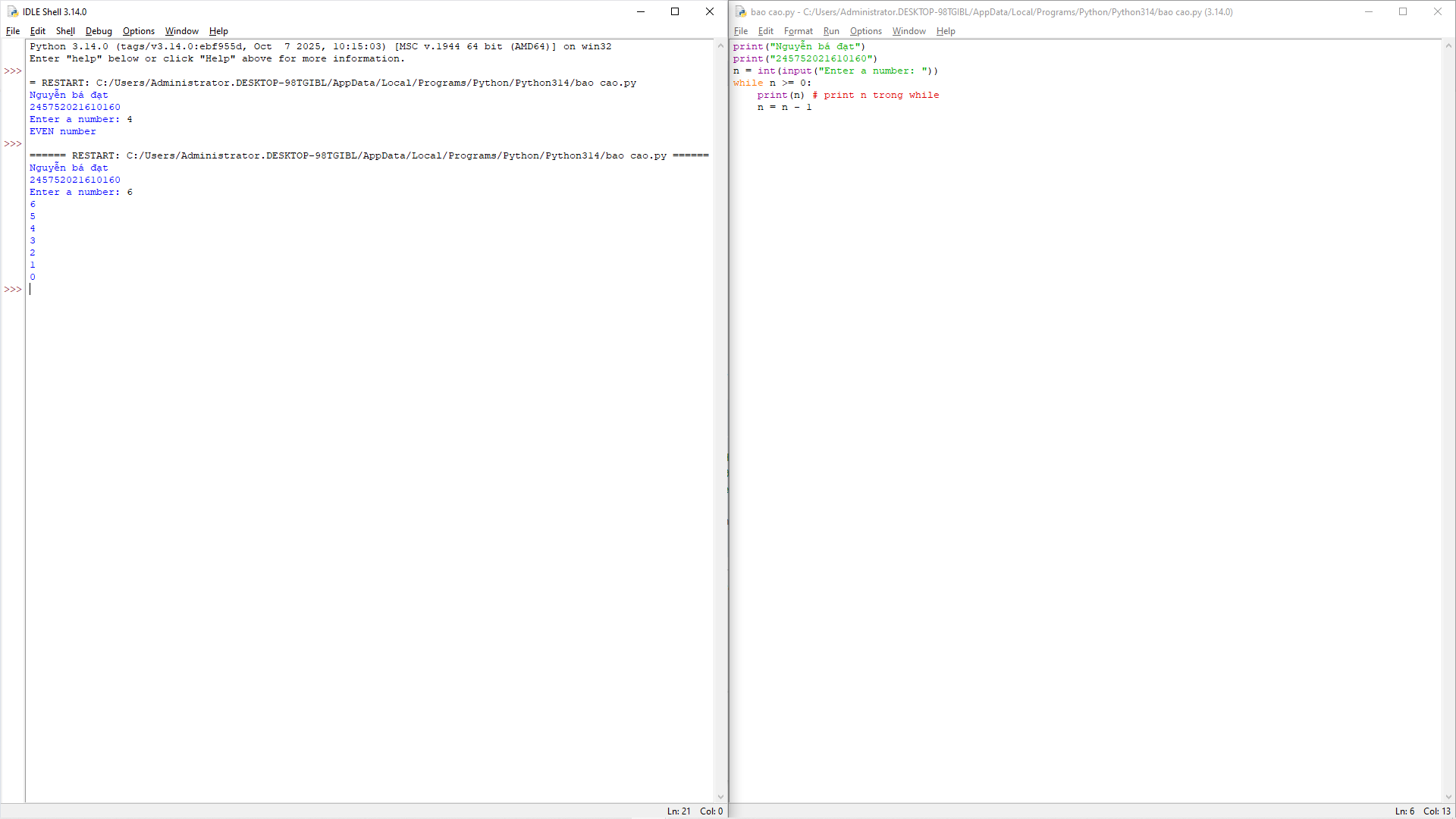
3. Viết chương trình nhập vào một số và kiểm tra số đó là chẵn hay lẻ, in thông báo ra màn hình:



4. Viết chương trình in ra màn hình số nghịch đảo và kết quả dưới dạng thập phân của một dãy số tự nhiên trong khoảng (a,b):



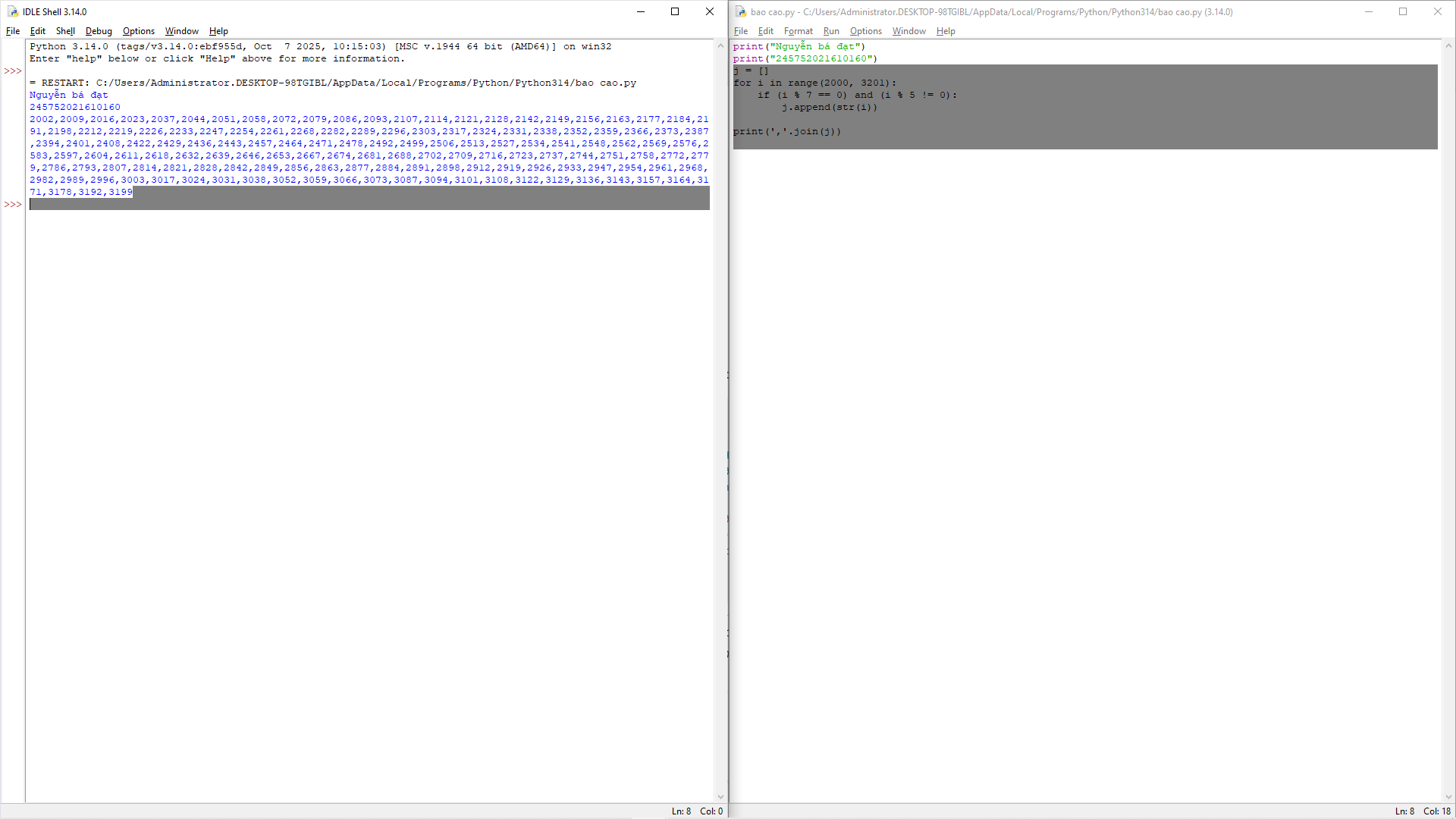
5. Viết chương trình nhập vào một số tự nhiên n > 0, in ra màn hình các số tự nhiên giảm dần từ n đến 0, mỗi ký tự in trên 1 hàng:



6. Viết chương trình tìm tất cả các số chia hết cho 7 nhưng không phải bội số của 5, nằm trong đoạn 2000 và 3200 (tính cả 2000 và 3200). Các số thu được sẽ được in thành chuỗi trên một dòng, cách nhau bằng dấu phẩy:

- Hàm range(): Dùng để tạo ra một chuỗi số liên tiếp từ 2000 đến 3200.

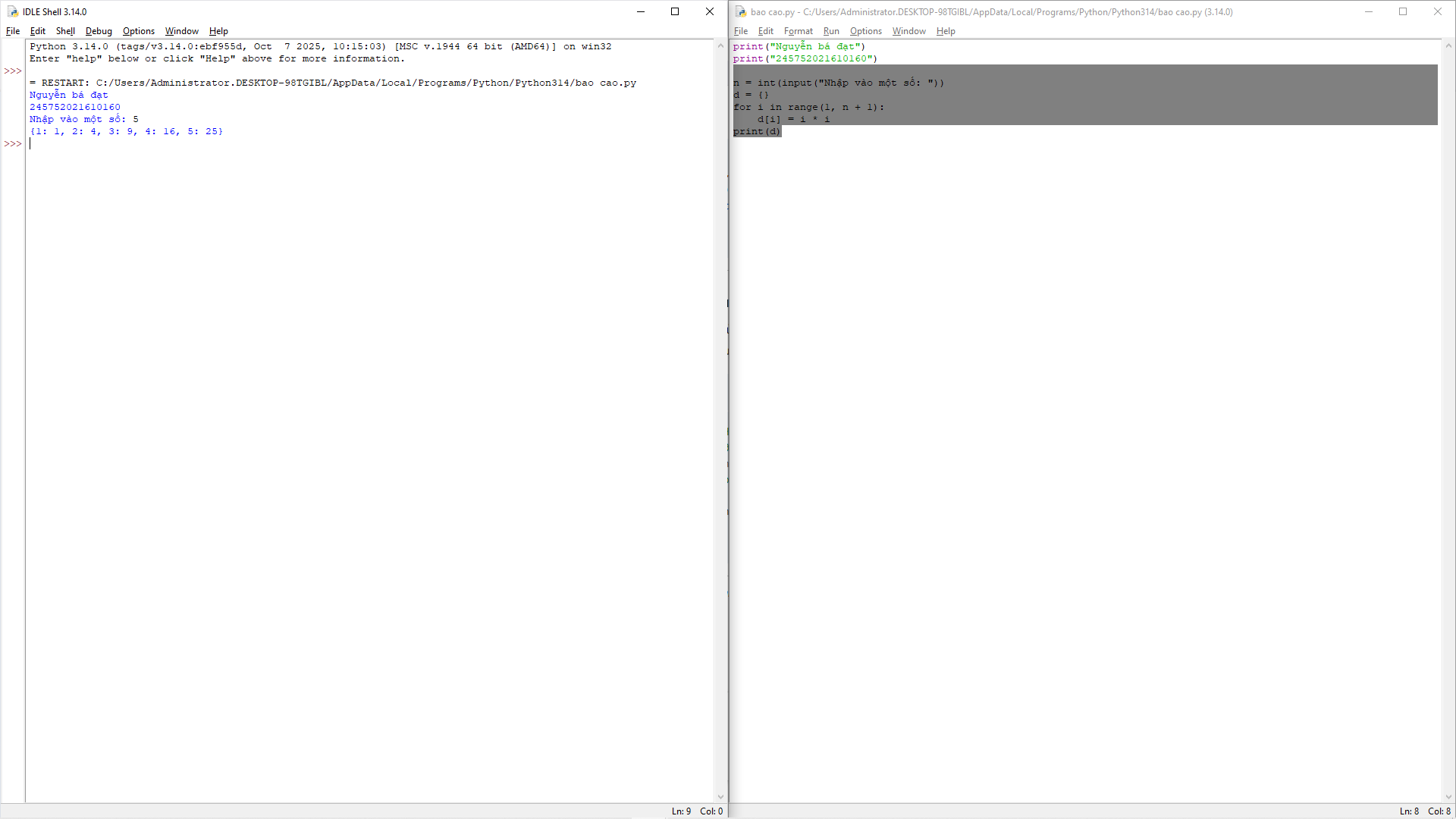
- Toán tử (%): Dùng để kiểm tra tính chia hết.



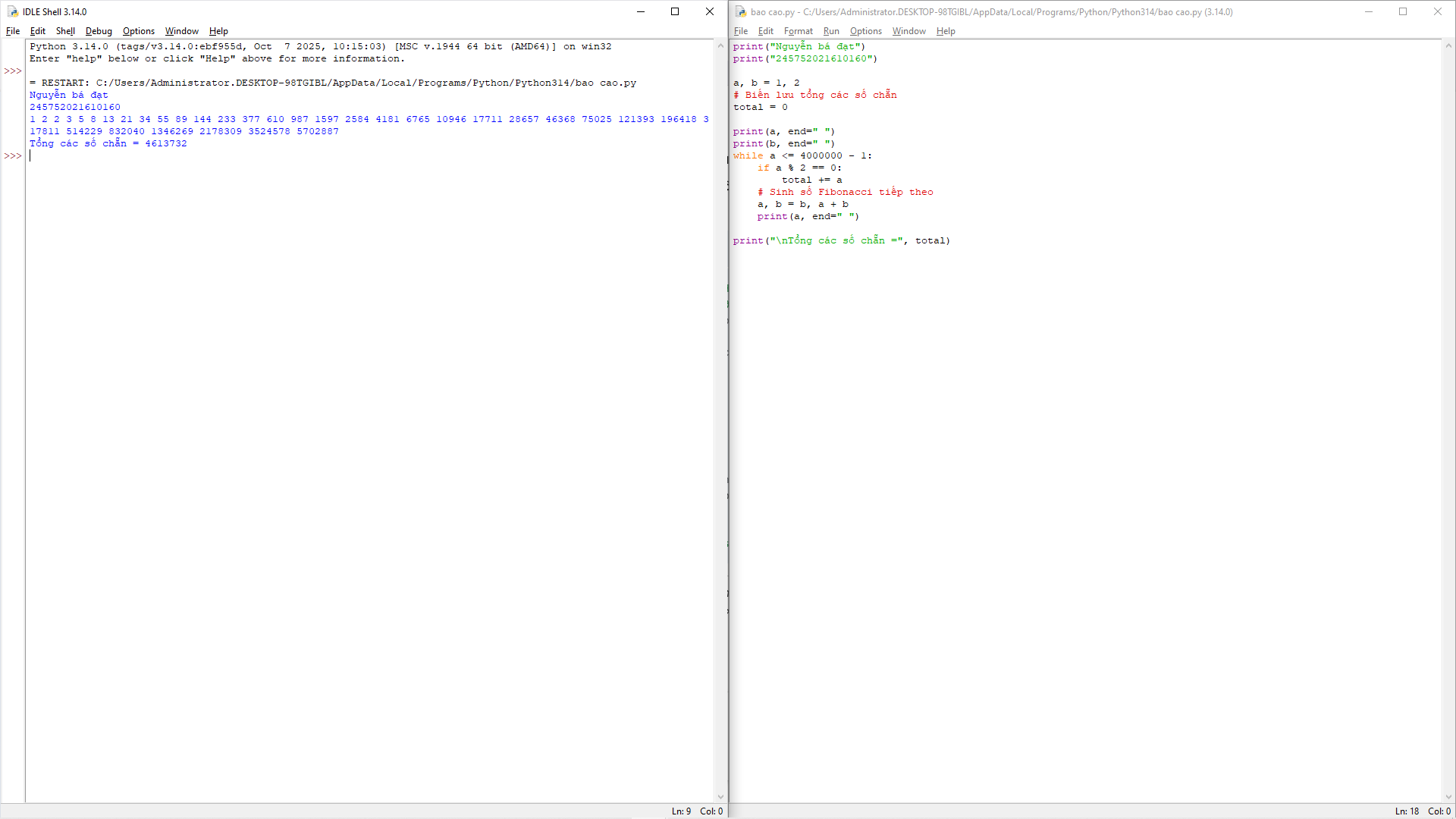
7. Với số nguyên n nhất định, hãy viết chương trình để tạo ra một dictionary chứa (i, i\*i) như là số nguyên từ 1 đến n (bao gồm cả 1 và n) sau đó in ra dictionary này:

- Nhận đầu vào từ người dùng và chuyển nó sang kiểu số nguyên (int).

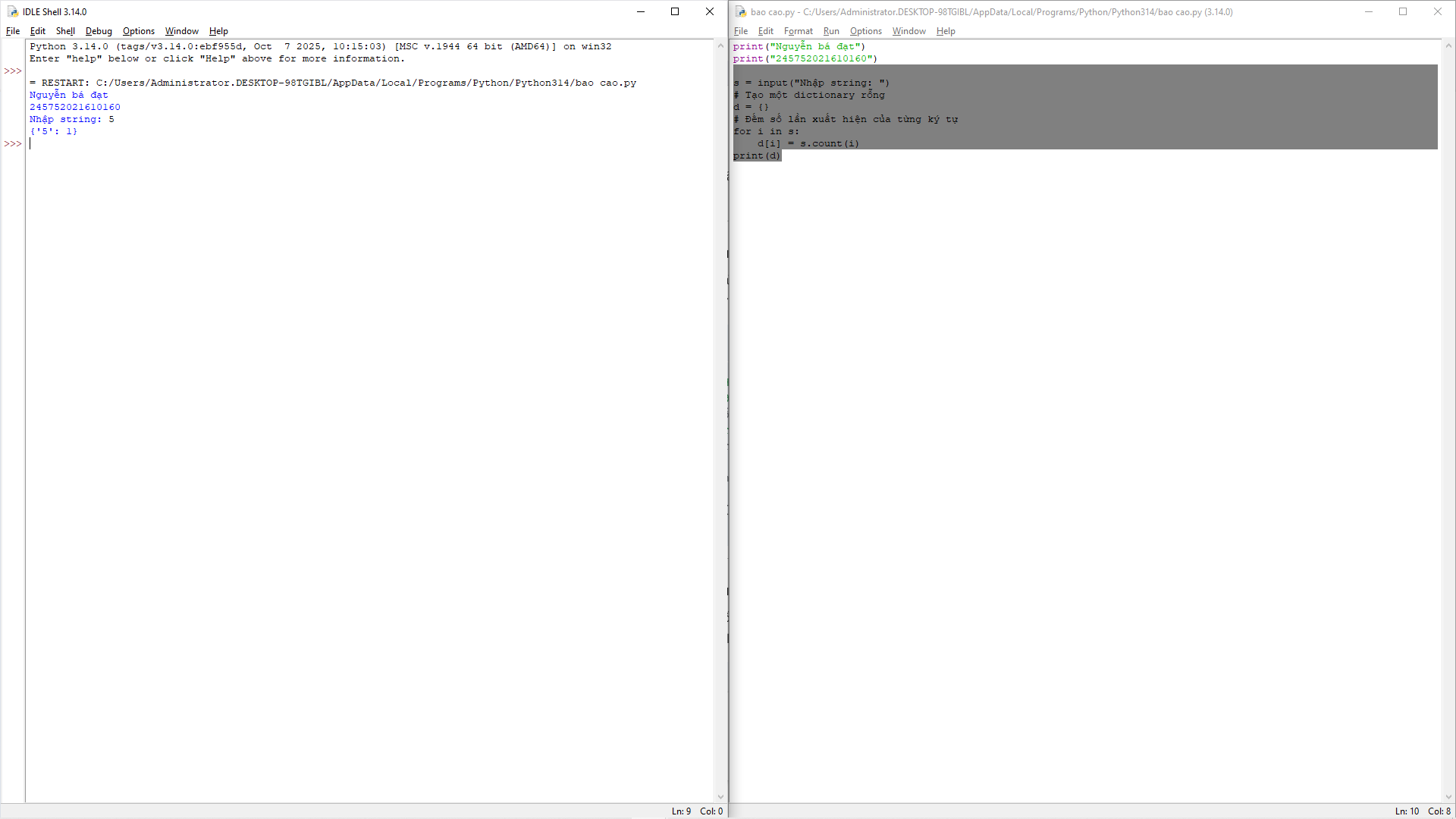
- Đây là Dictionary Comprehension. Nó lặp qua từng số i trong range() và tạo ra một cặp khóa-giá trị: Khóa là i, và Giá trị là bình phương của nó (i \* i).



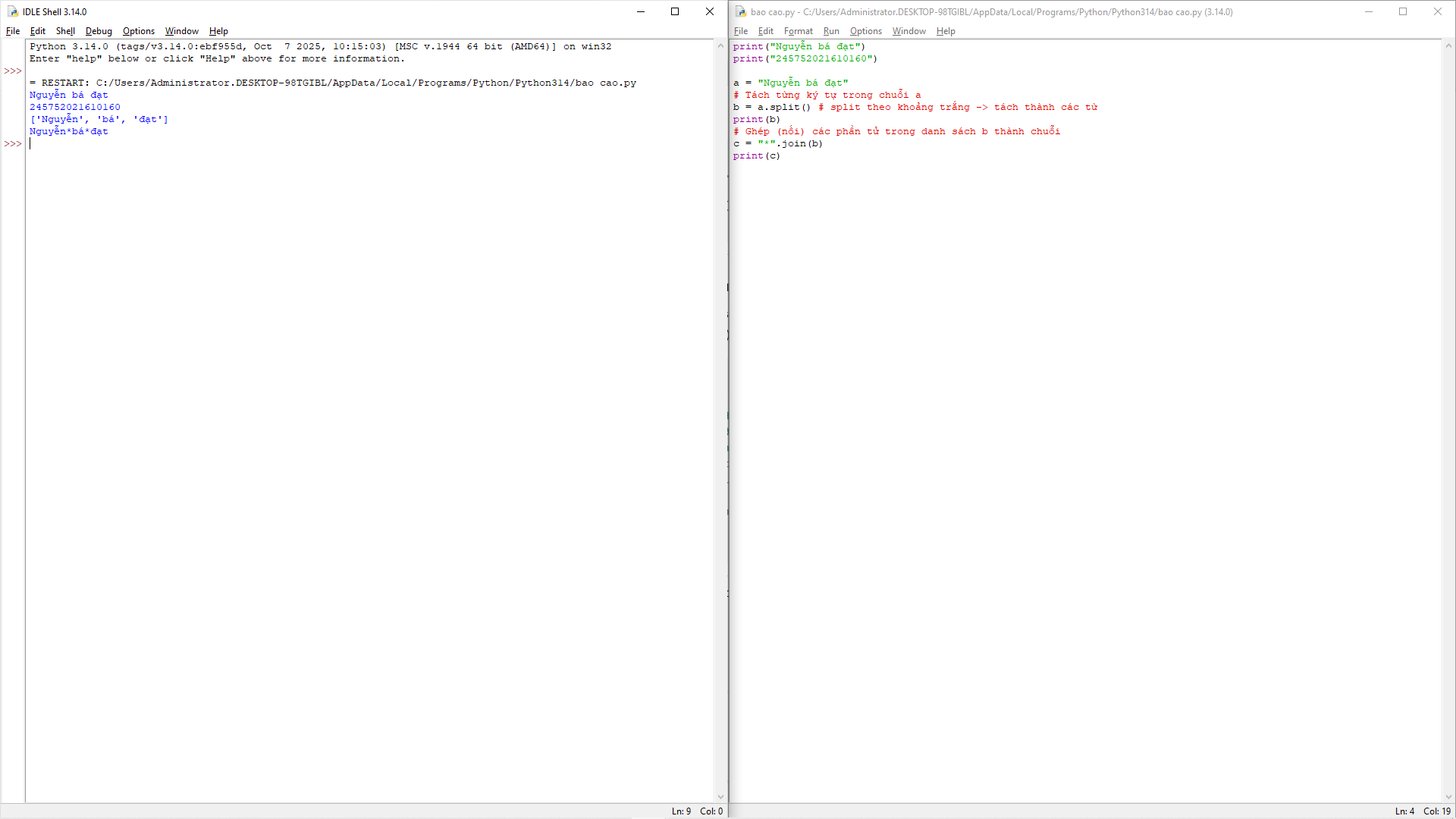
8. Viết chương trình in ra màn hình dãy số Fibonacci nhỏ hơn 4.000.000, tìm tổng các số chẵn trong dãy đã in:



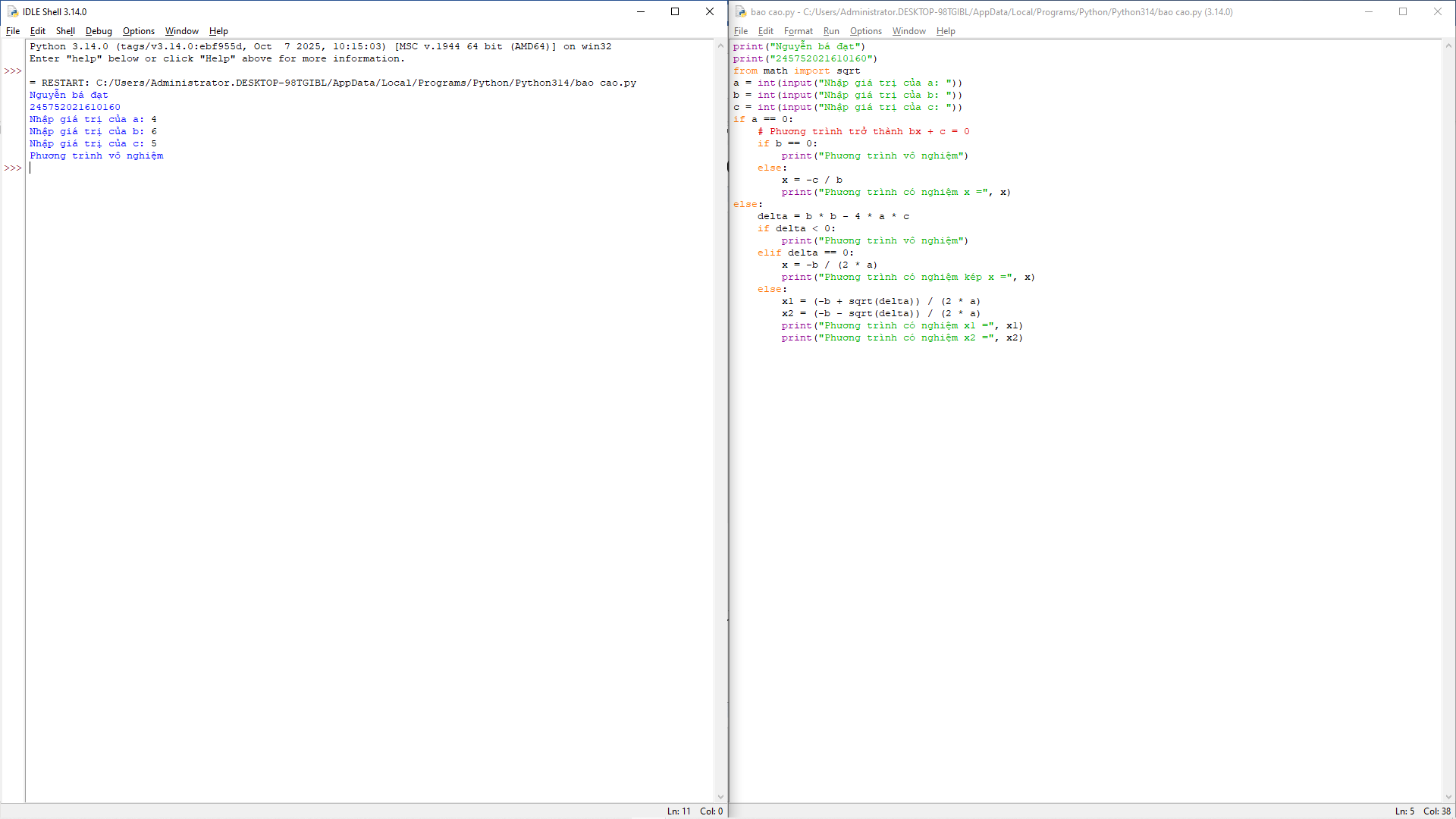
9. Viết chương trình đếm số ký tự trong 1 xâu ký tự nhập vào từ bàn phím, lưu các ký tự vào cấu trúc từ điển:



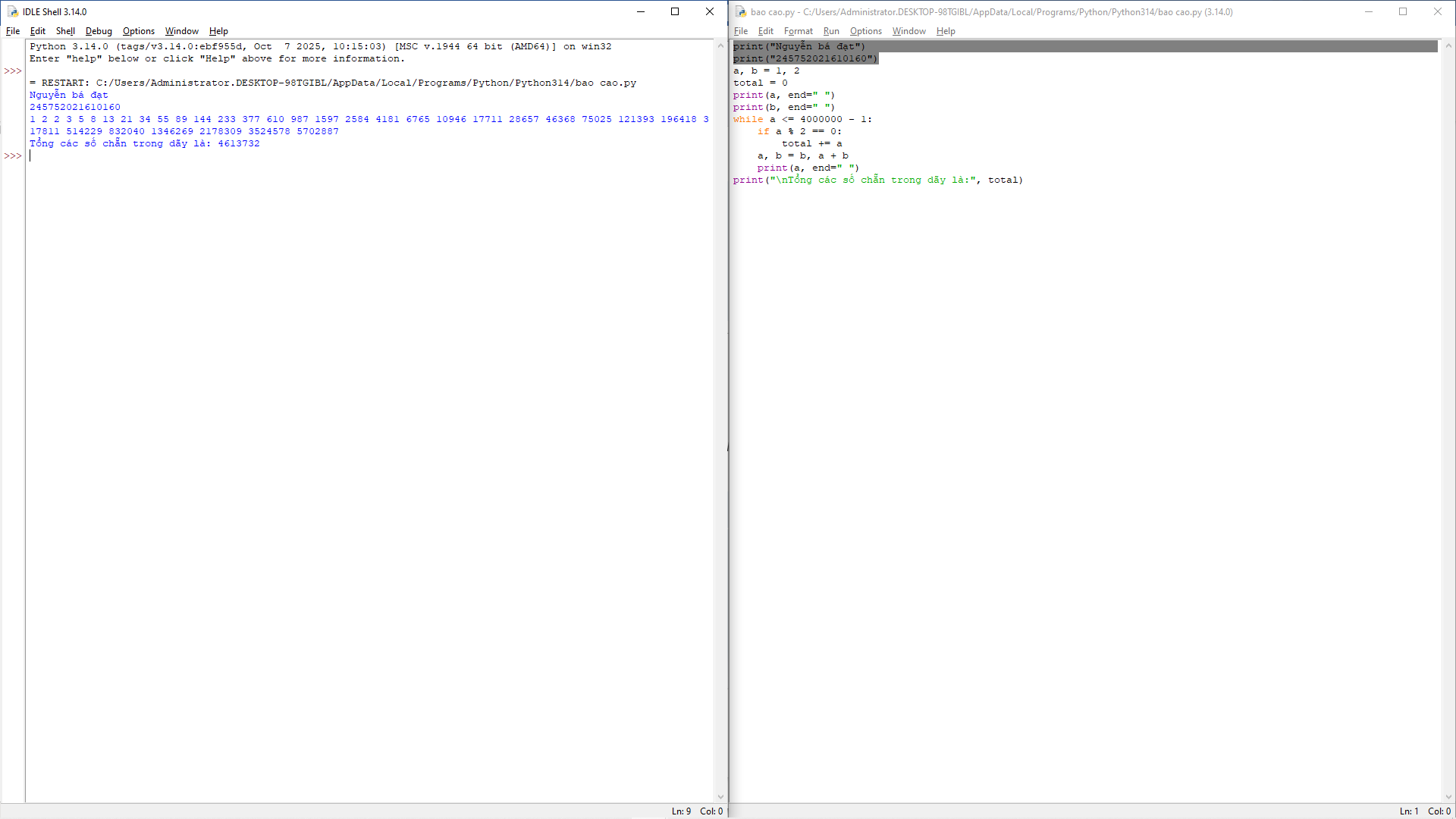
10.Viết chương trình sử dụng các phương thức split và join để tách nhập xâu ký tự:



11.Viết chương trình giải phương trình bậc 2: ax2 +bx+c=0, với các hệ số a, b, c nhập từ bàn phím:

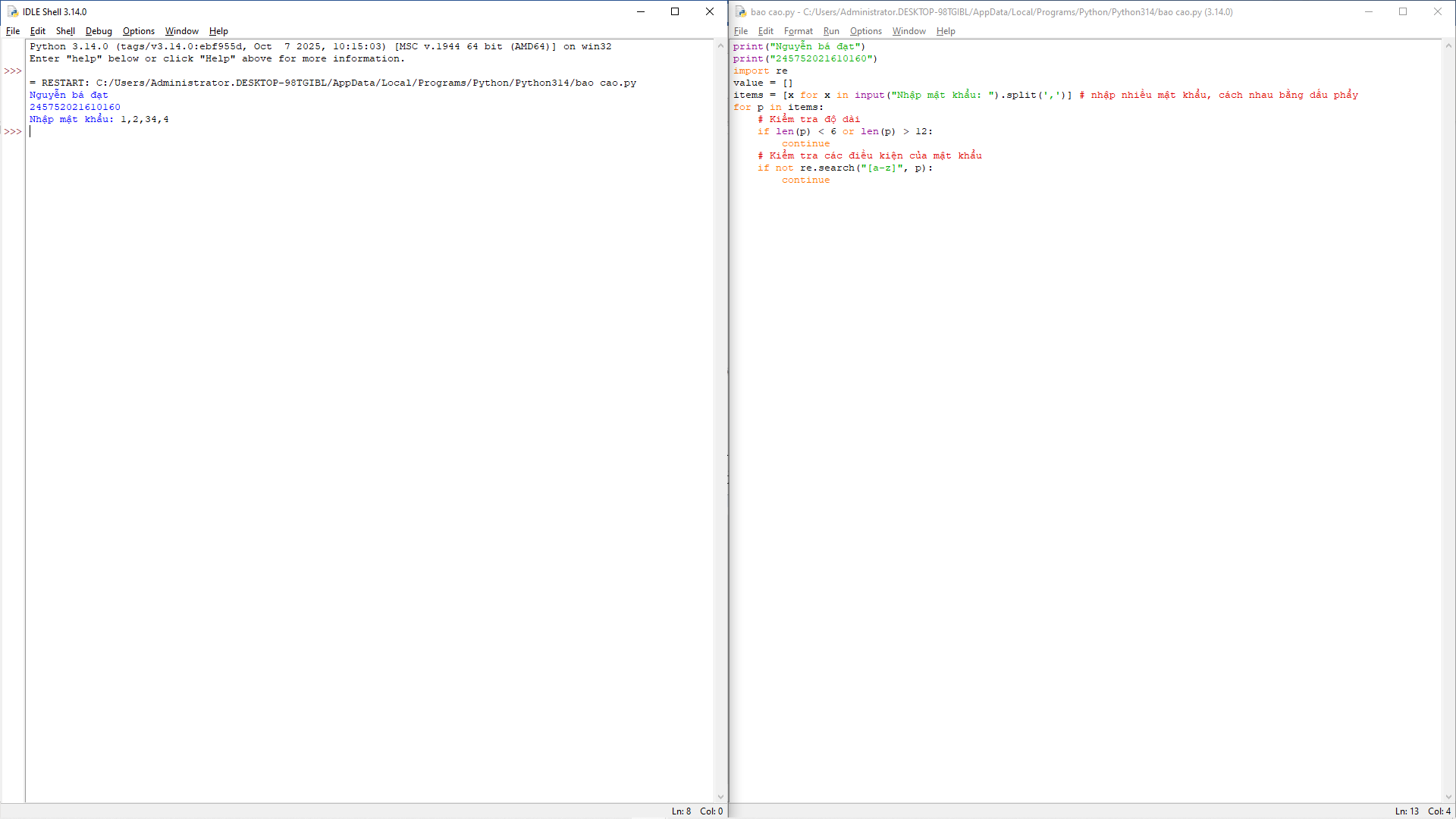


12. Viết chương trình in ra màn hình dãy số Fibonacci nhỏ hơn 4.000.000, tìm tổng các số chẵn trong dãy đã in:



13. Một website yêu cầu người dùng nhập tên người dùng và mật khẩu để đăng ký. Viết chương trình để kiểm tra tính hợp lệ của mật khẩu mà người dùng nhập vào.

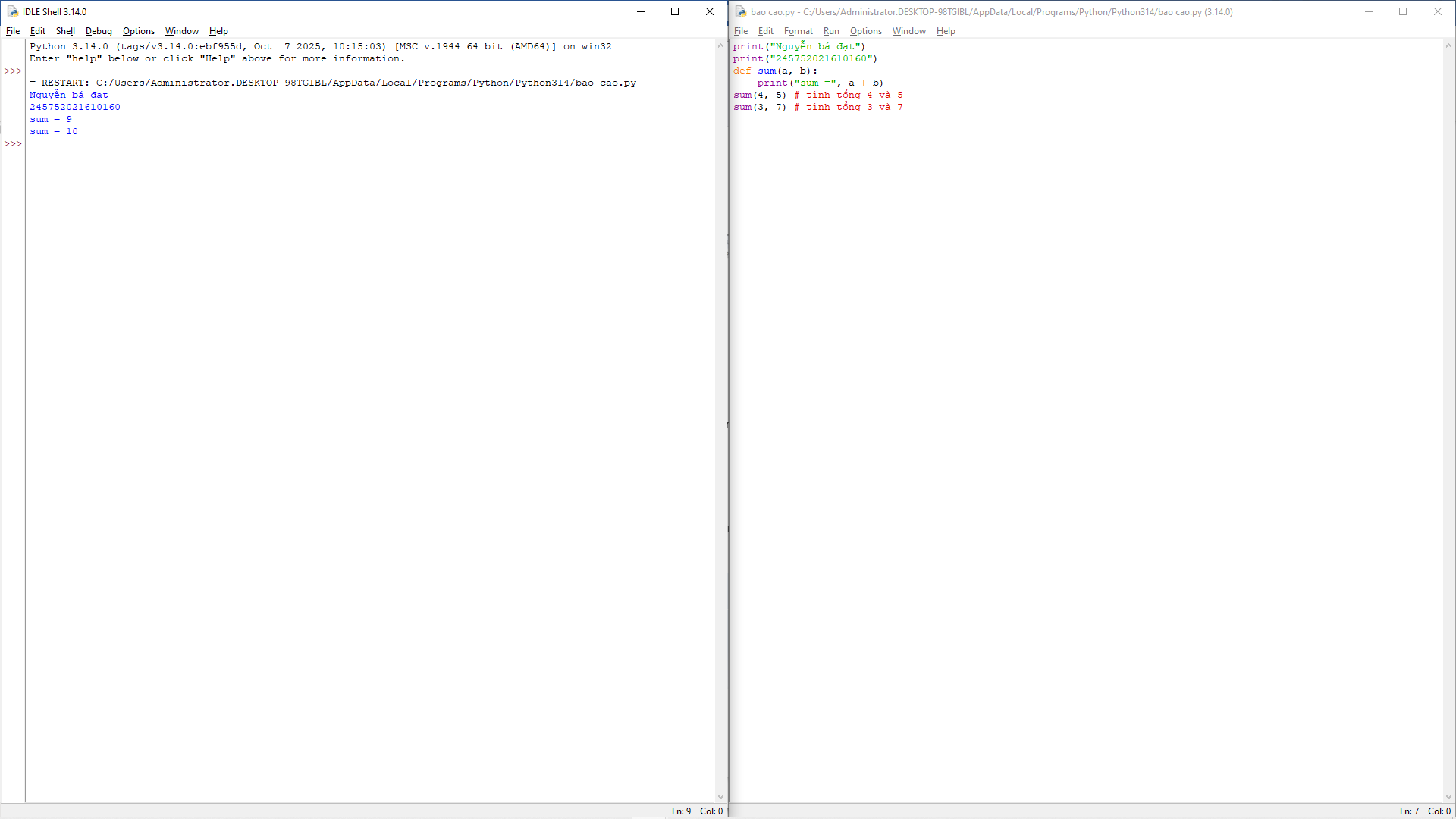
- Sử dụng các hàm có sẵn của Python như len() (kiểm tra độ dài), islower(), isupper(), isdigit() (kiểm tra loại ký tự).



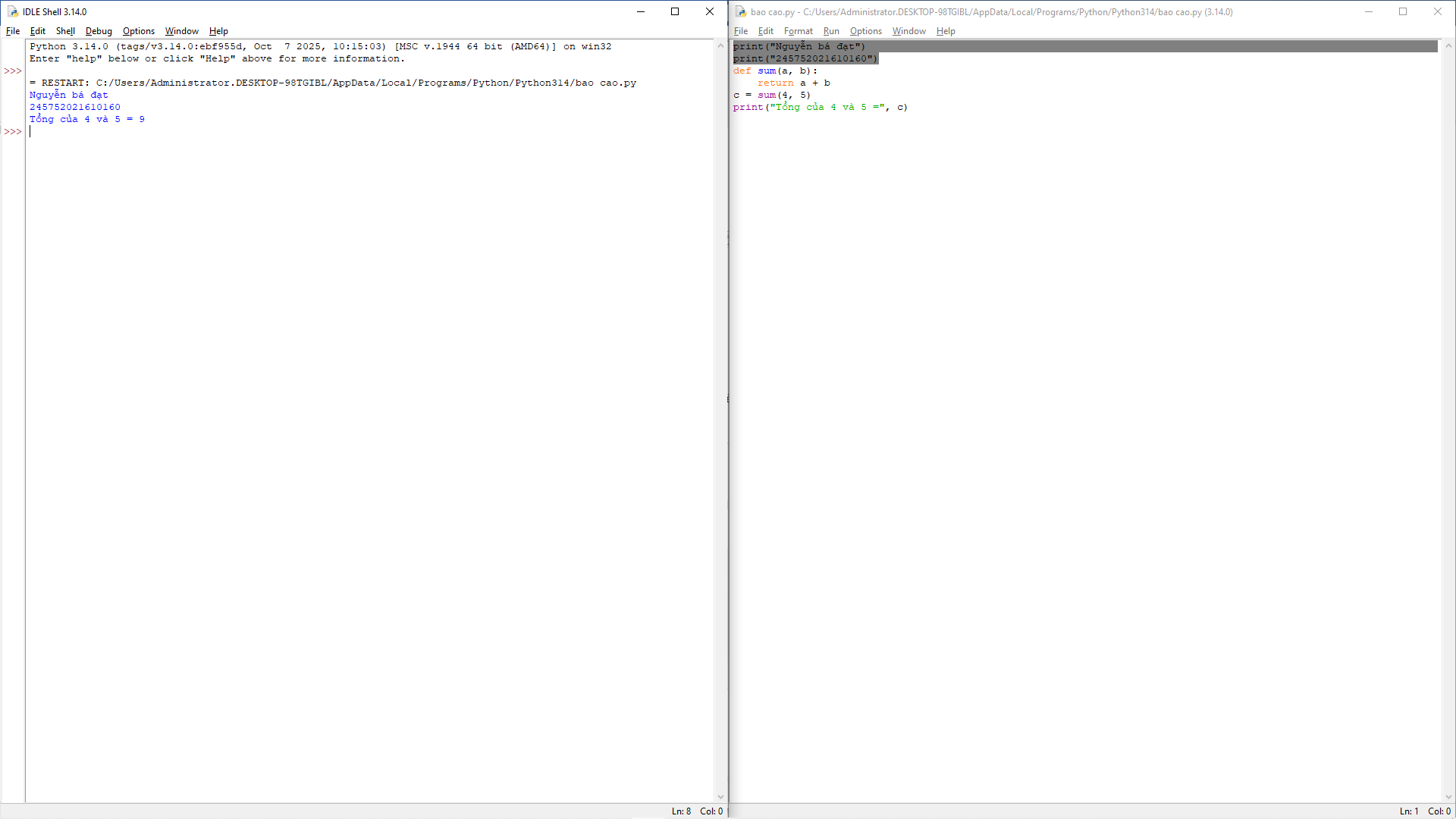
**Bài 3. Lập trình hàm trong Python**

1. **Mục đích:**
2. **Các bước thực hiện và kết quả:**

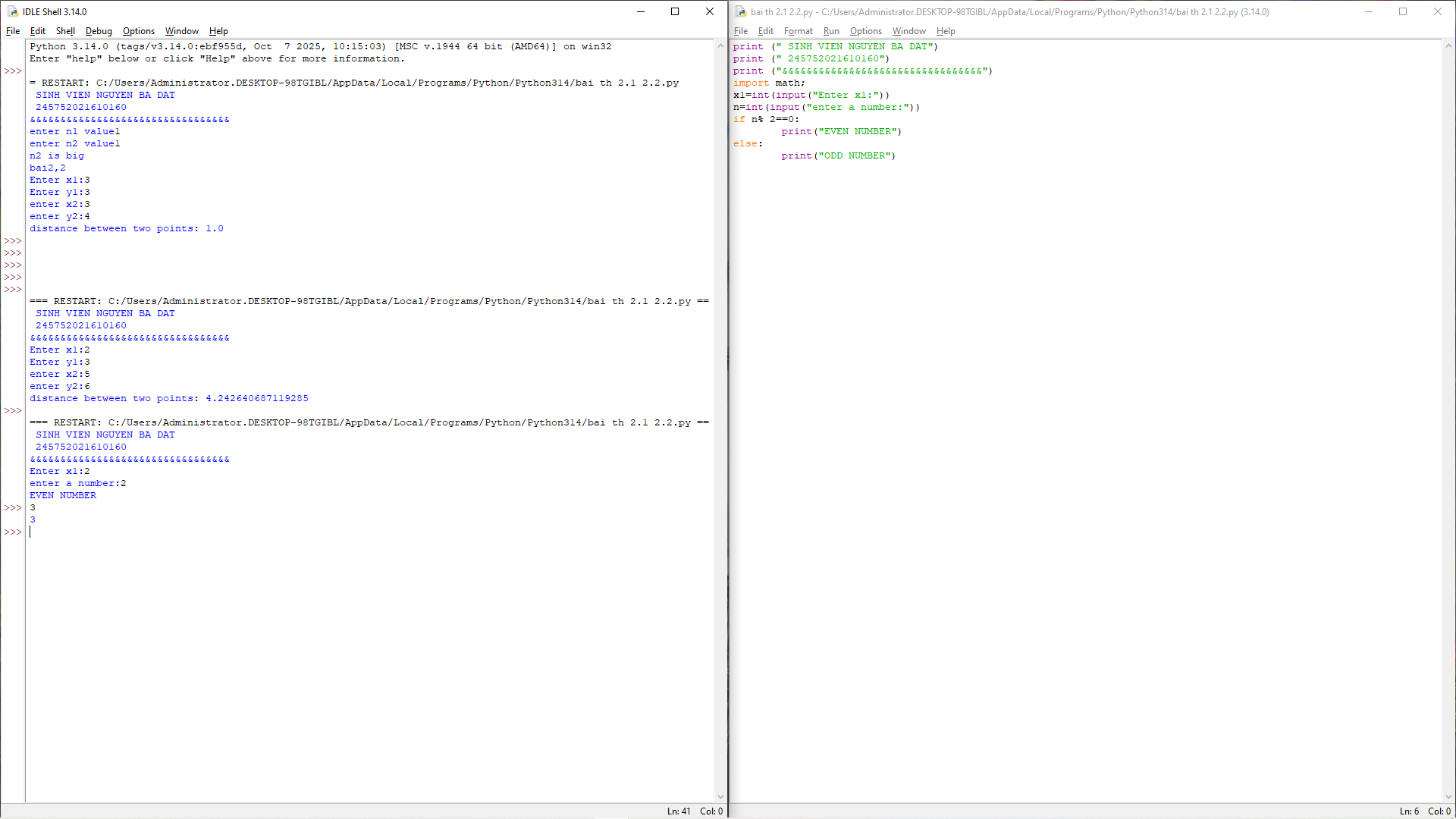
1. Viết hàm sum() tính tổng hai số:



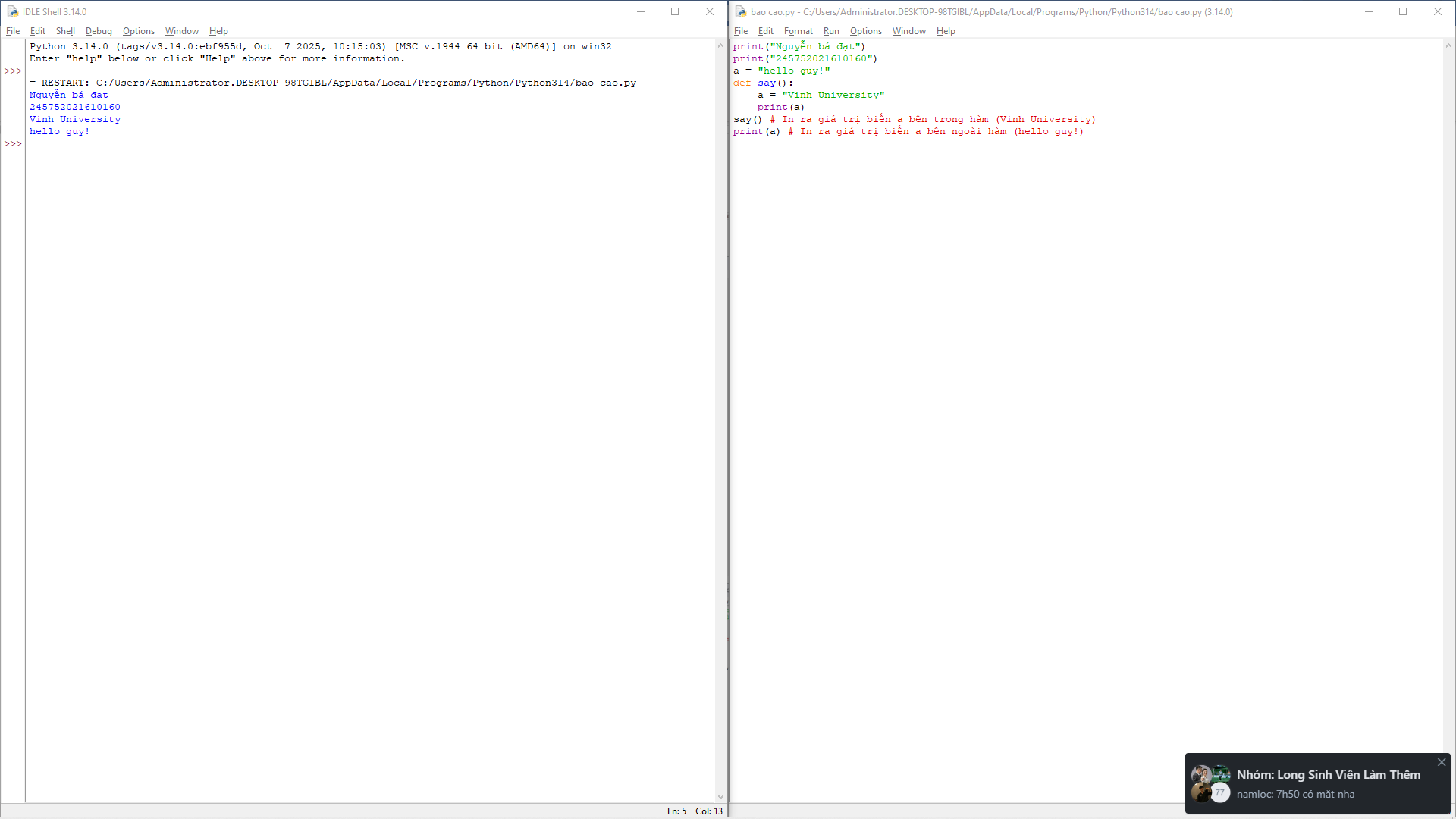
2. Viết hàm sum() với kết quả trả về:



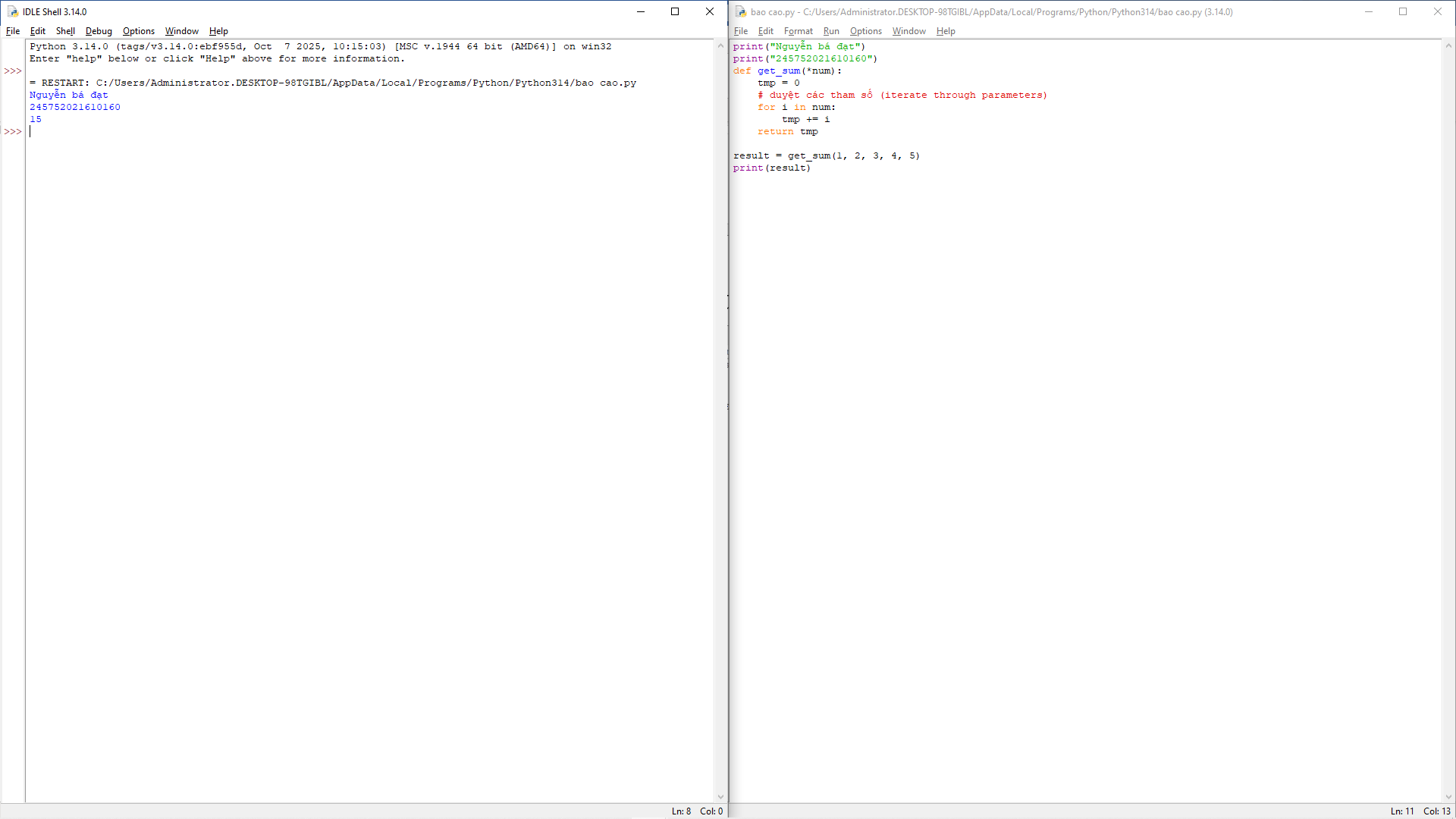
3. Tìm và sửa lỗi chương trình:



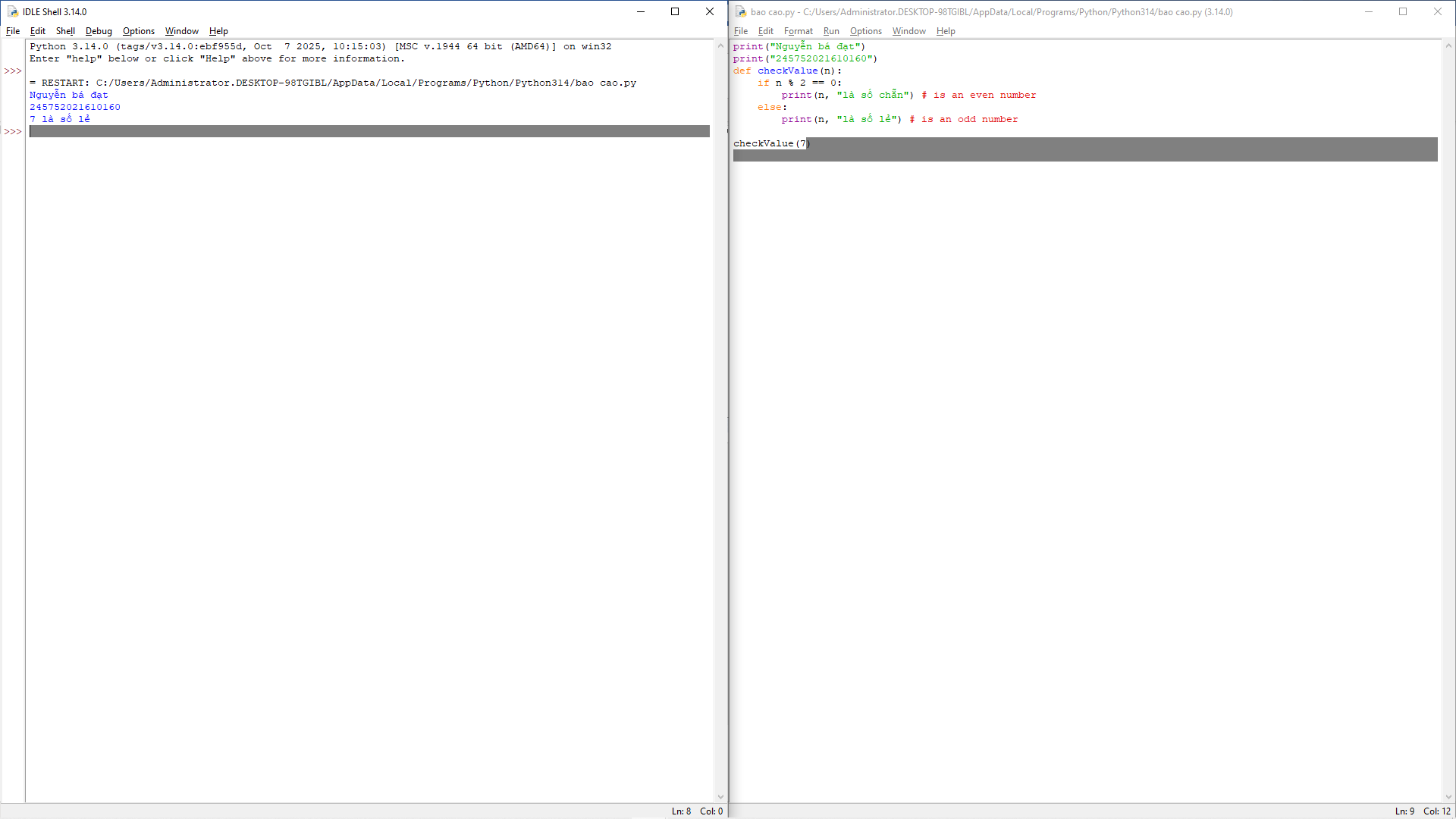
4. Viết chương trình có phạm vi biến như sau:



5. Viết chương trình sau và giải thích việc truyền tham số của hàm:



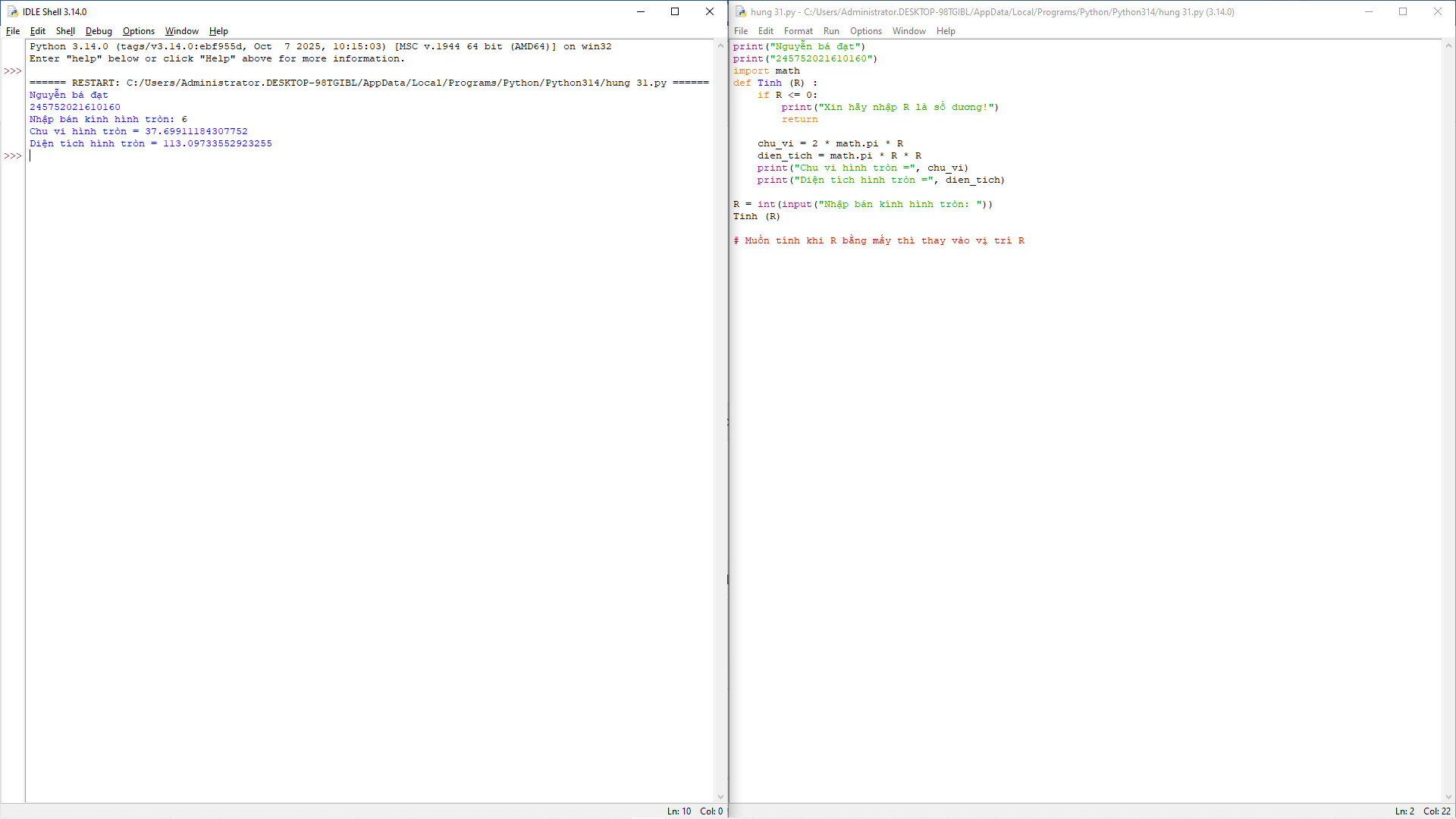
6. Định nghĩa hàm có thể chấp nhận input là số nguyên và in "Đây là một số chẵn" nếu nó chẵn và in "Đây là một số lẻ" nếu là số lẻ.



7. Một Robot di chuyển trong mặt phẳng bắt đầu từ điểm đầu tiên (0,0). Robot có thể di chuyển theo hướng UP, DOWN, LEFT và RIGHT với những bước nhất định. Dấu di chuyển của robot được đánh hiển thị như sau:



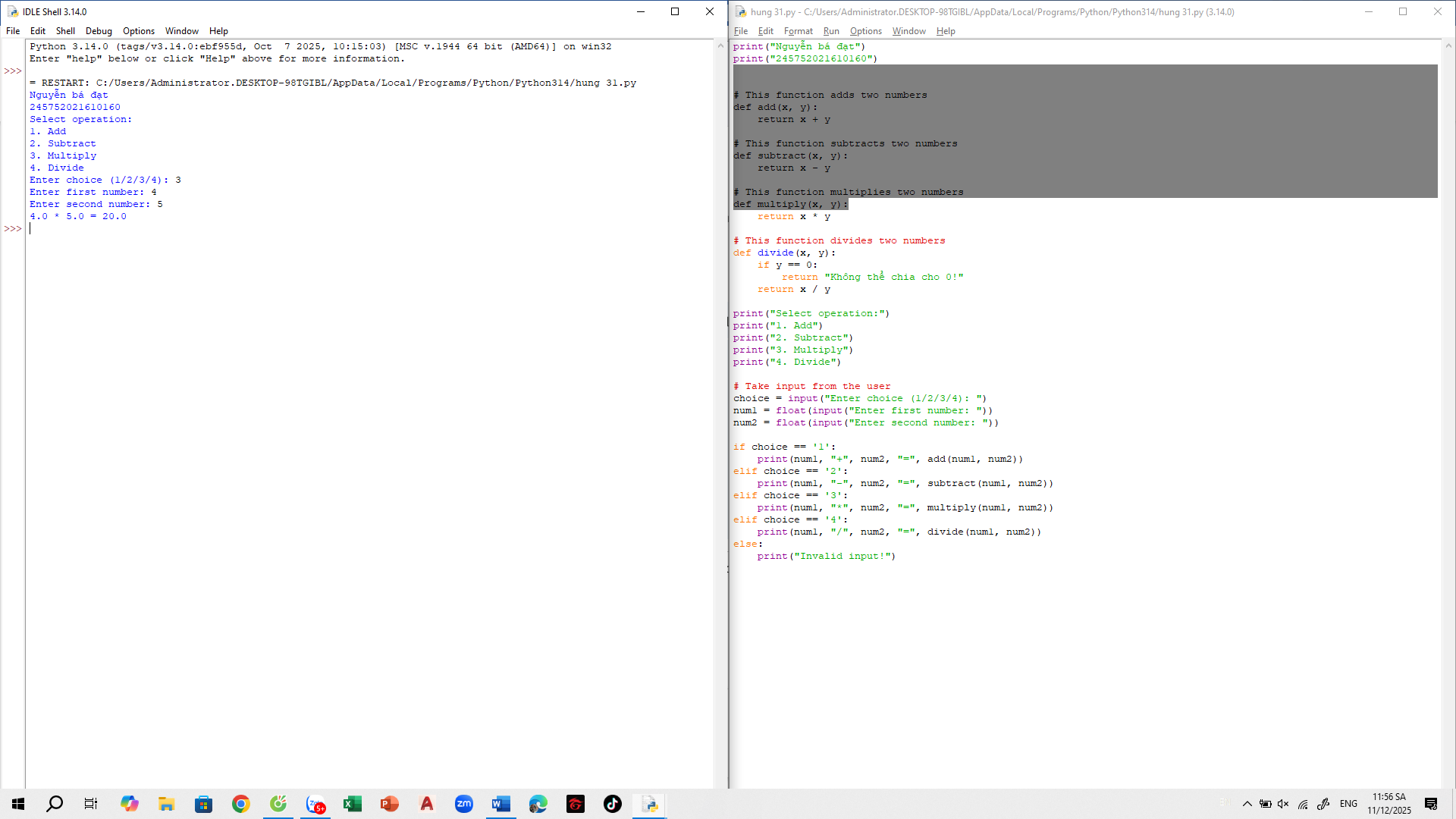
8. Một Robot di chuyển trong mặt phẳng bắt đầu từ điểm đầu tiên (0,0). Robot có thể di chuyển theo hướng UP, DOWN, LEFT và RIGHT với những bước nhất định. Dấu di chuyển của robot được đánh hiển thị như sau:

****

9. Biết lãi suất tiết kiệm là t%/tháng (nhập t từ bàn phím). Nhập số vốn ban đầu n và số tháng gửi k. Tính số tiền nhận được sau k tháng:

****

10. Chương trình máy tính thực hiện các phép tính đơn giản:

****

**Bài 4. Các kiểu dữ liệu có cấu trúc trong Python**

**1. Mục đích:**

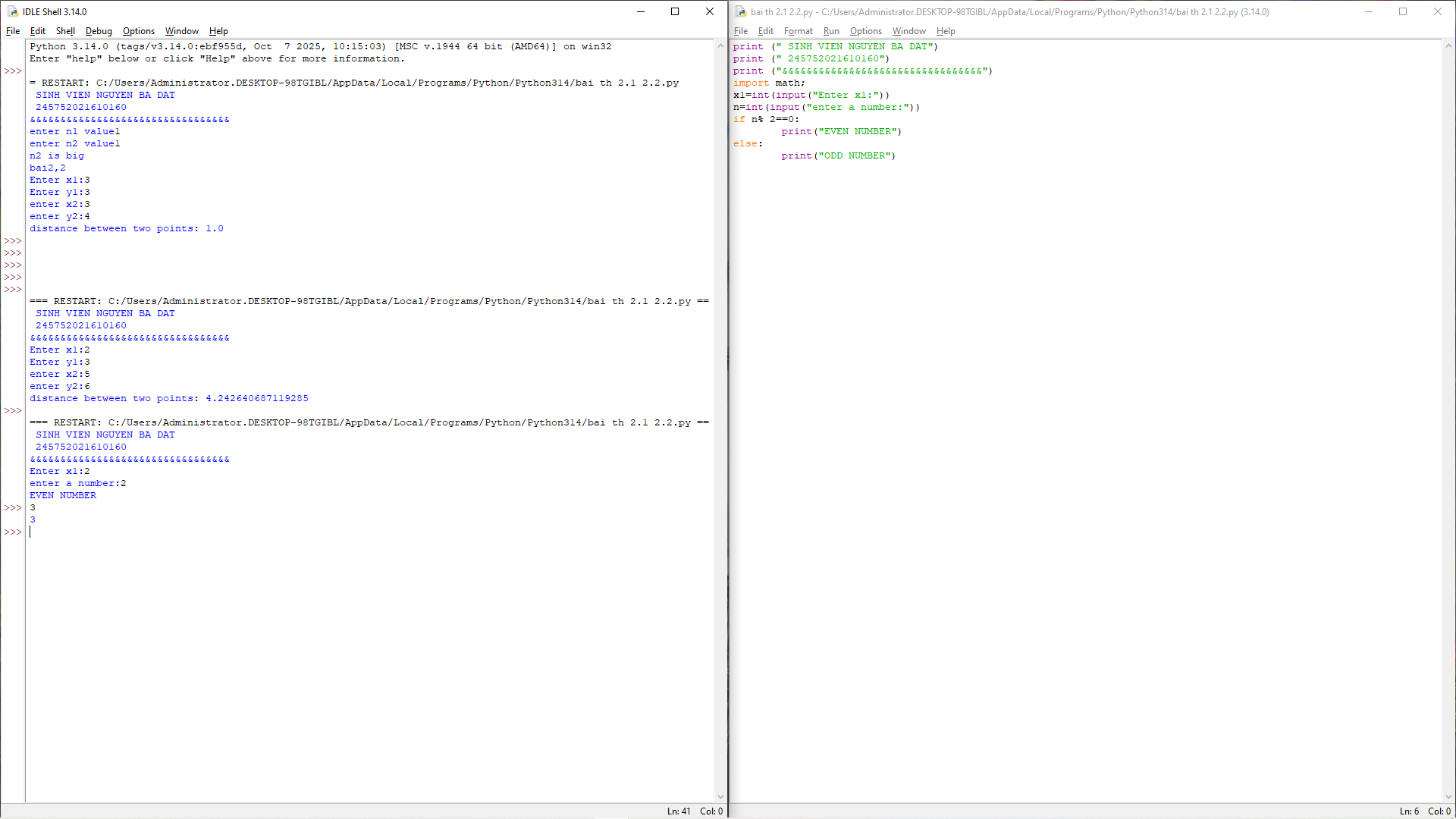
- Hiểu về Giải thuật: Giúp bạn hiểu hai trong số những giải thuật tìm kiếm cơ bản và quan trọng nhất.

- Module hóa: Rèn luyện kỹ năng tổ chức code thành các module (các file .py riêng biệt) để tái sử dụng và quản lý code dễ dàng hơn.

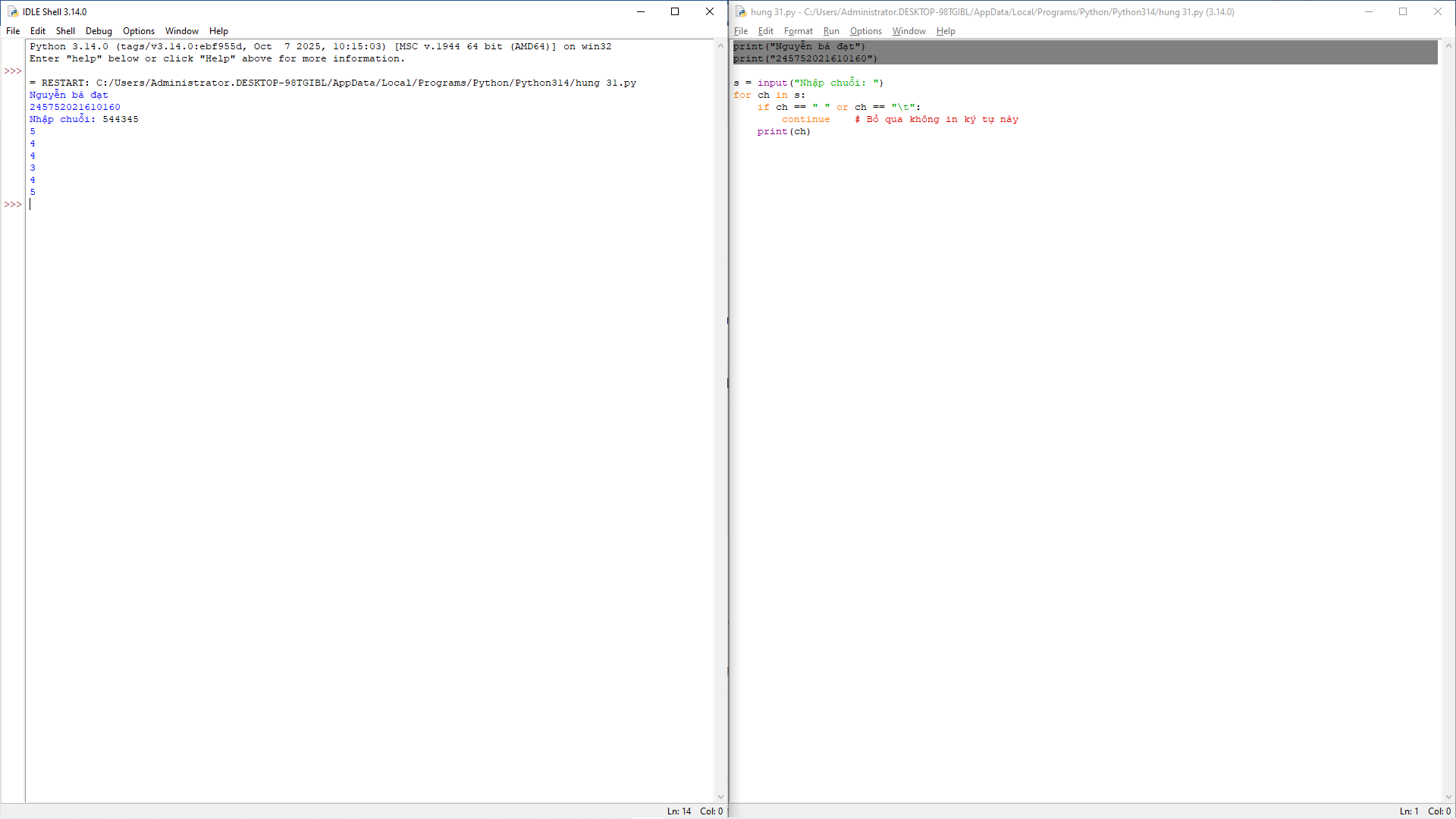
- Phân tích Hiệu suất: Thấy được sự khác biệt lớn về tốc độ giữa hai phương pháp.

**2. Các bước thực hiện và kết quả:**

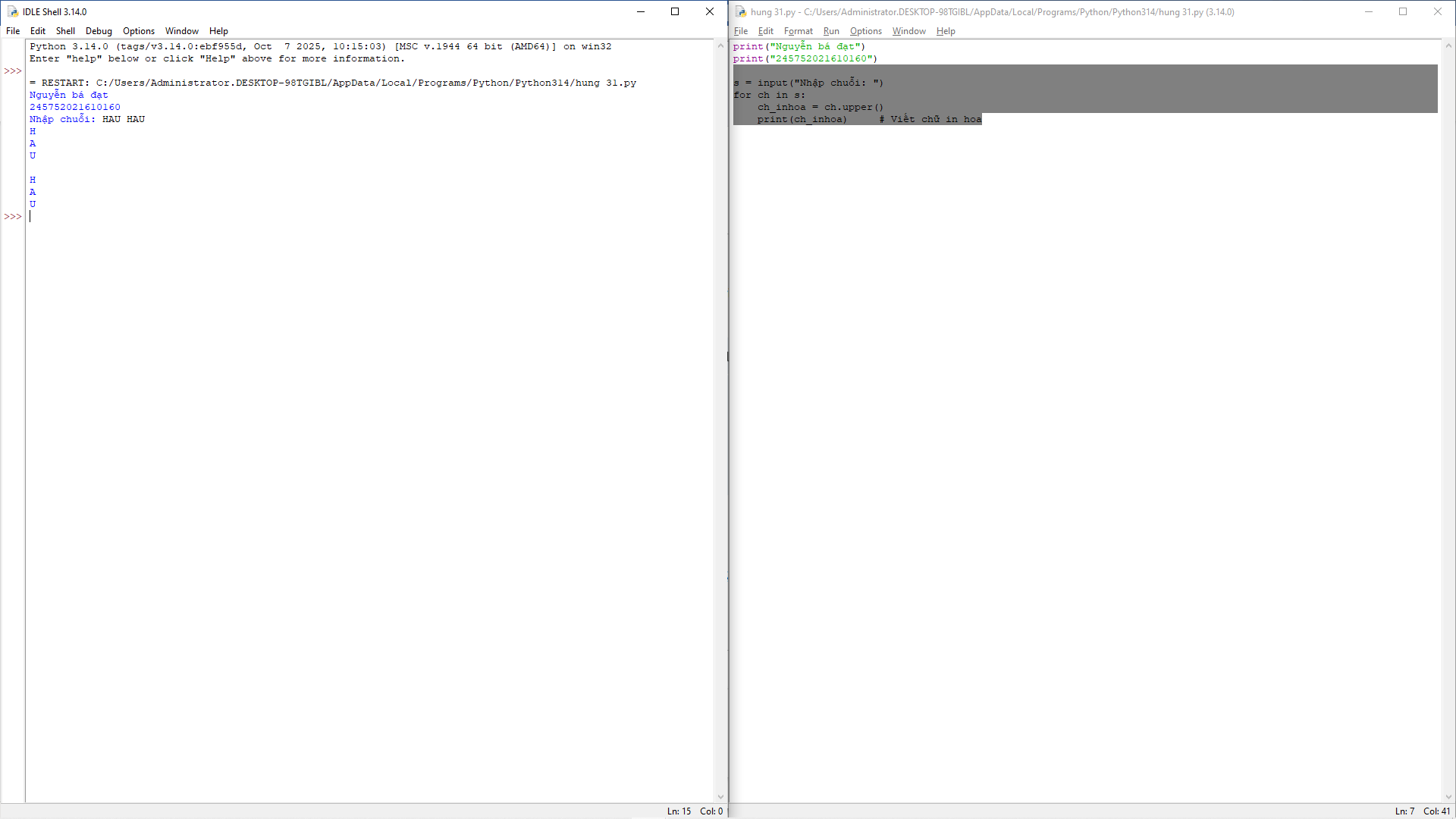
1. Nhập chuỗi S và in ra từng kí tự của S, mỗi kí tự trên một dòng.

****

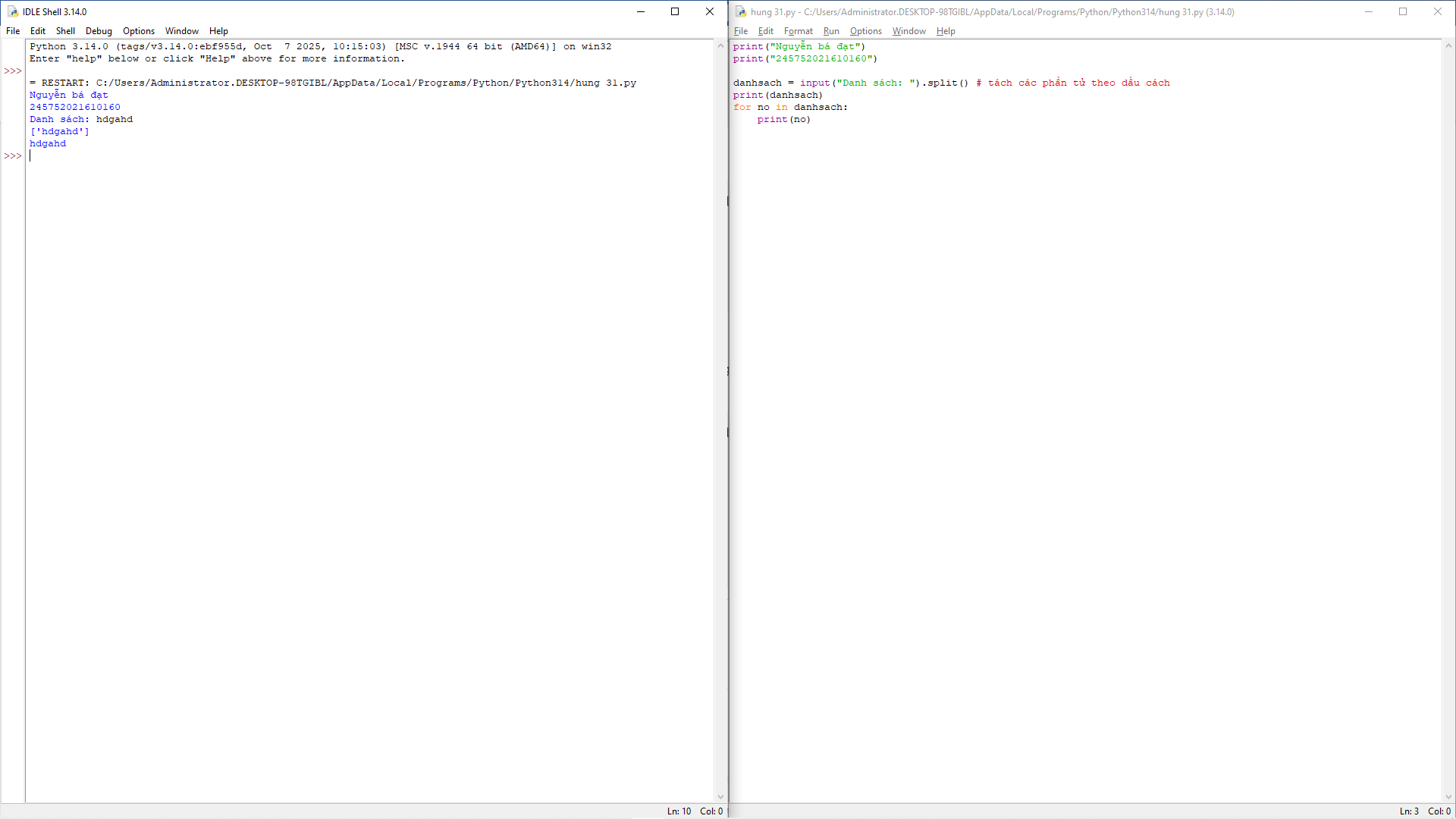
2. Chỉnh sửa ví dụ trên: hãy bỏ qua không in ra những kí tự “không nhìn thấy” (dấu space và dấu tab).

****

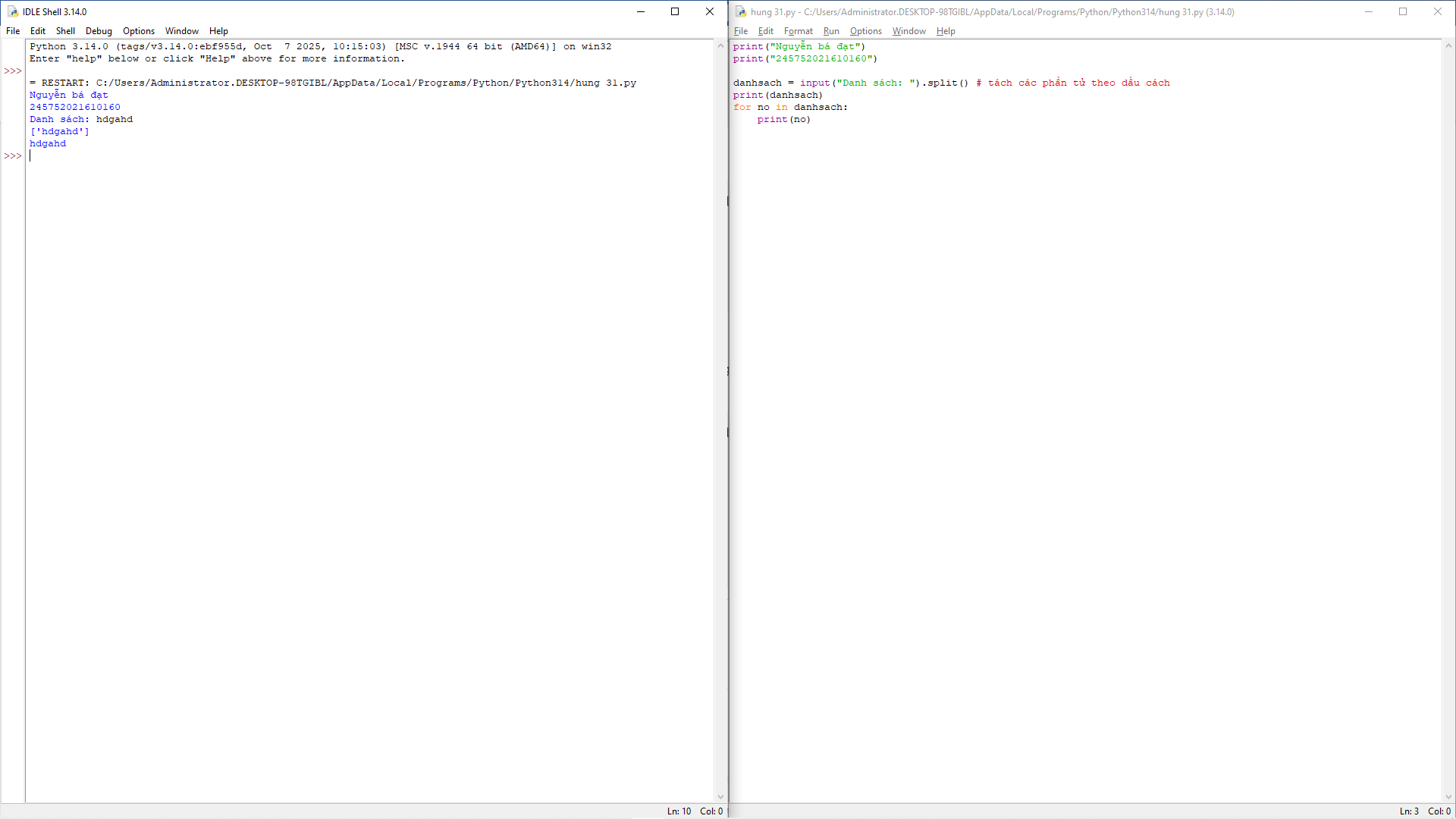
3. Chỉnh sửa ví dụ ở bài 1: hãy các kí tự ở dạng IN HOA.

****

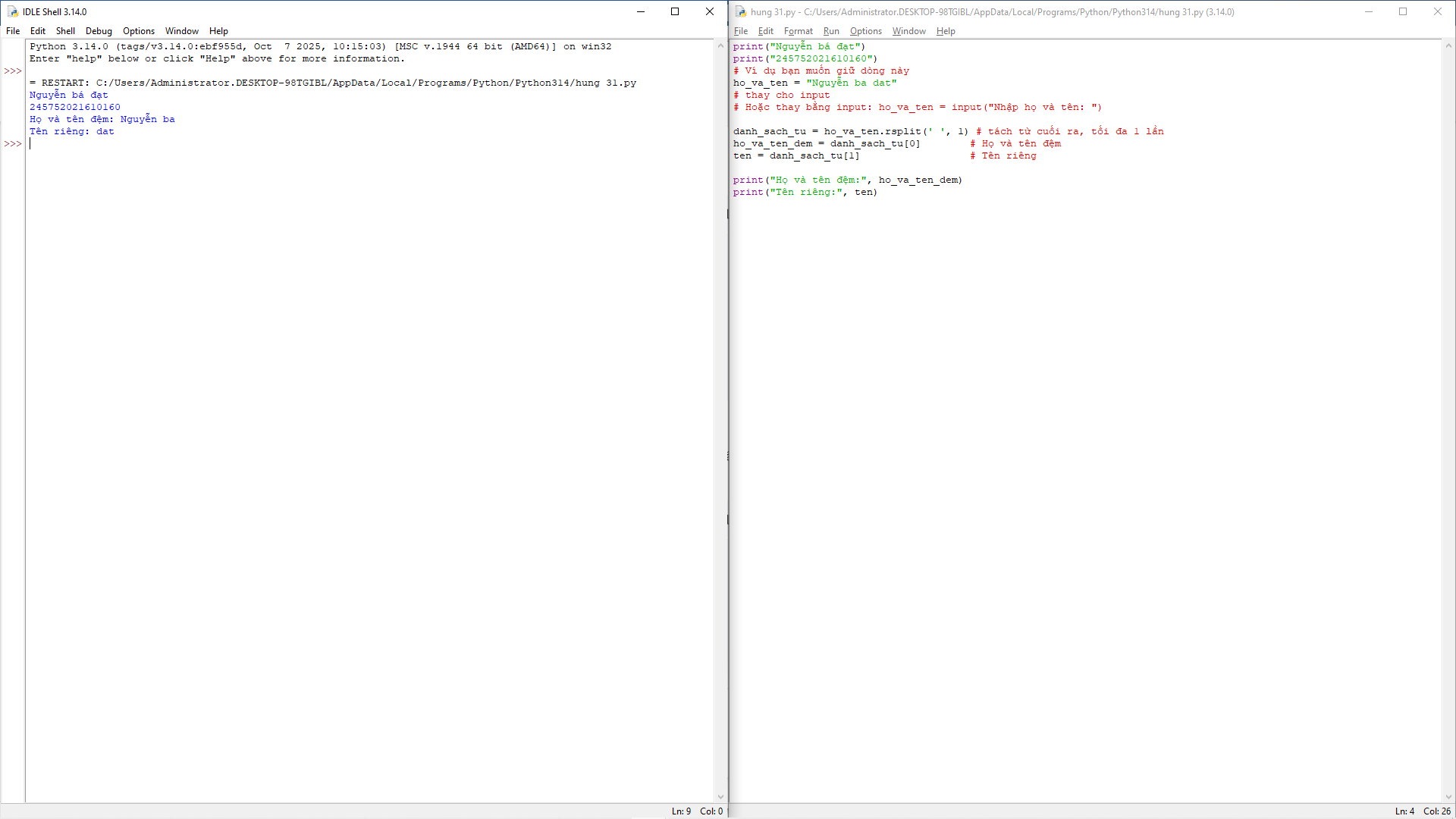
4. Nhập một danh sách trên một dòng, mỗi phần tử cách nhau bởi dấu trống hoặc tab, sau đó in ra dãy vừa nhập ra màn hình

****

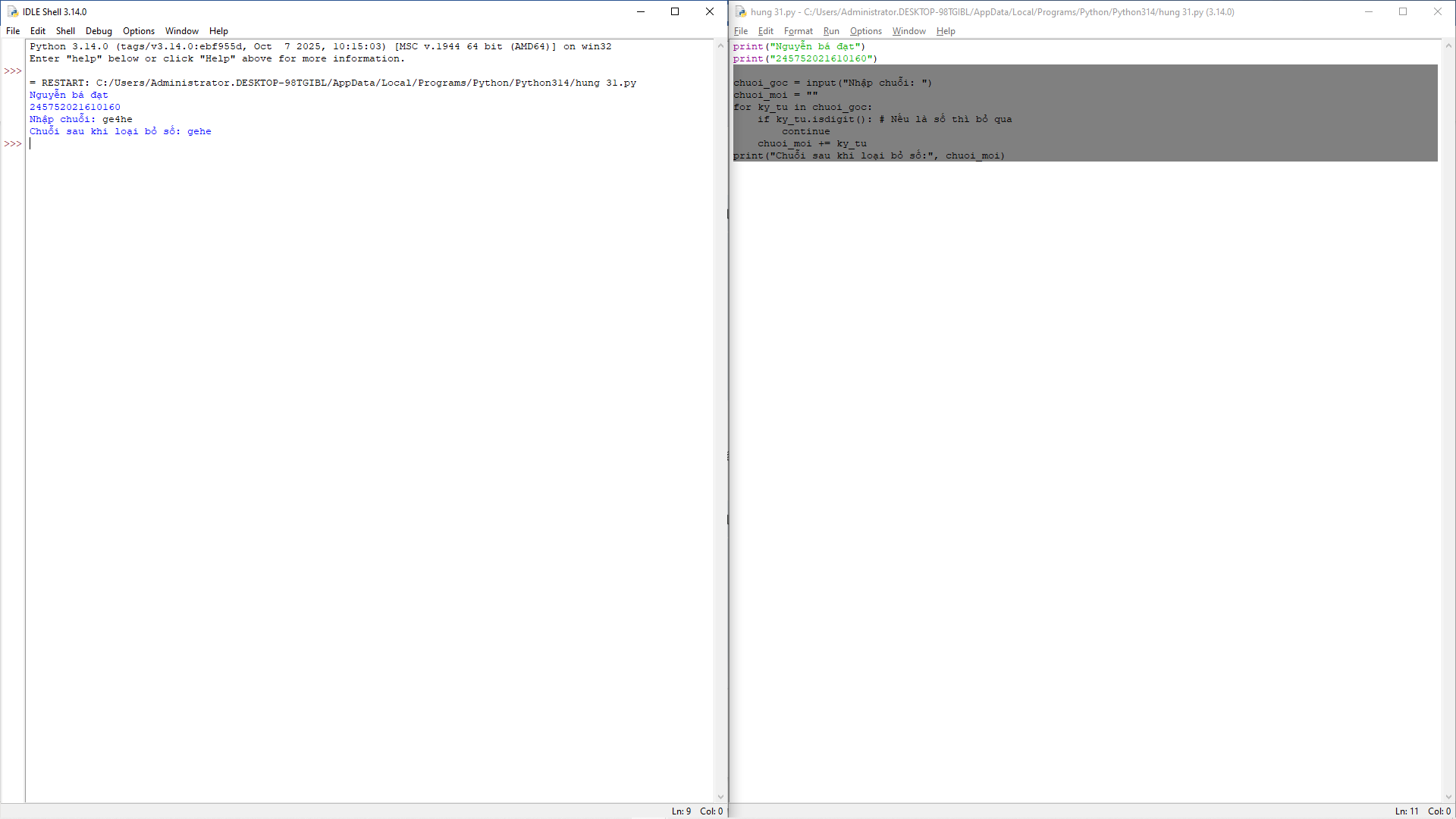
5. Chỉnh sửa ví dụ ở bài 4: nhập 1 danh sách các từ từ bàn phím, in ra các từ đó theo thứ tự ngược lại thứ tự vừa nhập (ví dụ nhập “mot hai ba” thì in ra “ba hai mot”)

****

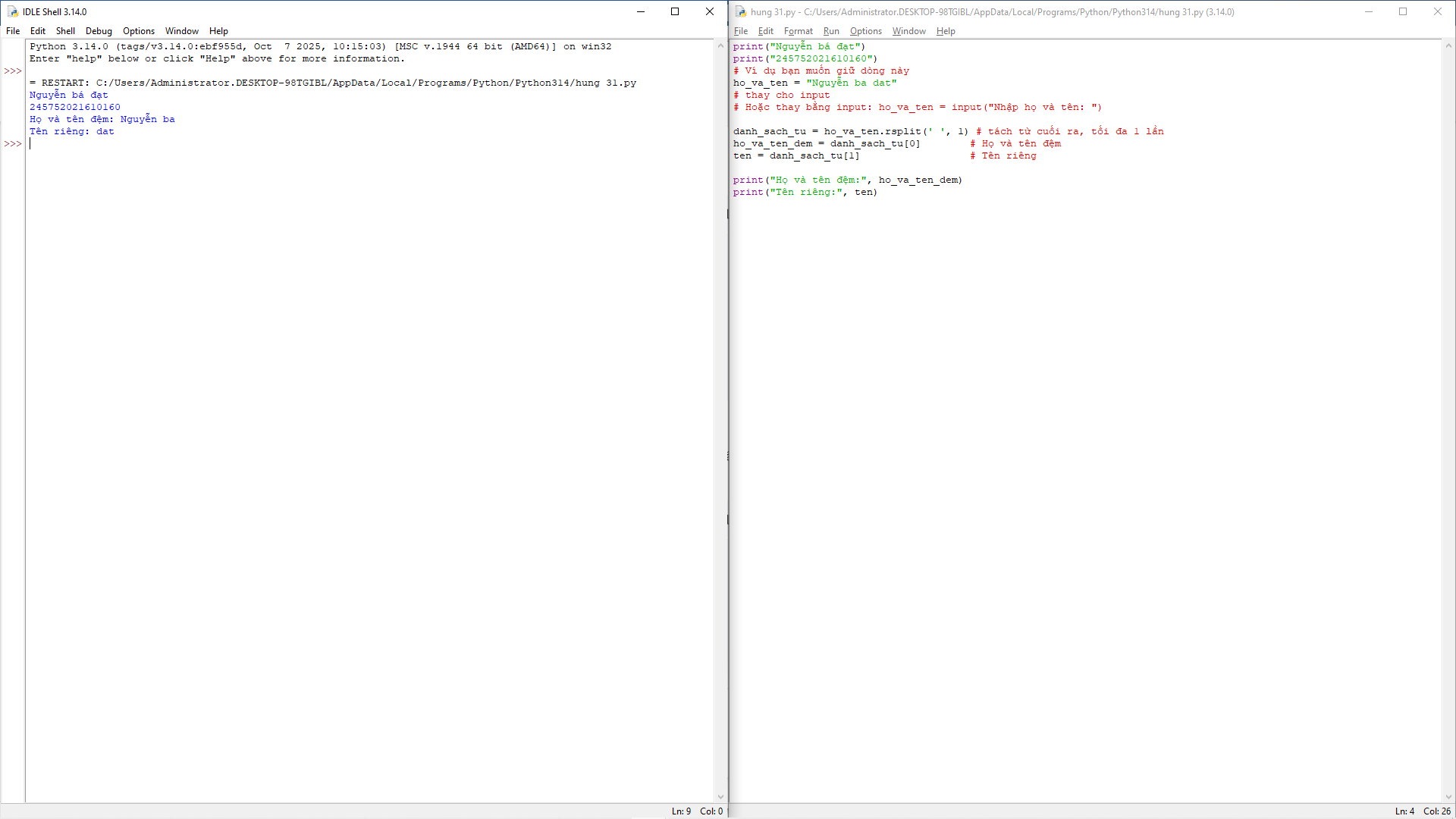
6. Nhập một tên người từ bàn phím, hãy tách phần họ và tên riêng của người đó và in chúng ra màn hình (giả thiết họ và tên riêng chỉ gồm một âm).

****

7. Nhập một chuỗi từ bàn phím, hãy loại bỏ tất cả các chữ số khỏi chuỗi và in lại nội dung chuỗi mới ra màn hình.

****

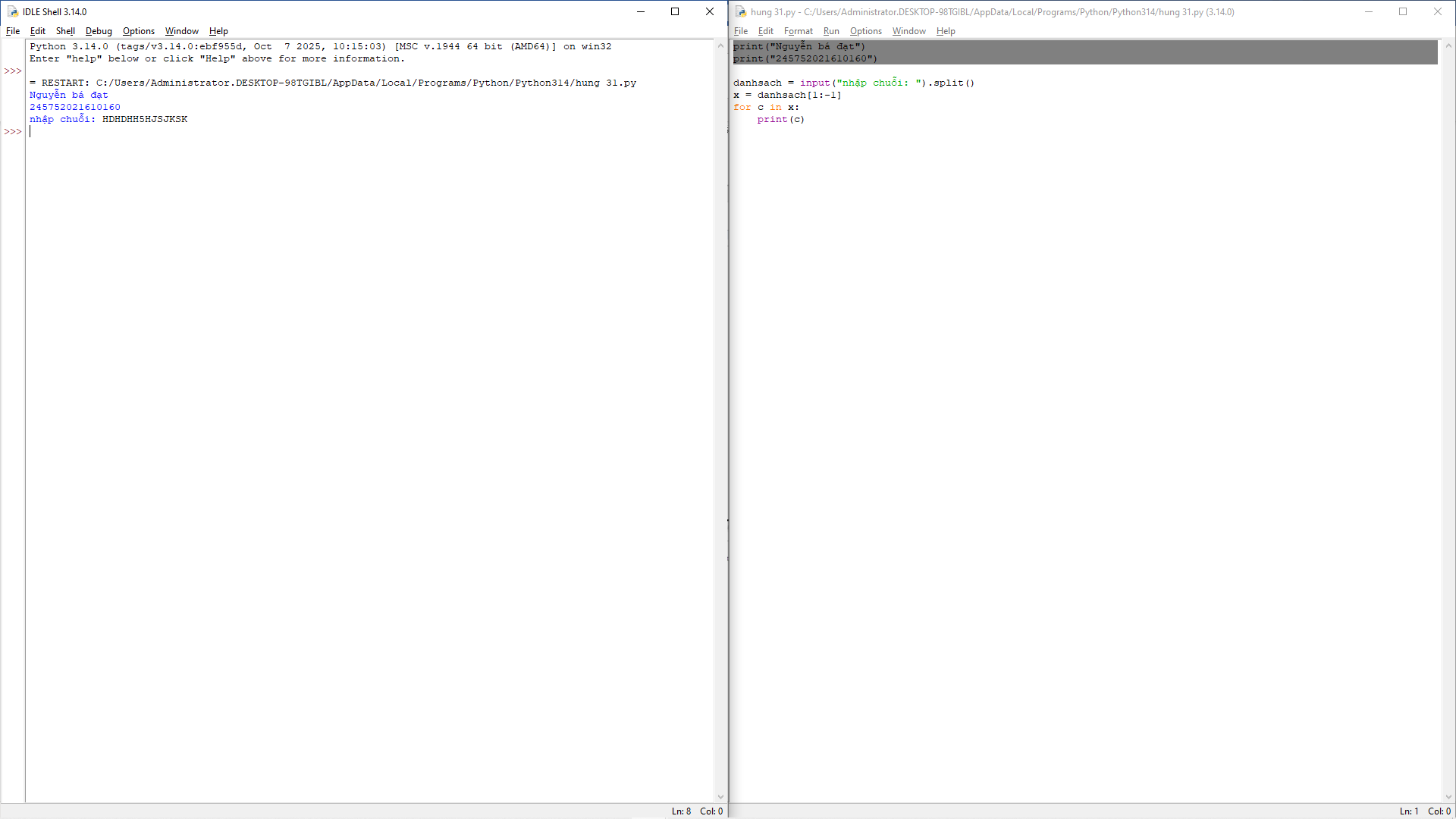
8. Nhập một dãy các từ từ bàn phím, hãy in ra từ dài nhất trong dãy vừa nhập, in ra mọi từ có cùng độ dài nhất.

****

9. Nhập một list từ bàn phím.

****

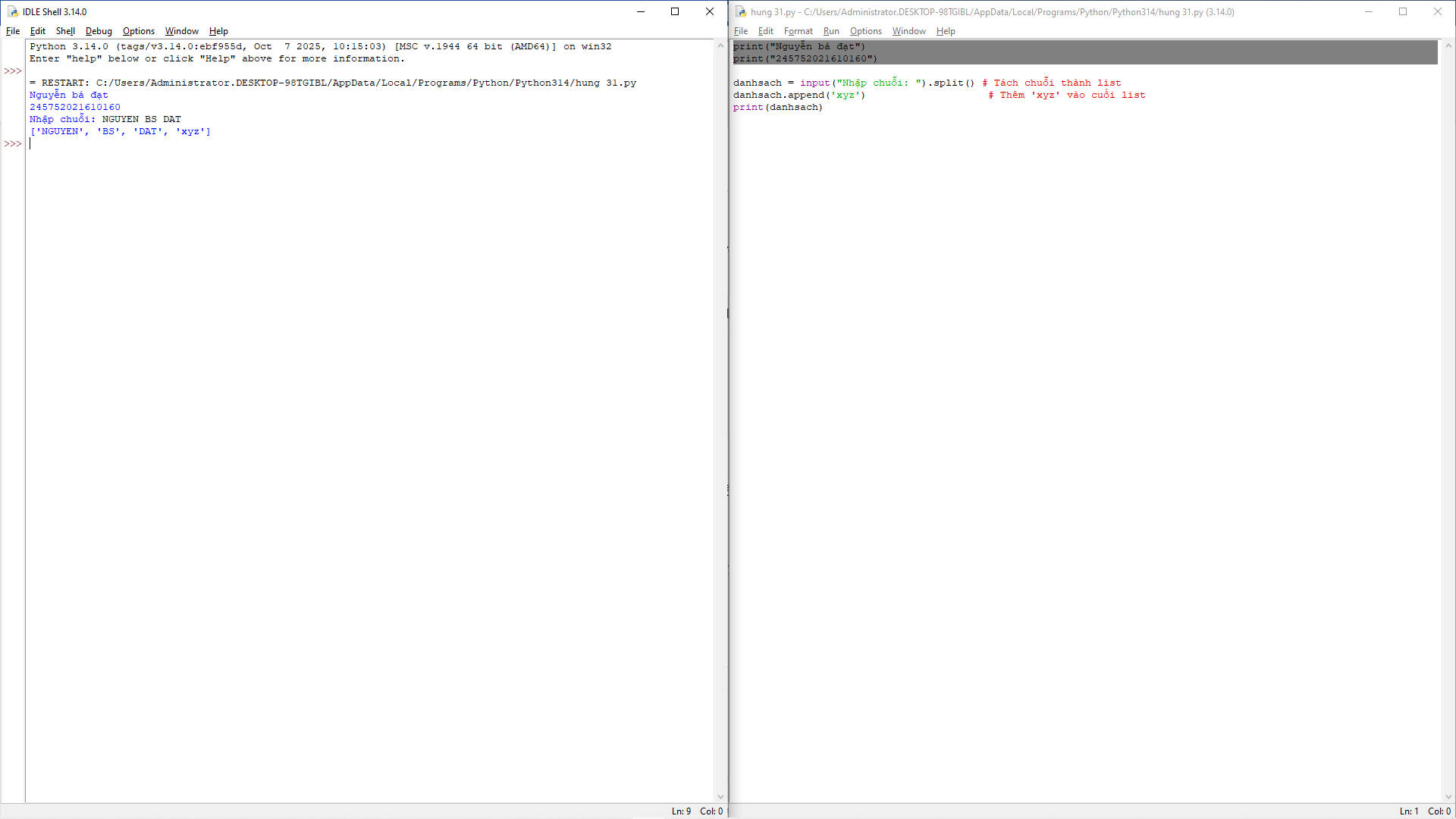
10.Cắt list: lấy list nhưng bỏ phần tử đầu và cuối.

****

11.Thêm phần tử vào list.

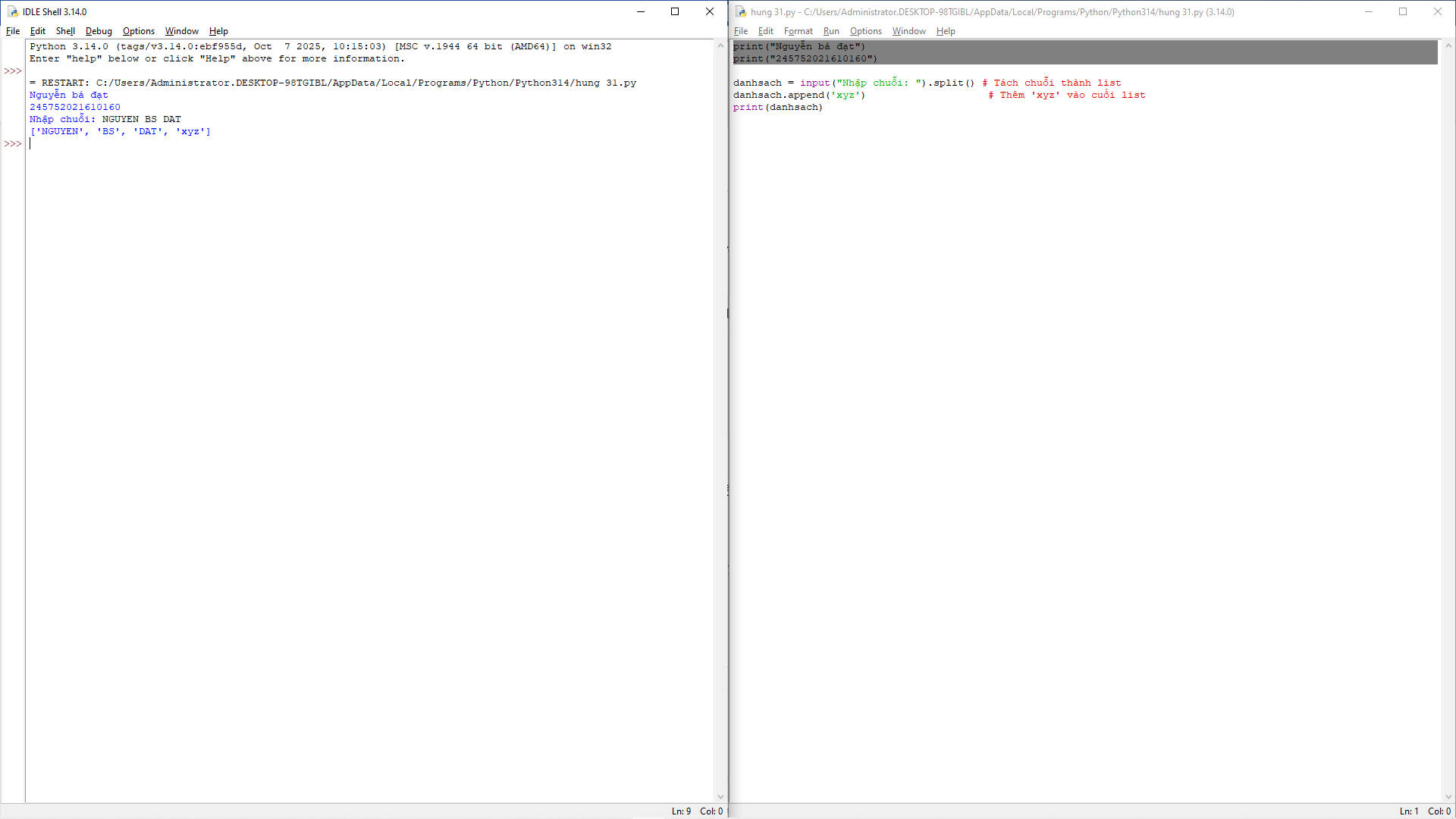
- Dùng lệnh append() để thêm phần tử vào list.

- Lệnh split(): dùng để cắt một chuỗi thành các phần tử nhỏ.



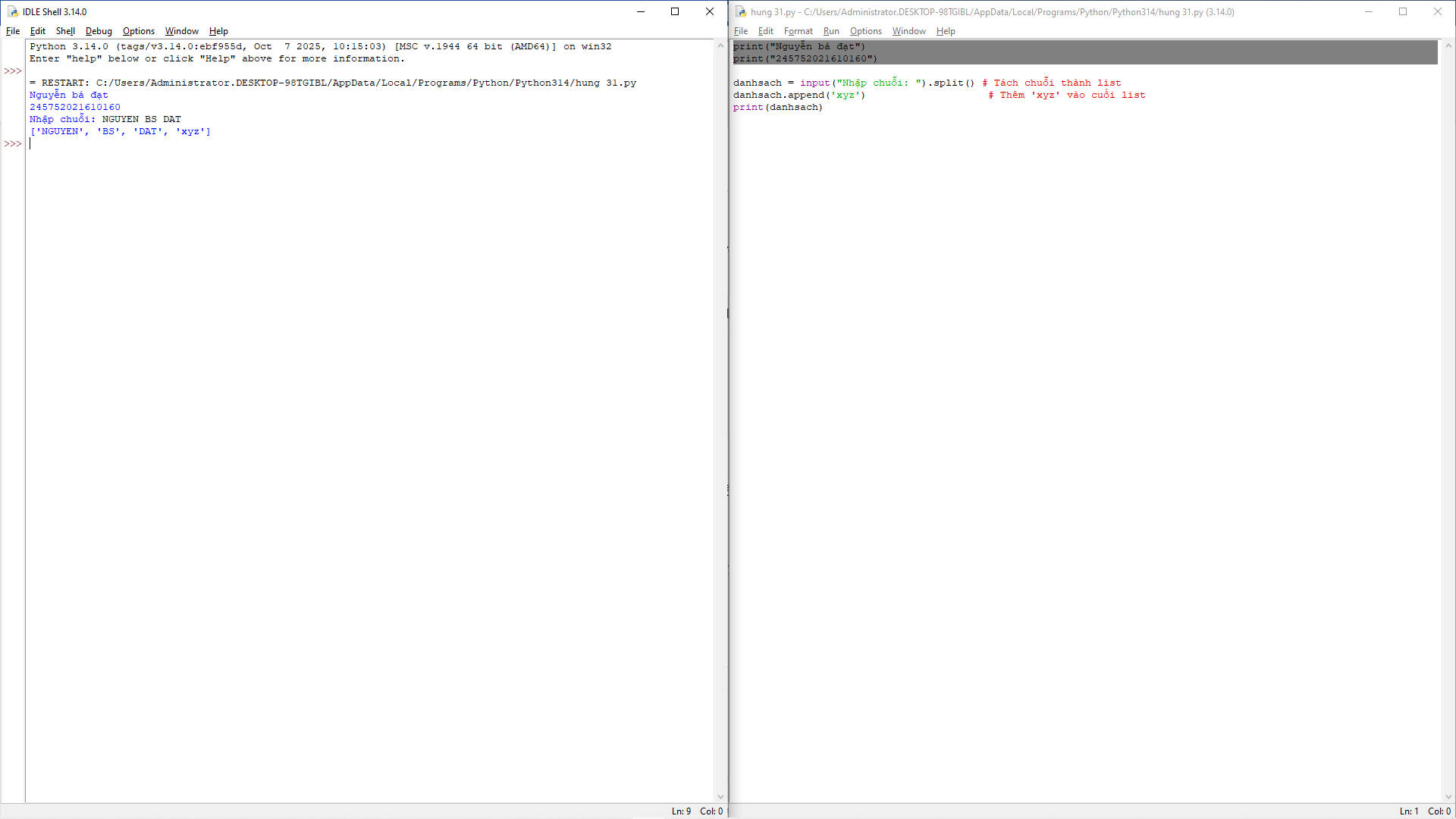
12.Bỏ phần tử khỏi list.

- Dùng lệnh remove(): để xoá phần tử khỏi list.

****

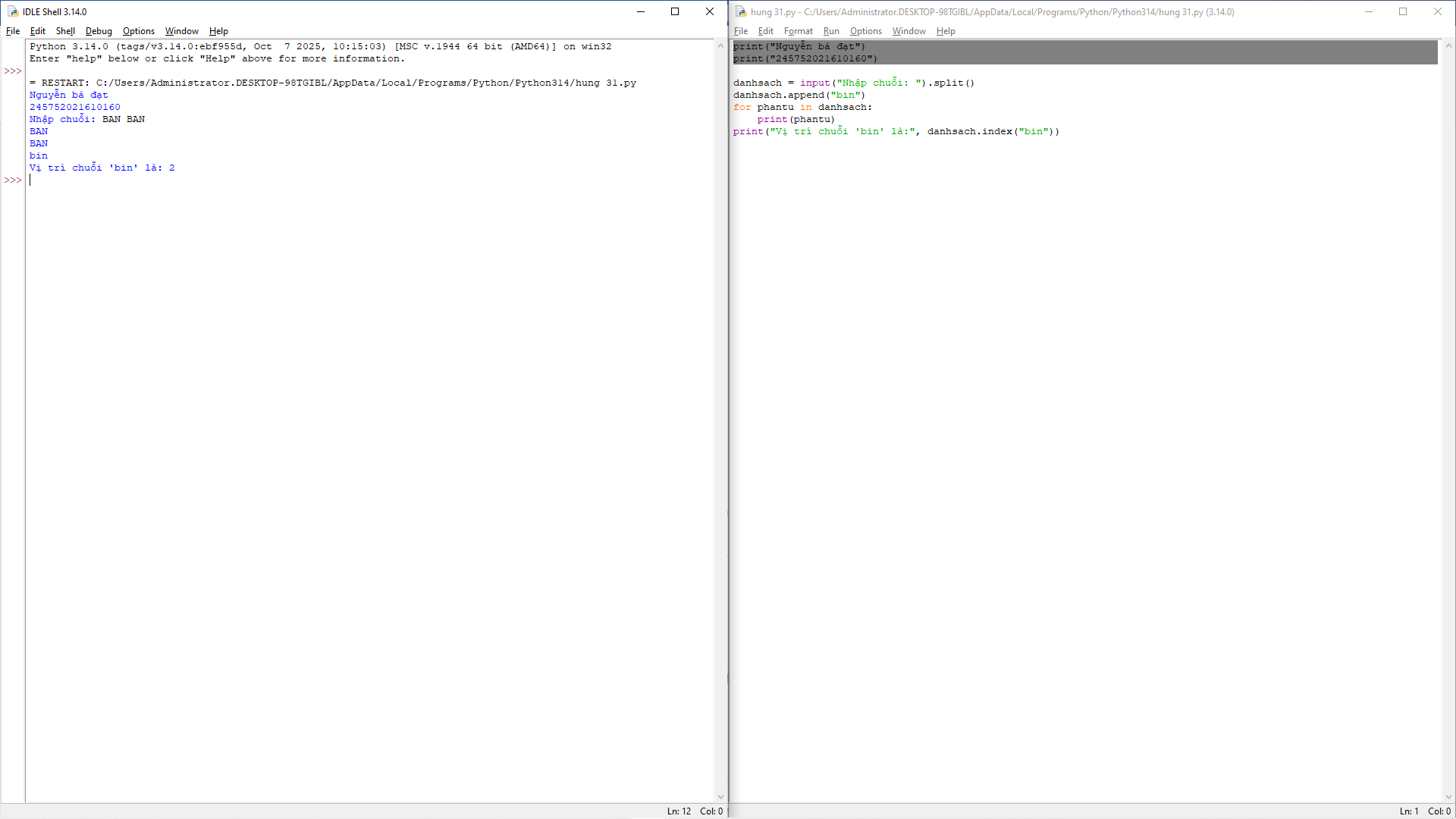
13.Tìm kiếm phần tử trong list.

- Lệnh index(): dùng để tìm vị trí của một phần tử trong danh sách.

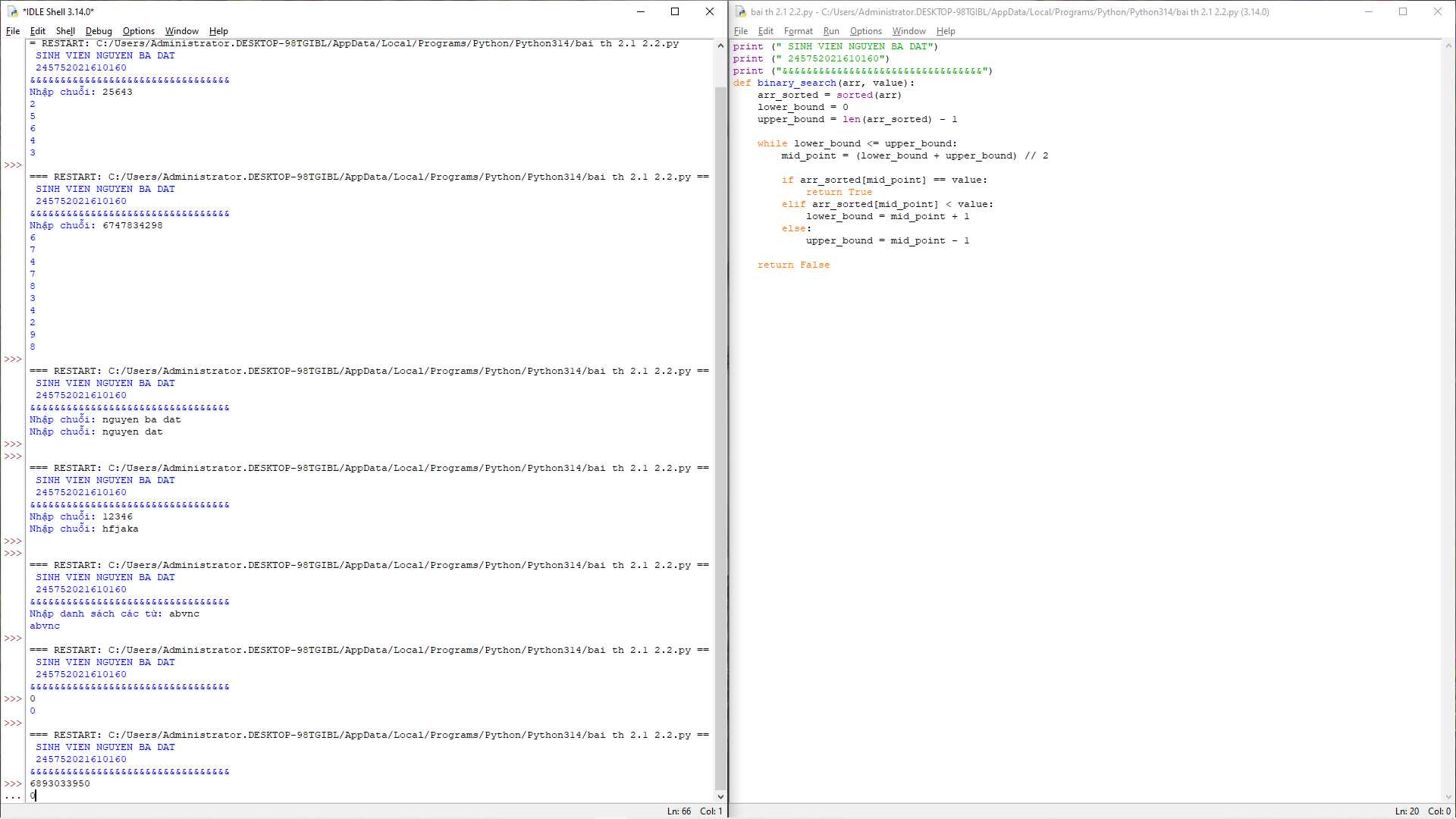
****

14.Sắp xếp các phần tử trong list.

- Lệnh sort() dùng để sắp xếp trực tiếp danh sách theo thứ tự tăng dần, nếu muốn giảm dần dùng sort(reverse=True).

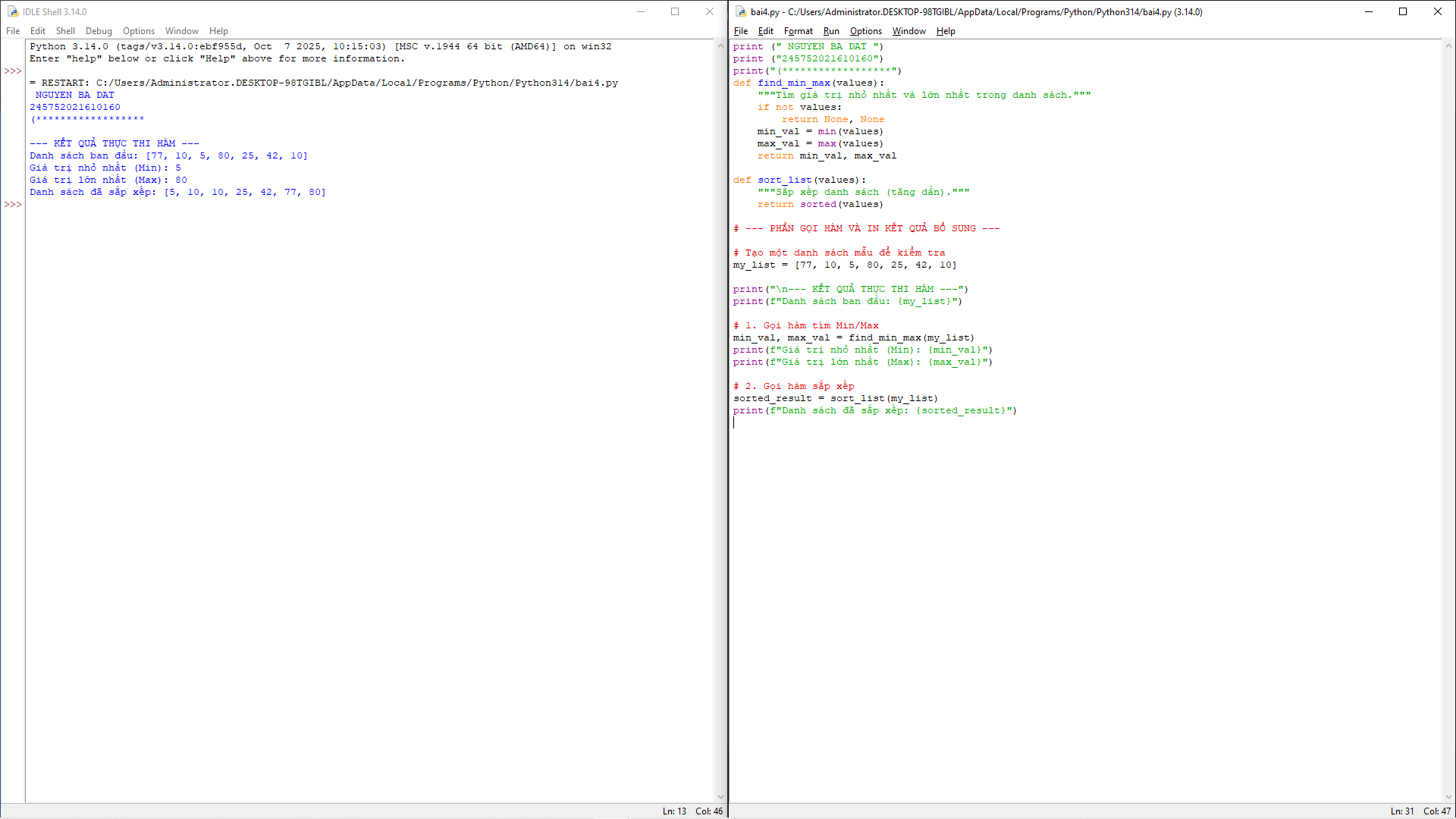
****

15.Người dùng nhập từ bàn phím liên tiếp các từ tiếng Anh viết tách nhau bởi dấu cách. Hãy nhập chuỗi đầu vào và tách thành các từ sau đó in ra màn hình các từ đó theo thứ tự từ điển.

****

16.Người dùng nhập từ bàn phím chuỗi các số nhị phân viết liên tiếp được nối nhau bởi dấu phẩy. Hãy nhập chuỗi đầu vào sau đó in ra những giá trị được nhập.

- Lệnh split(): dùng để cắt một chuỗi thành các phần tử nhỏ.

****

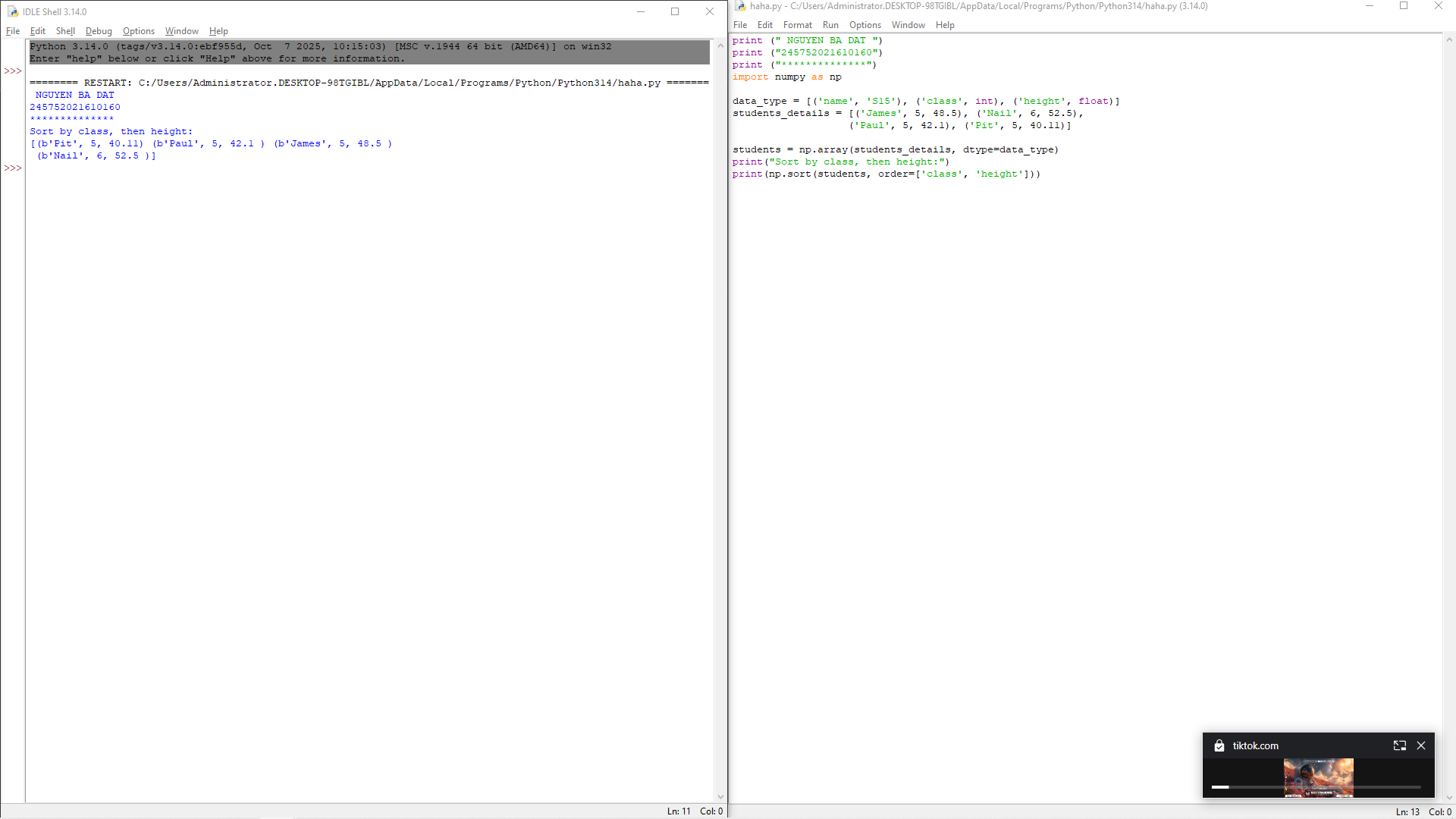
**Bài 5. Thiết kế module trong Python**

1. **Mục đích:**

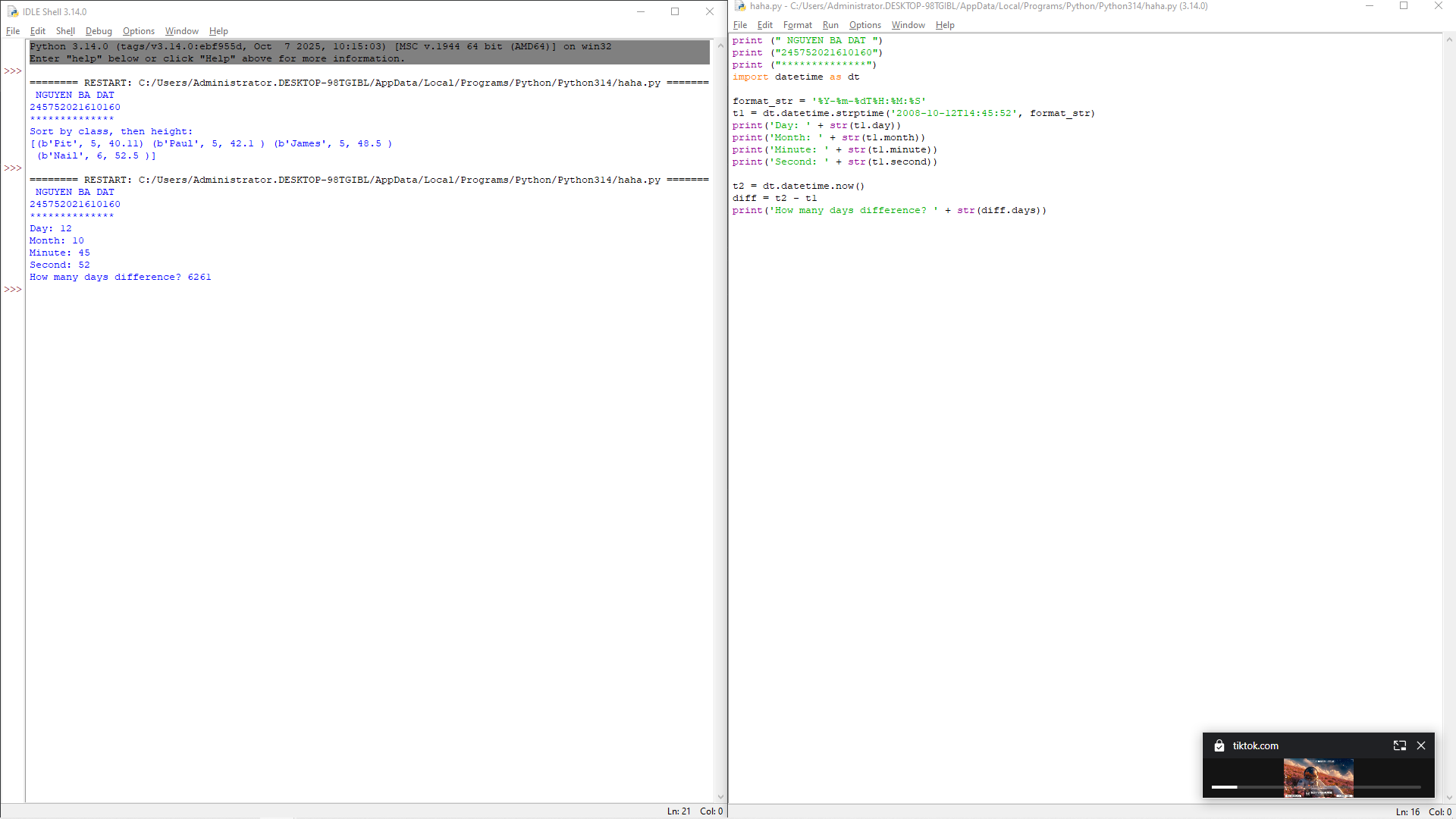
* Giúp sinh viên hiểu hơn về thiết kế module trong Python

1. **Các bước thực hiện và kết quả:**

1. Sử dụng module. Định nghĩa một module toán học gọi là mymath và sử dụng module này từ một tập lệnh riêng biệt.

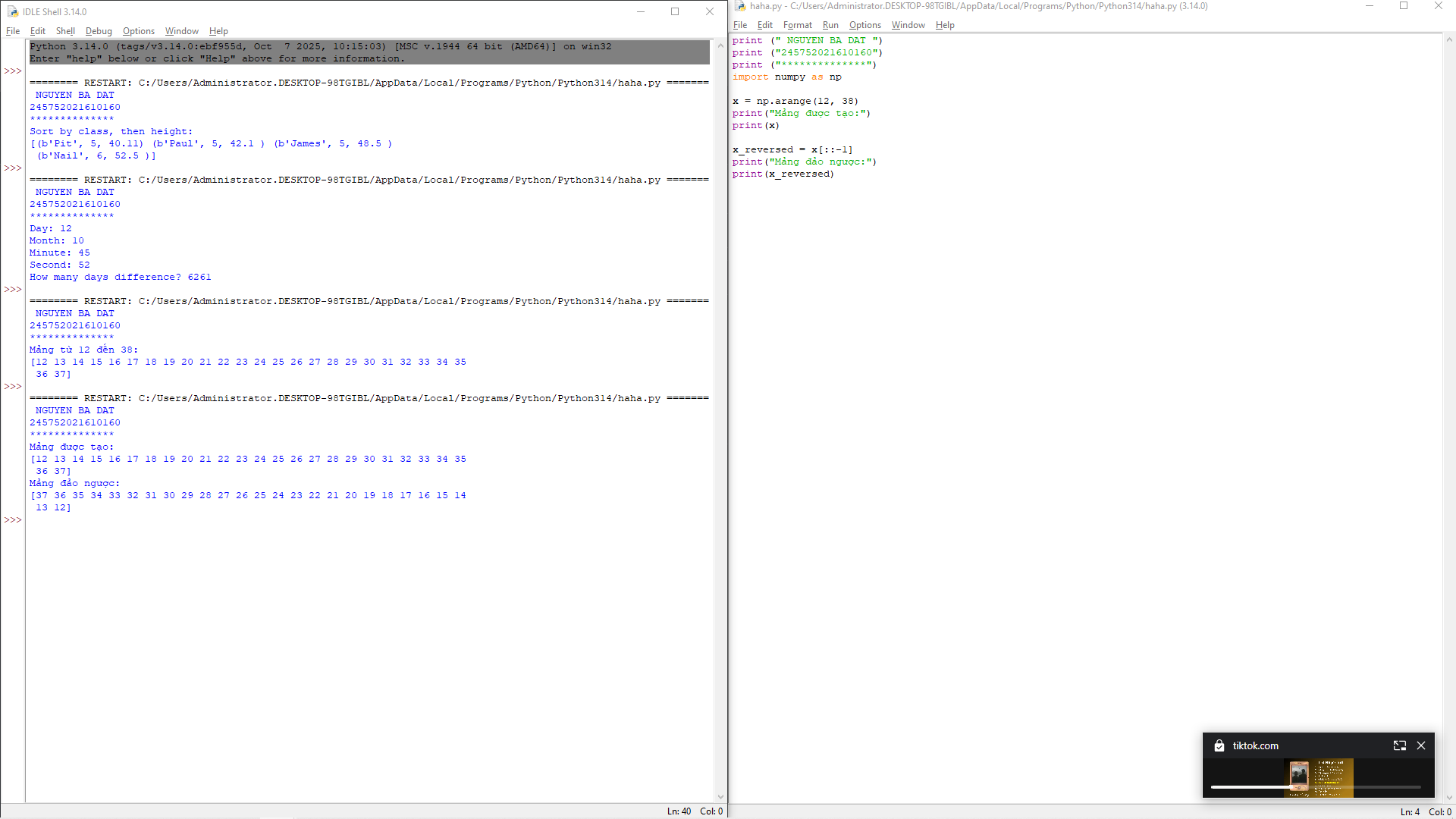
****

2. Sử dụng thư viện tiêu chuẩn của python (datetime).

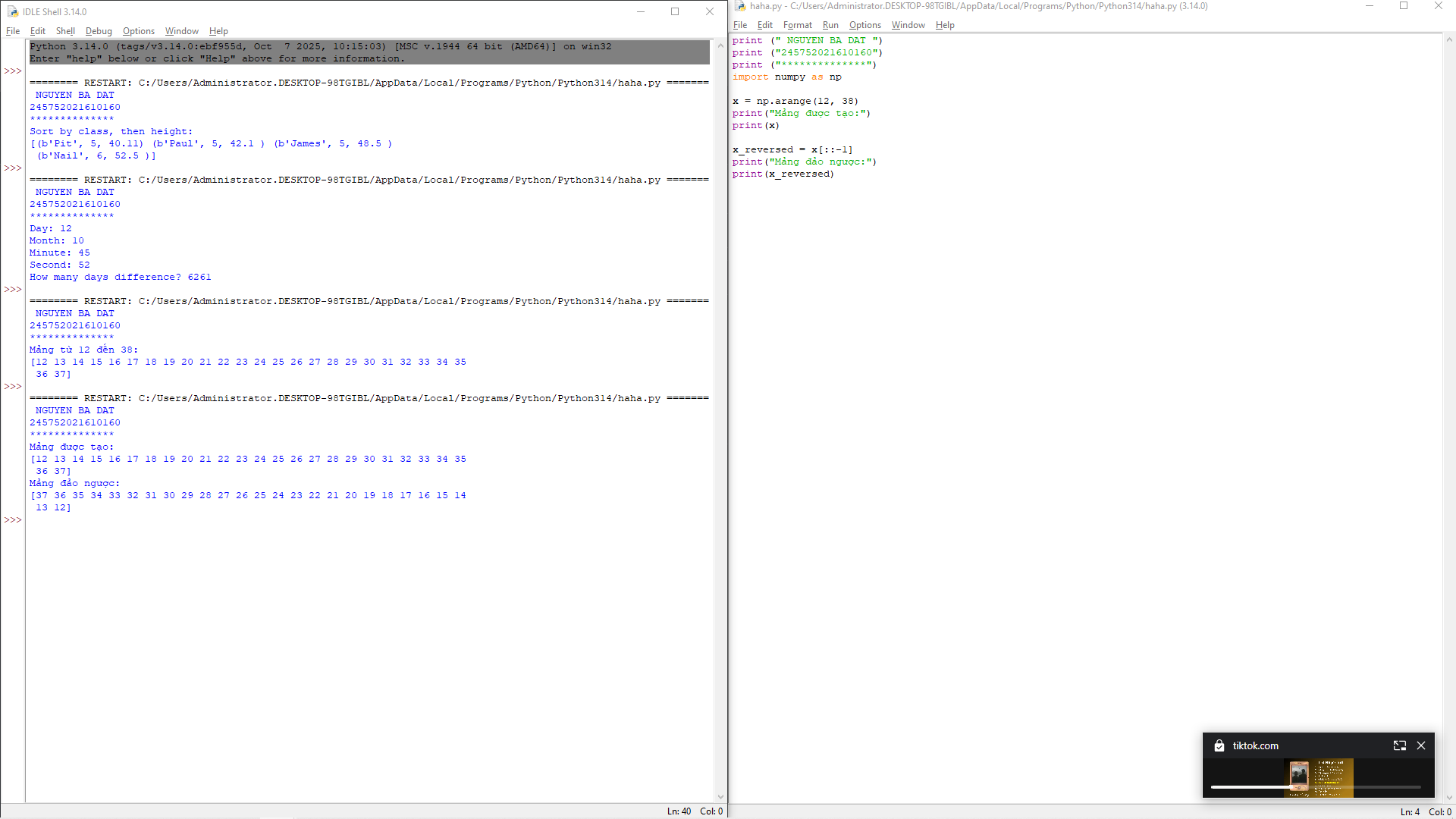


1. Viết chương trình sử dụng thư viện NumPy để tạo một mảng với các giá trị nằm trong khoảng từ 12 đến 38.

* Lệnh numpy.arange(start,stop,step): toạ một mảng số



4. Viết chương trình để tạo một mảng với các giá trị nằm trong khoảng từ 12 đến 38 và đảo ngược mảng đã tạo (phần tử đầu tiên trở thành cuối cùng).



5. Viết chương trình tìm phần tử lớn nhất và nhỏ nhất của một danh sách

- Số lượng và giá trị của list được nhập từ bàn phím

- Phương thức sắp xếp và tìm phần tử lớn nhất được viết thành module

A screenshot of a computer

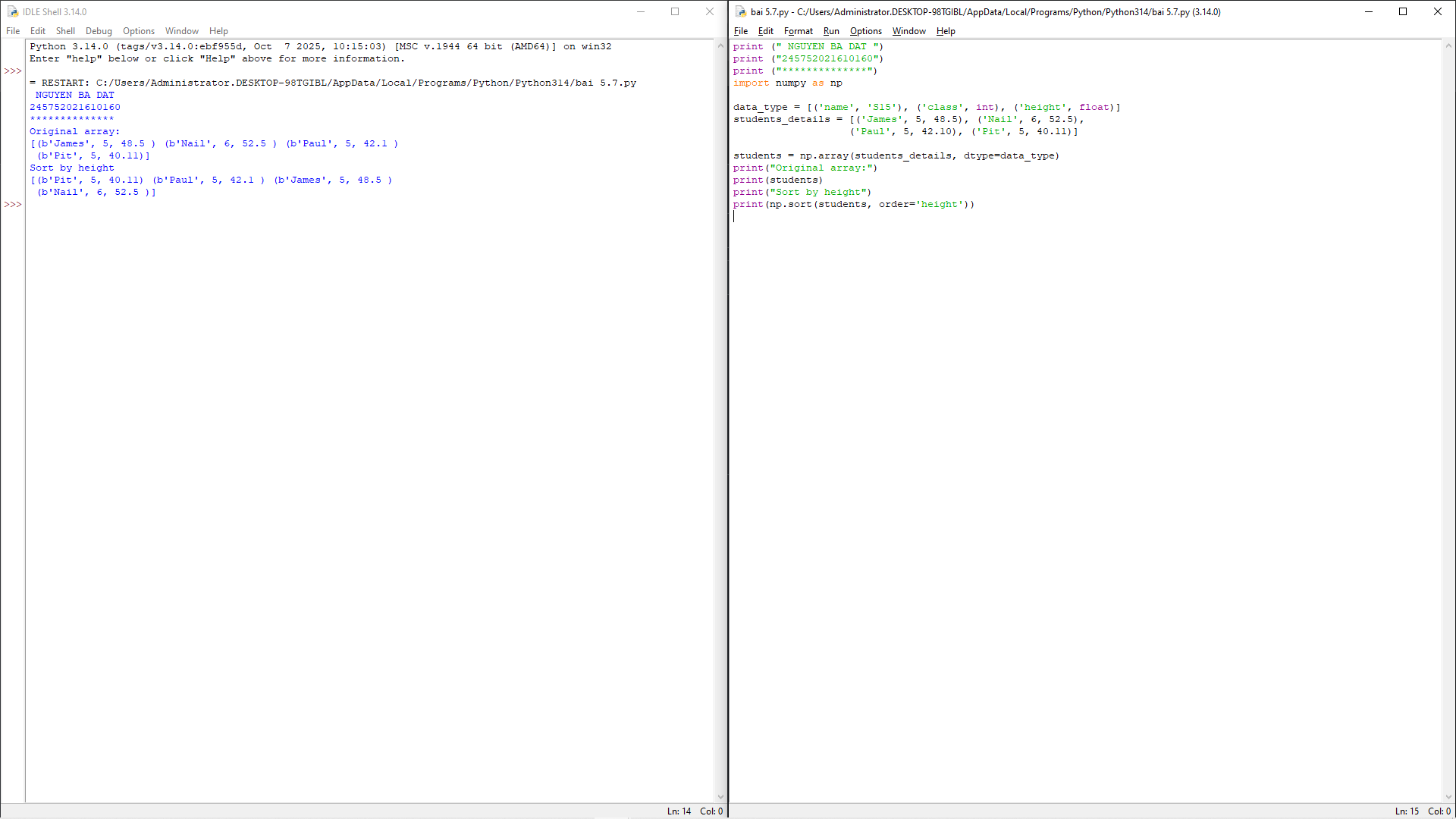
AI-generated content may be incorrect.

6. In ra vị trí phần tử lớn nhất và nhỏ nhất tìm được ở bài tập trên.

A screenshot of a computer

AI-generated content may be incorrect.

7. Viết chương trình sử dụng thư viện NumPy để tạo một mảng có cấu trúc từ tên sinh viên, chiều cao, lớp và các kiểu dữ liệu của họ. Bây giờ sắp xếp các mảng theo chiều cao.



8. Xây dựng hàm “Sequential\_Search(dlist, item)” (giải thuật tìm kiếm tuyến tính) dưới dạng module. Viết chương trình nhập một dlist n phần tử từ bàn phím và tìm kiếm phần tử item bất kỳ.

A screenshot of a computer

AI-generated content may be incorrect.

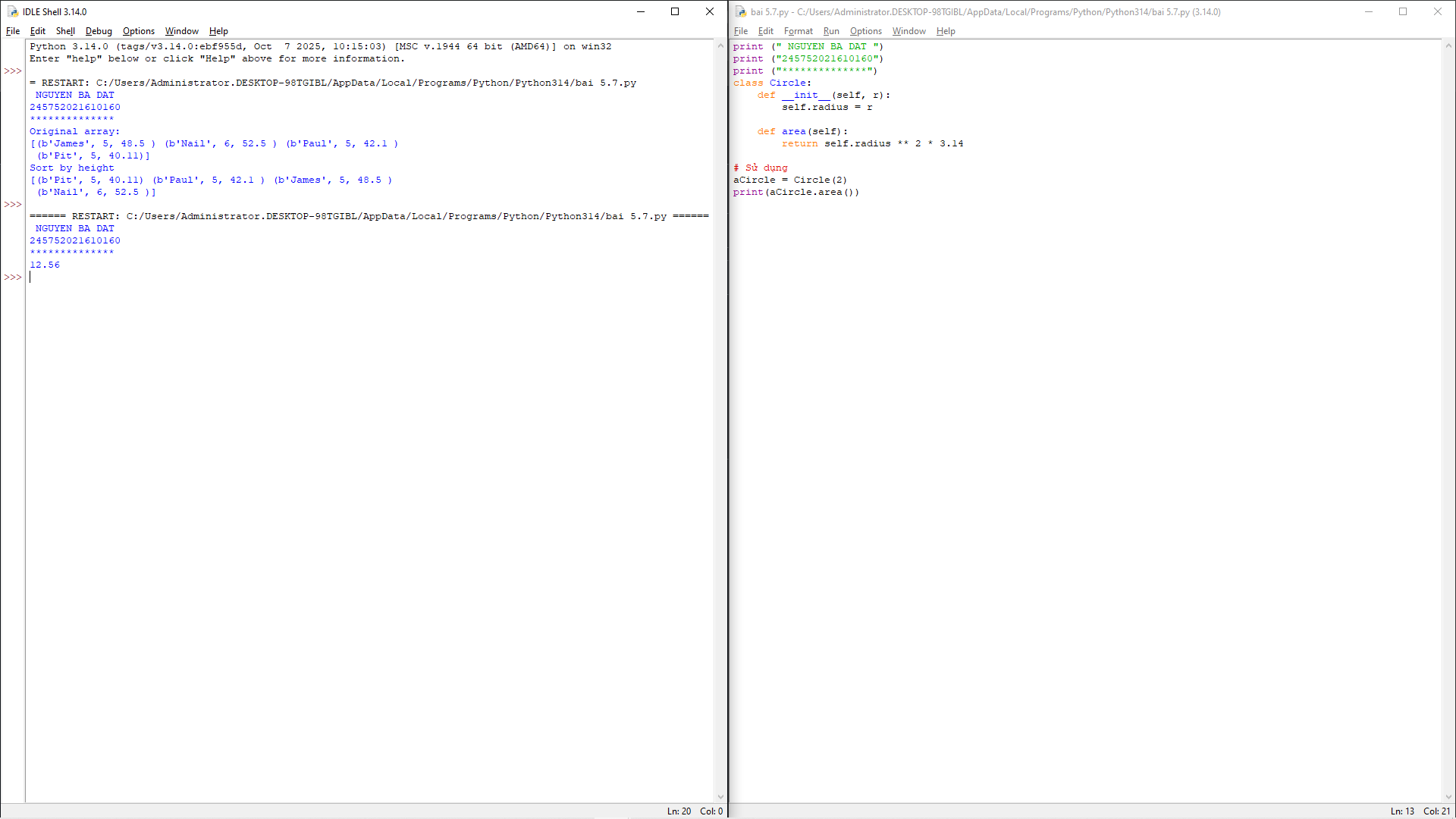
**Bài 6: Lập trình hướng đối tượng trong Python**

1. **Mục đích:**

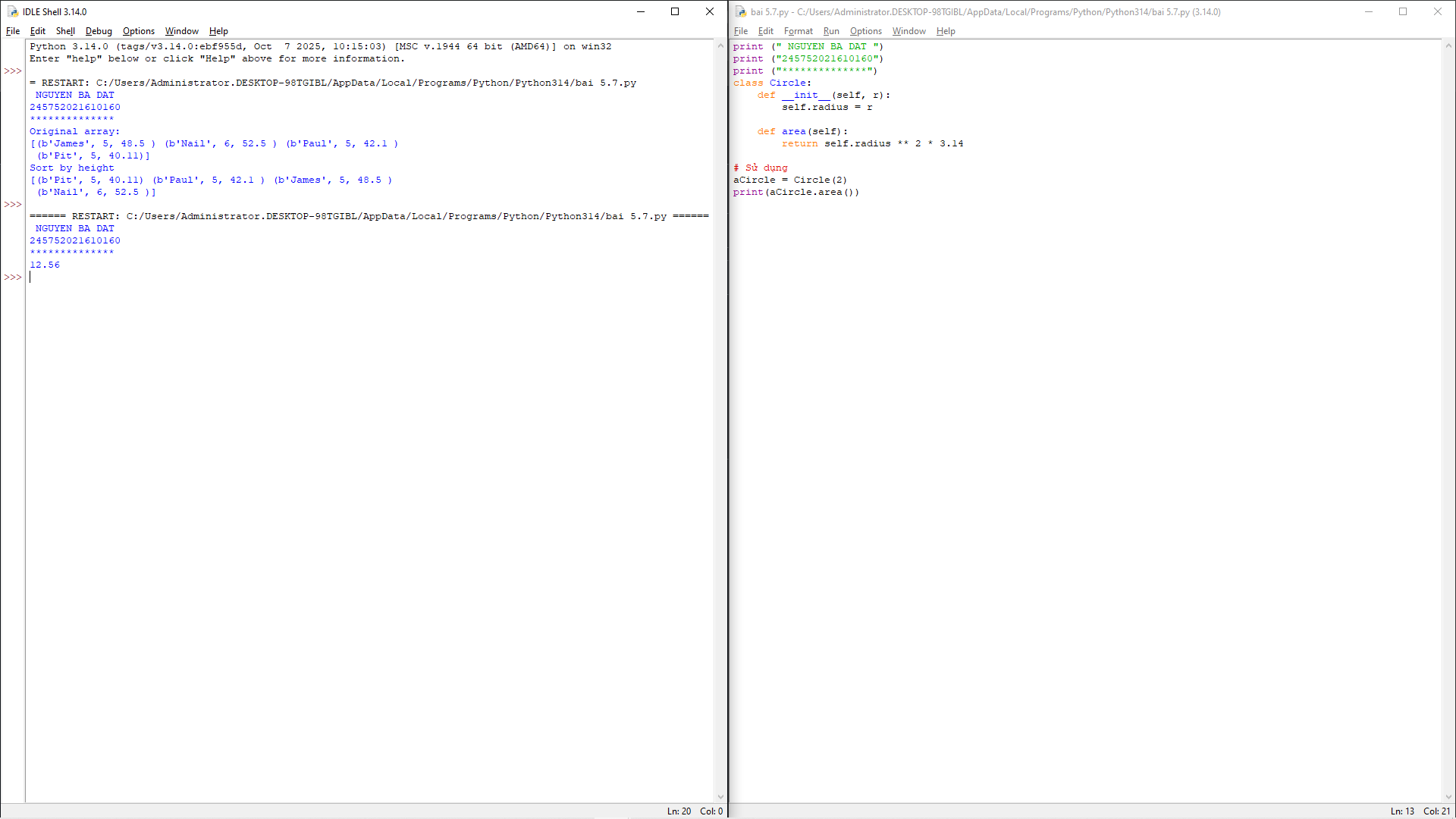
* Giúp sinh viên nắm bắt về lập trình hướng đối tượng trong lập trình python.

1. **Các bước thực hiện và kết quả:**

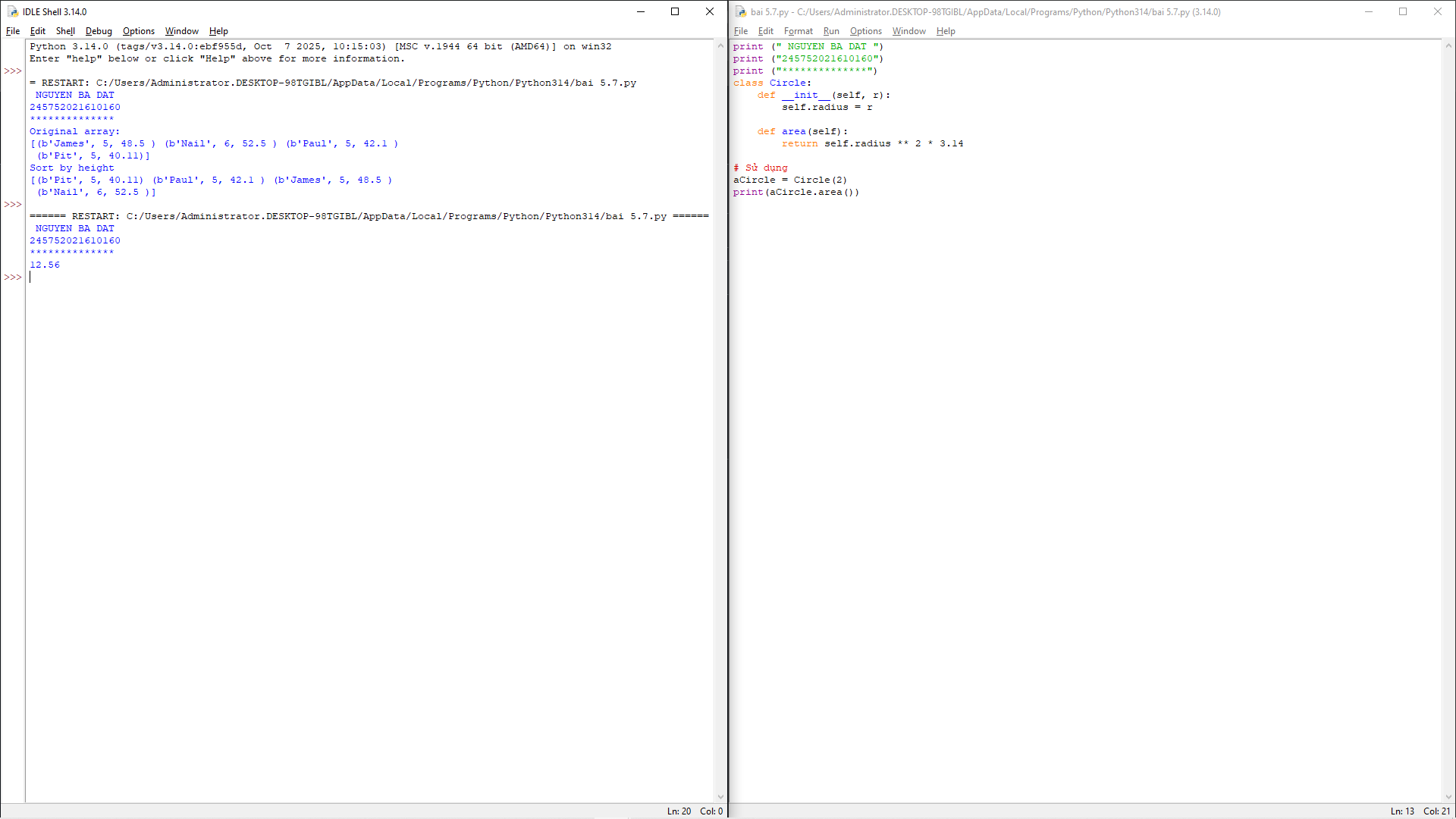
1. Định nghĩa một class có tên là Circle có thể được xây dựng từ bán kính. Circle có một method có thể tính diện tích.



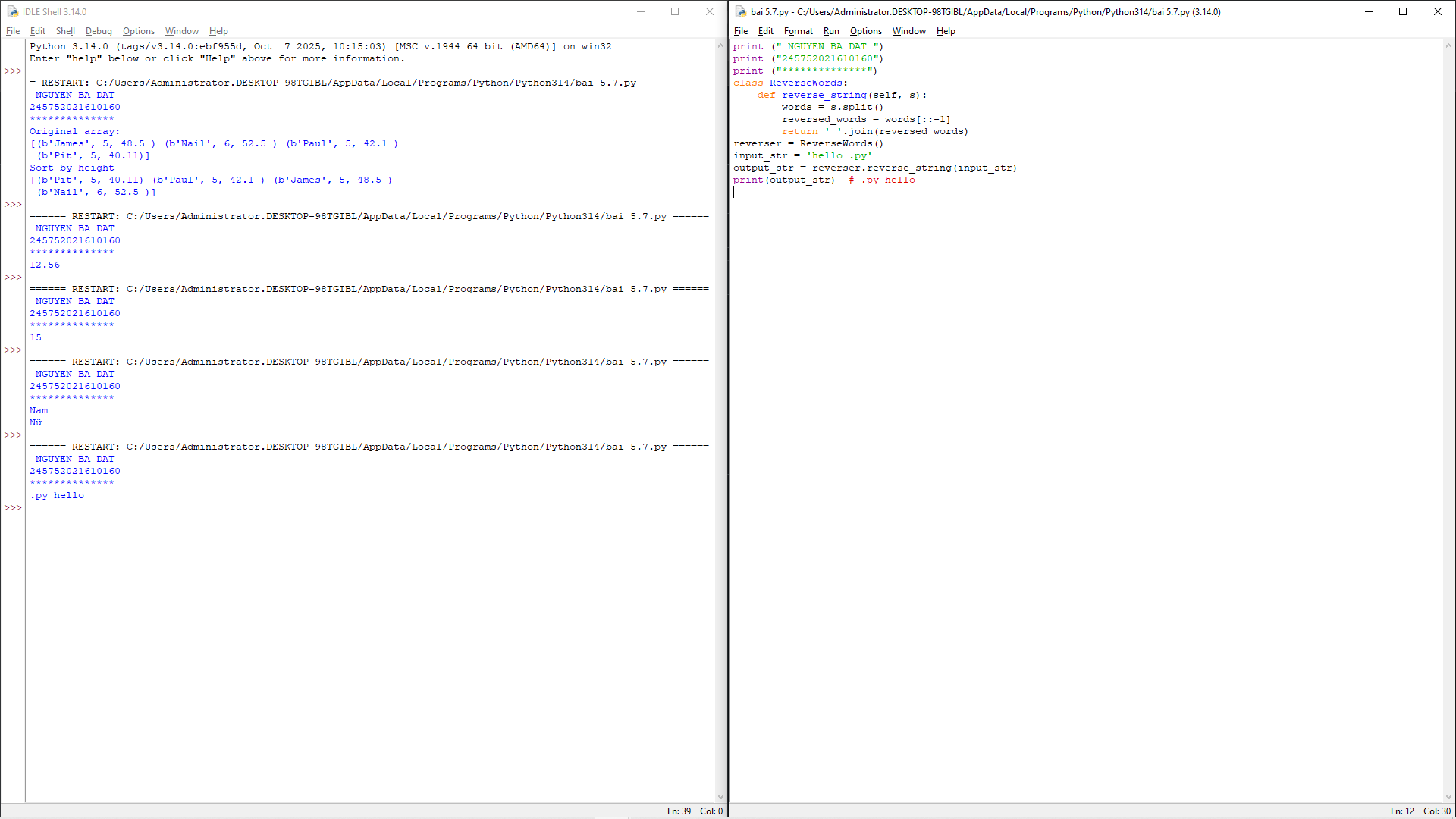
2. Định nghĩa class có tên là Hinhchunhat được xây dựng bằng chiều dài và chiều rộng. Class Hinhchunhat có method để tính diện tích.



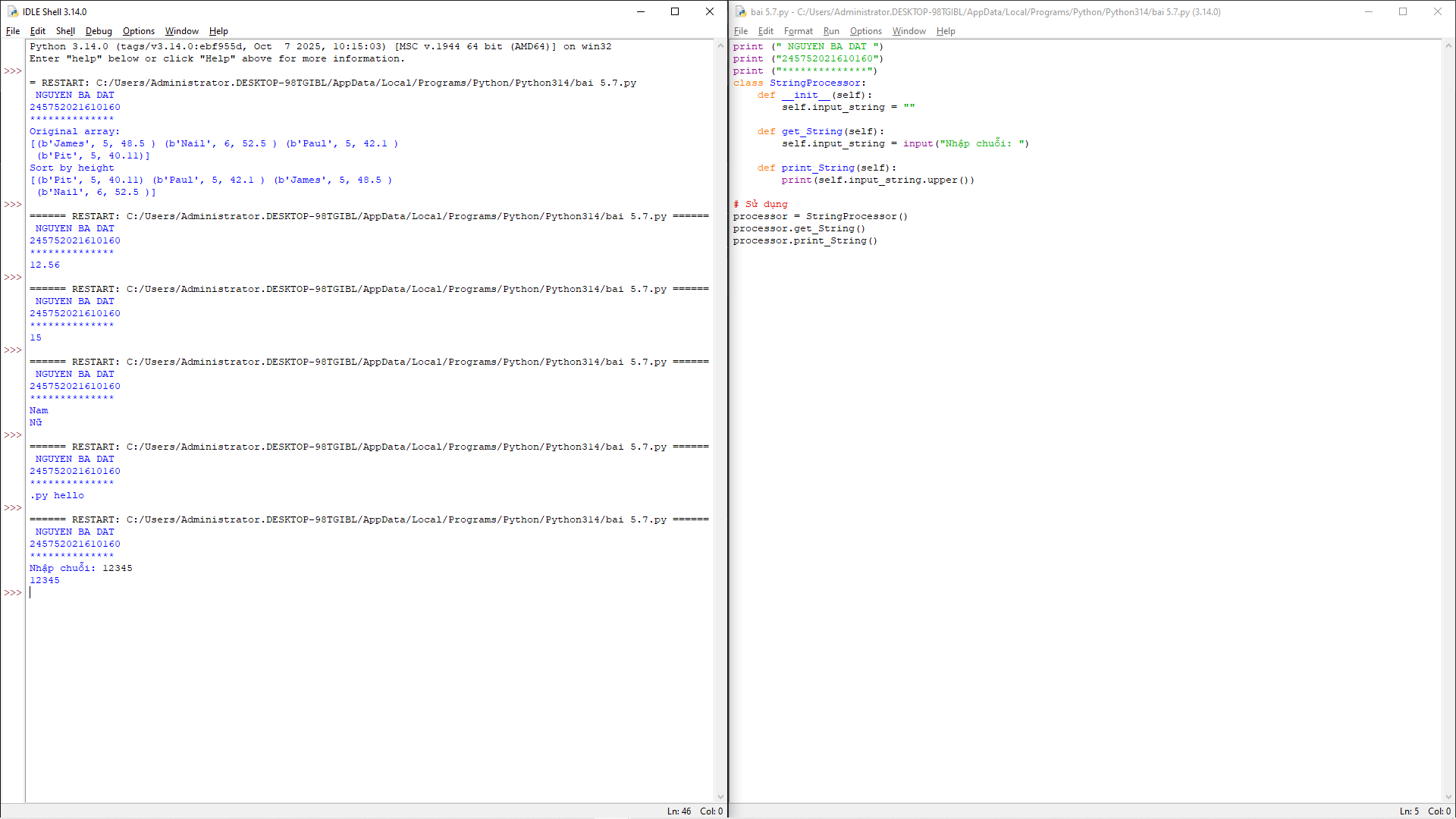
3. Định nghĩa class Nguoi và 2 class con của nó: Nam, Nu. Tất cả các class có method "getGender" có thể in "Nam" cho class Nam và "Nữ" cho class Nu.



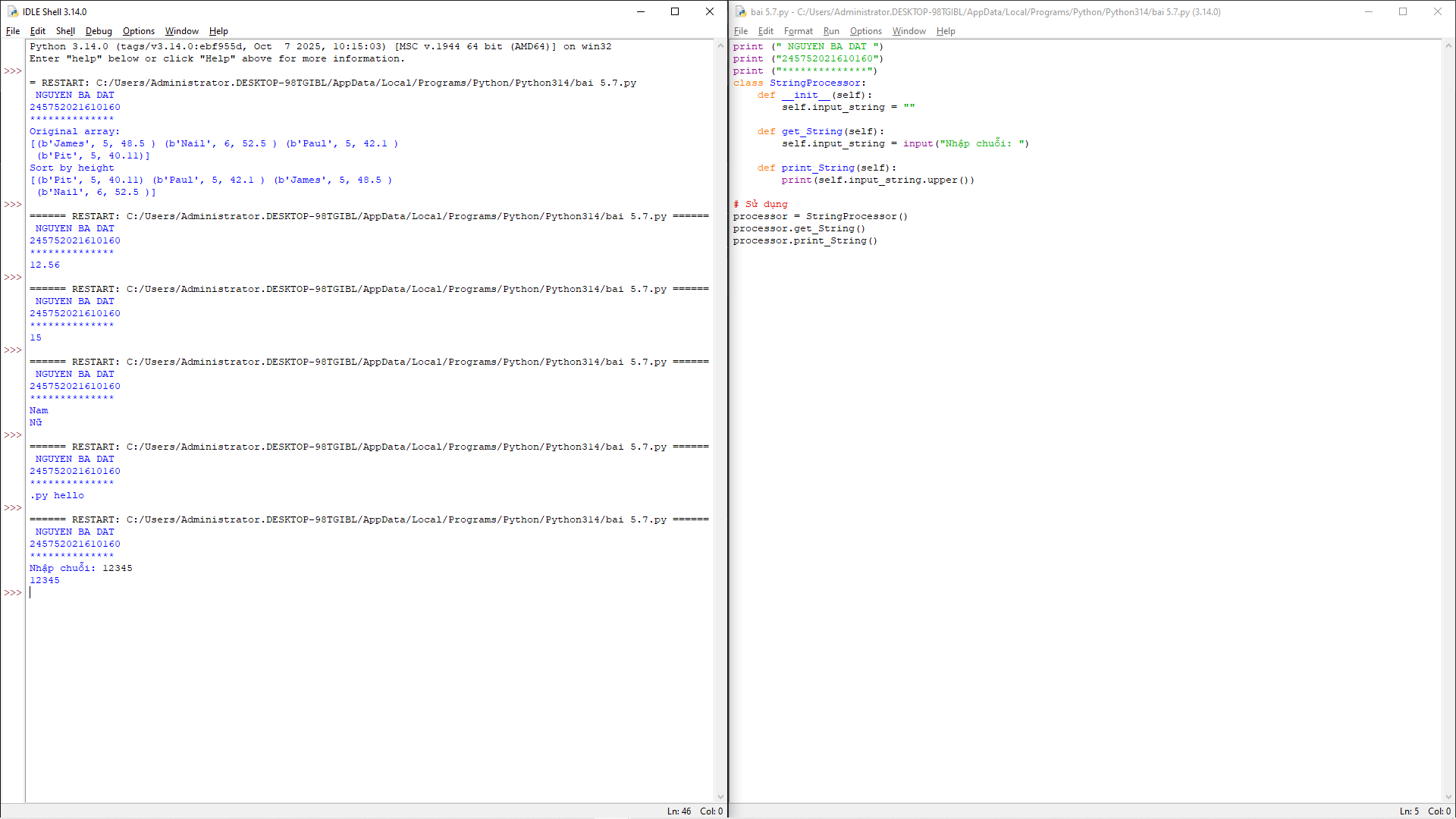
4. Viết chương trình Python dưới dạng class để chuyển đổi một số La Mã thành một số nguyên.



5. Viết chương trình Python dưới dạng class để đảo ngược chuỗi từ từng chữ.



6. Viết một class Python có hai phương thức get\_String và print\_String. get\_String chấp nhận một chuỗi từ người dùng và print\_String in chuỗi đó bằng chữ in hoa.

****

7. Viết một class Python có tên Circle được xây dựng theo bán kính và hai phương thức sẽ tính diện tích và chu vi của hình tròn.

**A screenshot of a computer

AI-generated content may be incorrect.**

**Bài 7: Thao tác trên tập tin và thư mục trong Python**

1. **Mục đích:**

* Giúp sinh viên có thể thao tác với các file văn bản sử dụng python.

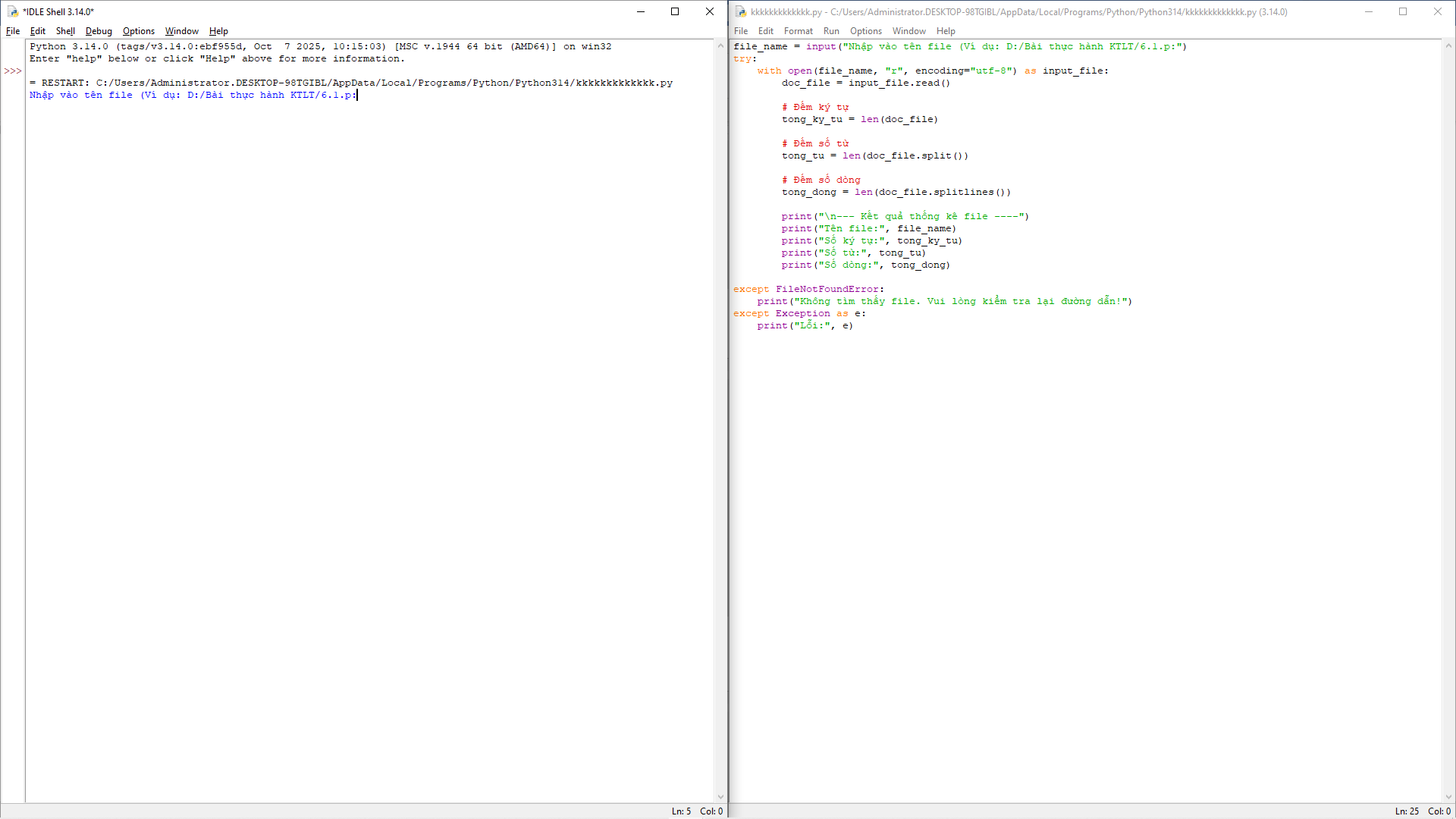
1. **Các bước thực hiện và kết quả:**
2. Chương trình đọc file và in đảo ngược kết quả.

* Để đọc dữ liệu từ một file, ta sử dụng hàm: open(tên\_file, chế\_độ)
* Để loại bỏ ký tự xuống dòng \n , ta dùng: line.rstrip()
* Có thể đảo ngược một chuỗi bằng toán tử cắt chuỗi: chuoi[::-1]
* [::-1] nghĩa là lấy toàn bộ chuỗi từ đầu đến cuối nhưng theo bước **-1**, tức là đi lùi từ cuối về đầu.

****

1. Chương trình đọc một file, tính số ký tự, số từ và số dòng của file.

* Để đọc dữ liệu từ một file, ta sử dụng hàm: open(tên\_file, chế\_độ)

****

3. Viết chương trình Python để đọc toàn bộ tệp văn bản.

* Để đọc dữ liệu từ một file, ta sử dụng hàm: open(tên\_file, chế\_độ)

**A screenshot of a computer

AI-generated content may be incorrect.**

4. Chương trình Python để đọc n dòng đầu tiên của tệp.

**A screenshot of a computer

AI-generated content may be incorrect.**

5. Chương trình Python để nối văn bản vào tệp và hiển thị văn bản.

**A screenshot of a computer

AI-generated content may be incorrect.**

6. Chương trình Python để đọc n dòng cuối cùng của tệp

**A screenshot of a computer

AI-generated content may be incorrect.**

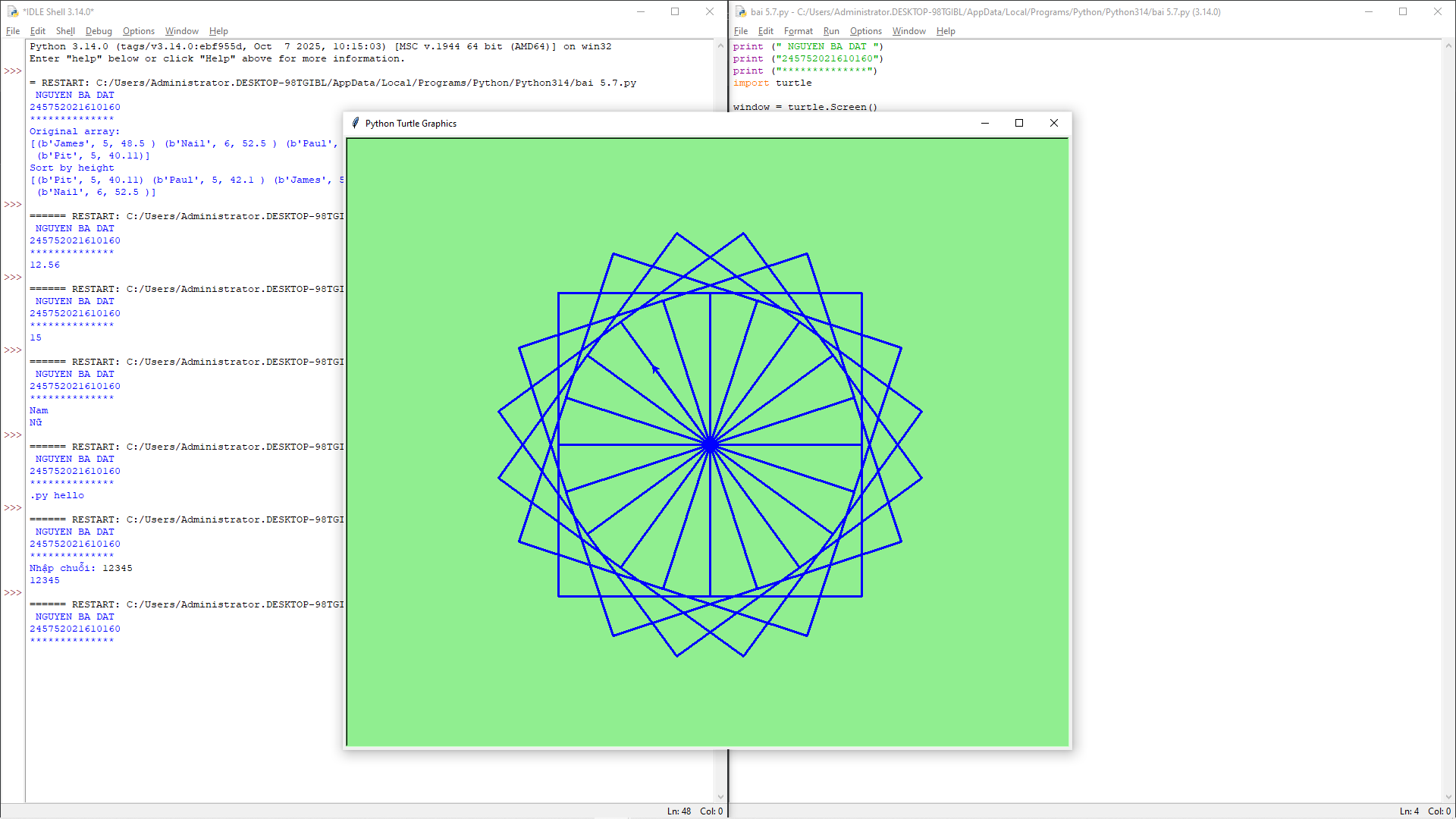
**Bài 8: Lập trình giao diện trong Python**

1. **Mục đích:**

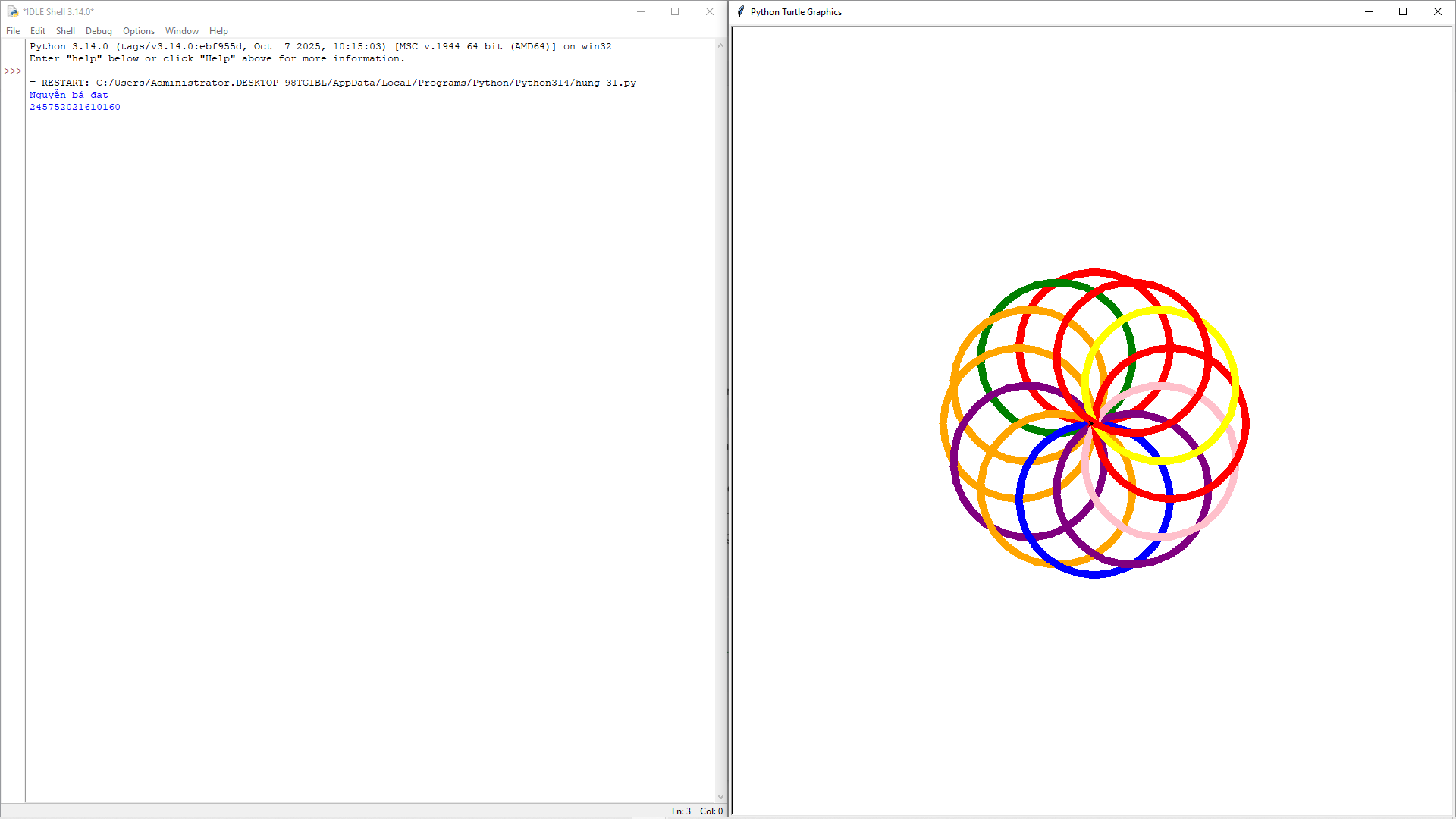
* Giúp sinh viên làm quen với lập trình đồ họa và sử dụng thư viện turtle, Tkinter trong python.

1. **Các bước thực hiện và kết quả:**
2. Viết chương trình đồ họa sử dụng thư viện turtle.

* Turtle.Screen() : dùng để tạo cửa sổ .
* Bg(“color”): dùng để đặt màu nền.
* Fg(“color”): dùng để đặt màu chữ.
* Pencolor(): dùng để đặt màu nét vẽ.
* Pensize(): dùng để chỉnh kích thước nét vẽ.

****

2. Viết chương trình đồ họa sử dụng thư viện turtle.

****

3. Dựa trên các kết quả đạt được từ các chương trình trên hãy viết chương trình hiển thị hình ảnh đồ họa sau.

**A screenshot of a computer

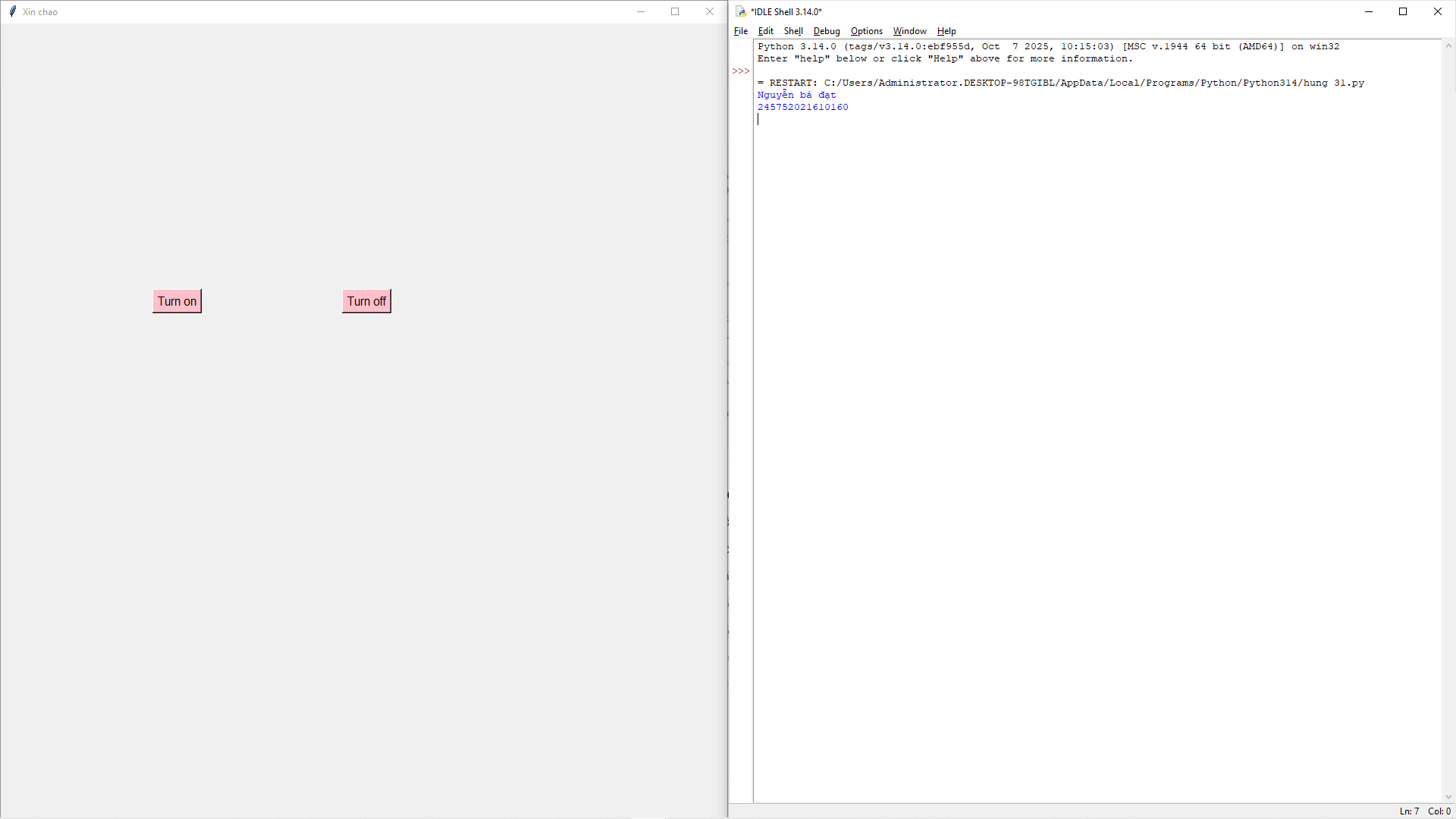
AI-generated content may be incorrect.**

4. Viết chương sử dụng thư viện đồ họa tkinter thực hiện:

a) Xây dựng cửa sổ đồ họa window form

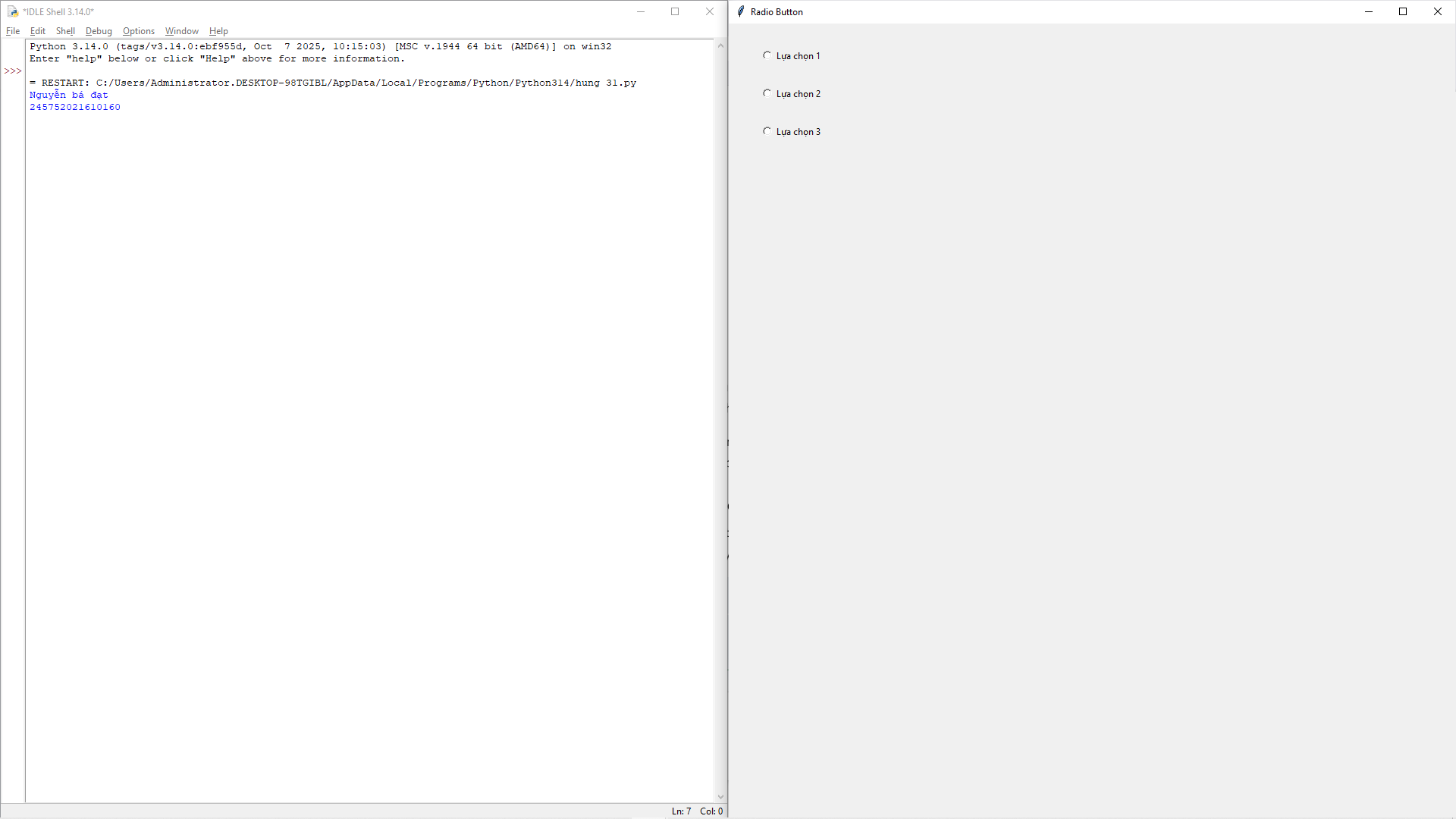
b) Thêm một widget (button) vào window form

c) Xây dựng phương thức xử lý sự kiện phím bấm

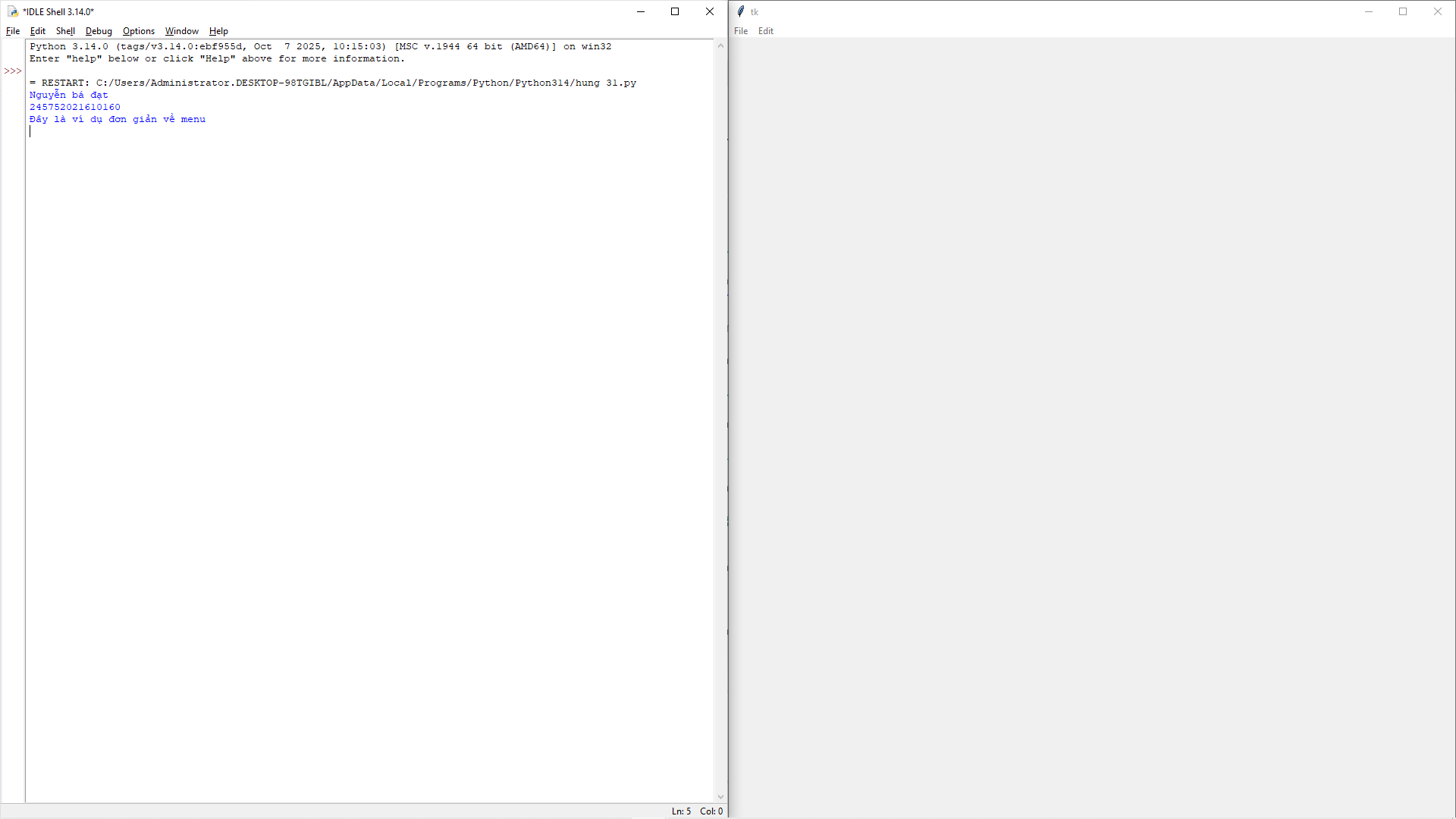
****

5. Sử dụng thư viện tkinter thực hiện:

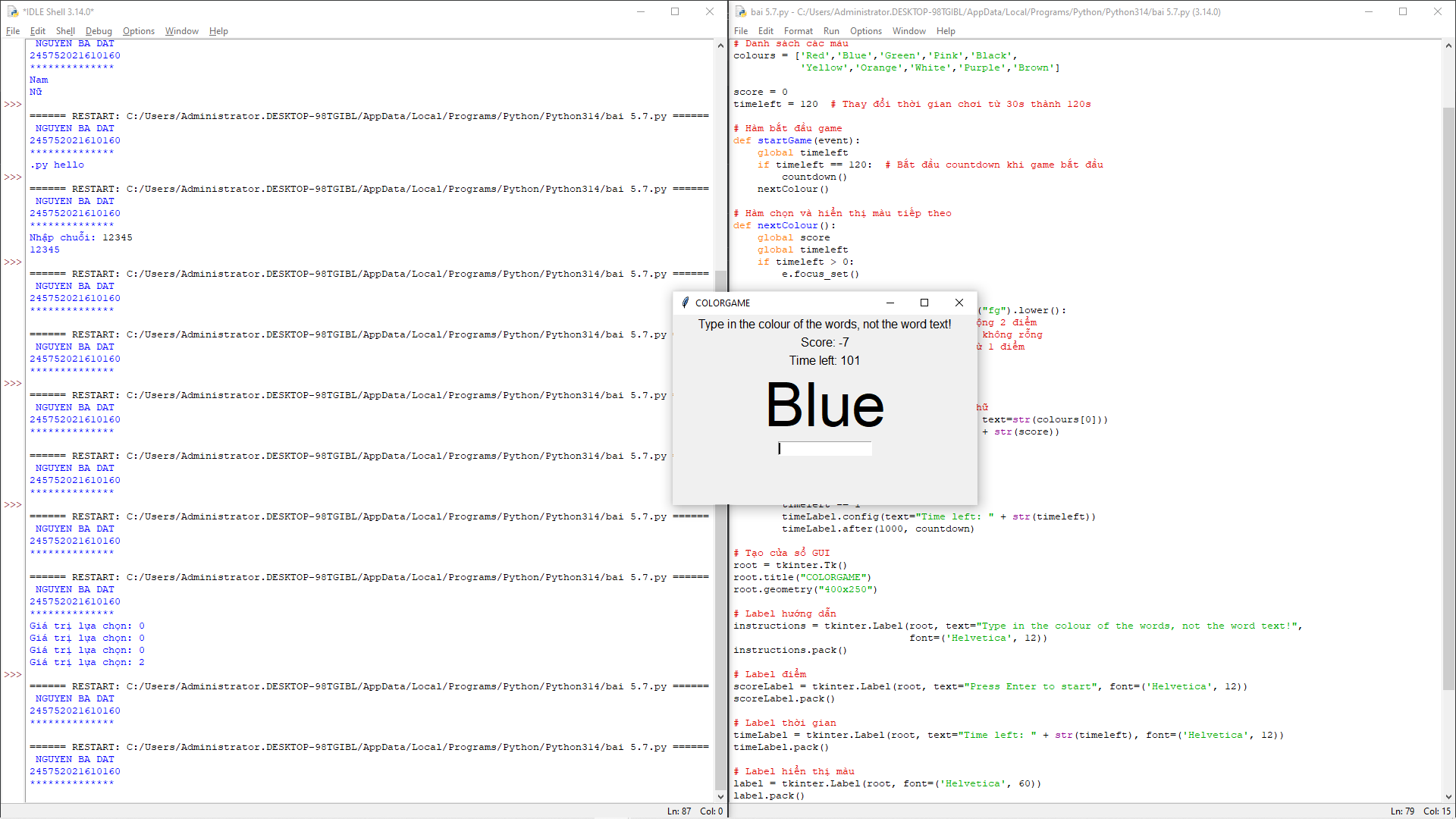
- Xây dựng các radio button cho phép thực hiện các lựa chọn khác nhau

****

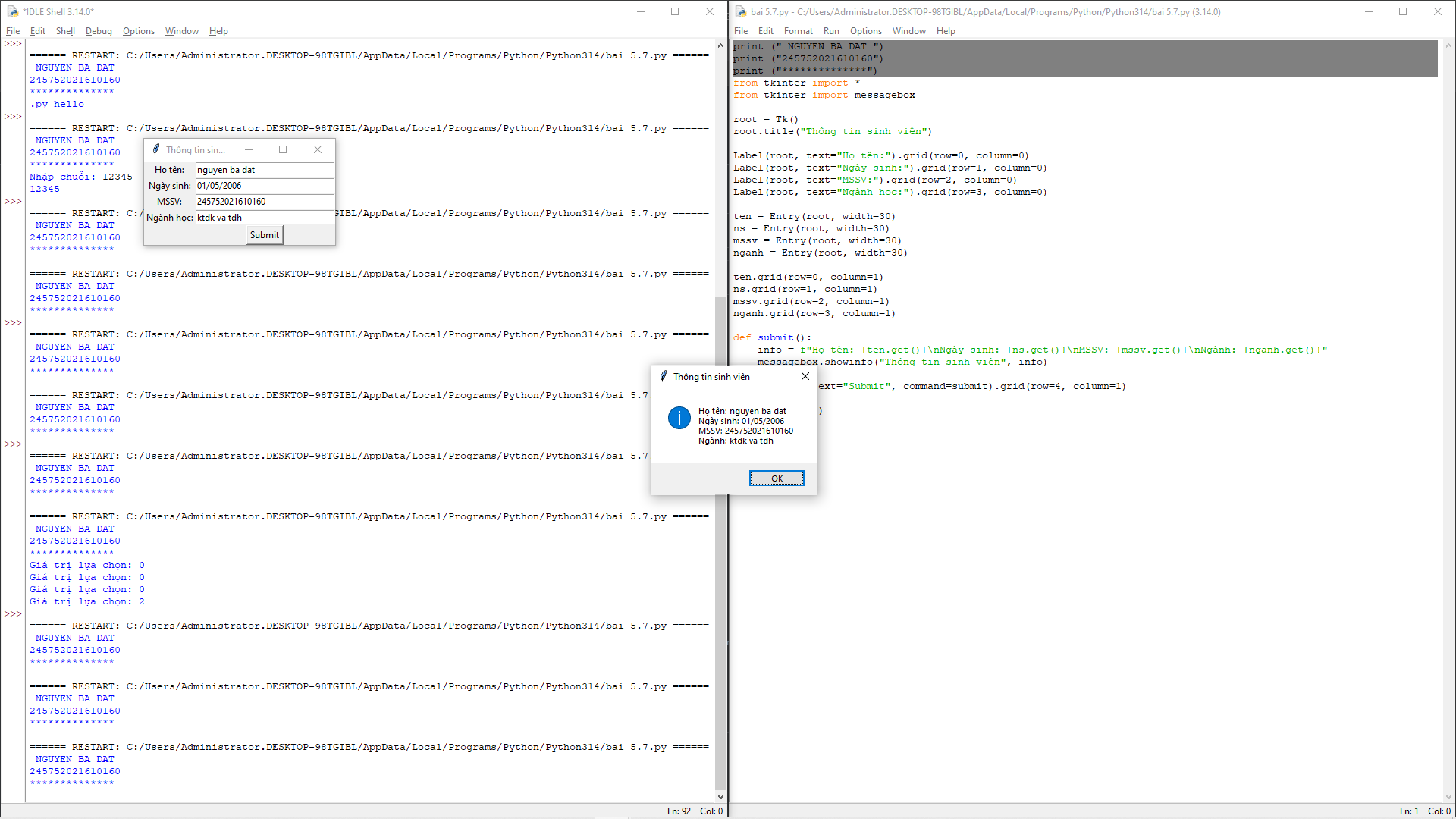
6. Viết chương trình thực hiện tạo menu.

****

7. Sử dụng thư viện đồ họa tkinter tạo game tiếng anh.

****

8. Viết chương trình graphic sử dụng thư viện Tkinter xây dựng form hiển thị thôn tin cá nhân và form có nội dung như hình ở dưới, khi bấm vào nút “Click Me” thông tin nút radio button đang lựa chọn sẽ được chỉ ra.

****

**TÀI LIỆU THAM KHẢO**

[1]. Lutz, Mark.(2013) “*Learning Python, 5th Edition*”

[2]. Sweigart, Al. (2019) “*Automate the Boring Stuff with Python, 2nd Edition”*.

[3]. Codecademy. “*Learn Python 3”*.

[4] Zelle, John.(2016) “*Python Programming: An Introduction to Computer Science, 3rd Edition”*.

[5] McKinney, Wes.(2022) “*Python for Data Analysis, 3rd Edition”*.

[6] Allen B. Downey, Think Python, O'Reilly Media, Inc, 2015.

[7] Corey Schafer – *Python Tutorials*. YouTube.