|  |
| --- |
| TRƯỜNG ĐẠI HỌC VINH  **VIỆN KỸ THUẬT CÔNG NGHỆ** |



BÁO CÁO THỰC HÀNH

**KỸ THUẬT LẬP TRÌNH**

MÃ HỌC PHẦN: ELE20004

SVTH: TRẦN VĂN CƯƠNG

MSSV: 235752021610043

GVHD: MAI THẾ ANH

**NGHỆ AN - 2024**

**Mục lục**

***Bài 1. Thực hiện các thuật toán bằng phần mềm Flowgorithm (2-4)***

***Bài 2. Các cú pháp, kiểu dữ liệu, lệnh điều khiển trong lập trình Python (5-9)***

***Bài 3. Lập trình hàm trong Python (10-14)***

***Bài 4. Các kiểu dữ liệu có cấu trúc trong Python(15-22)***

***Bài 5. Thiết kế module trong Python (23-28)***

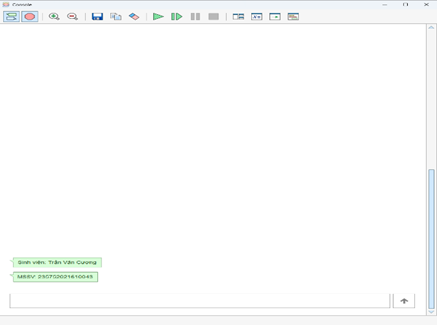
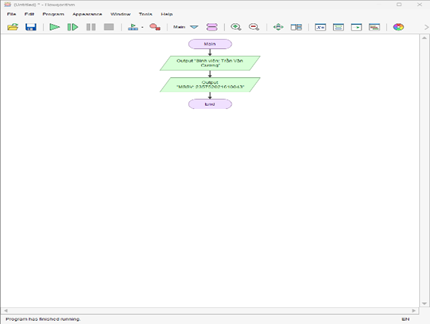
***Bài 6: Lập trình hướng đối tượng trong Python(29-31)***

***Bài 7: Thao tác trên tập tin và thư mục trong Python (32-34)***

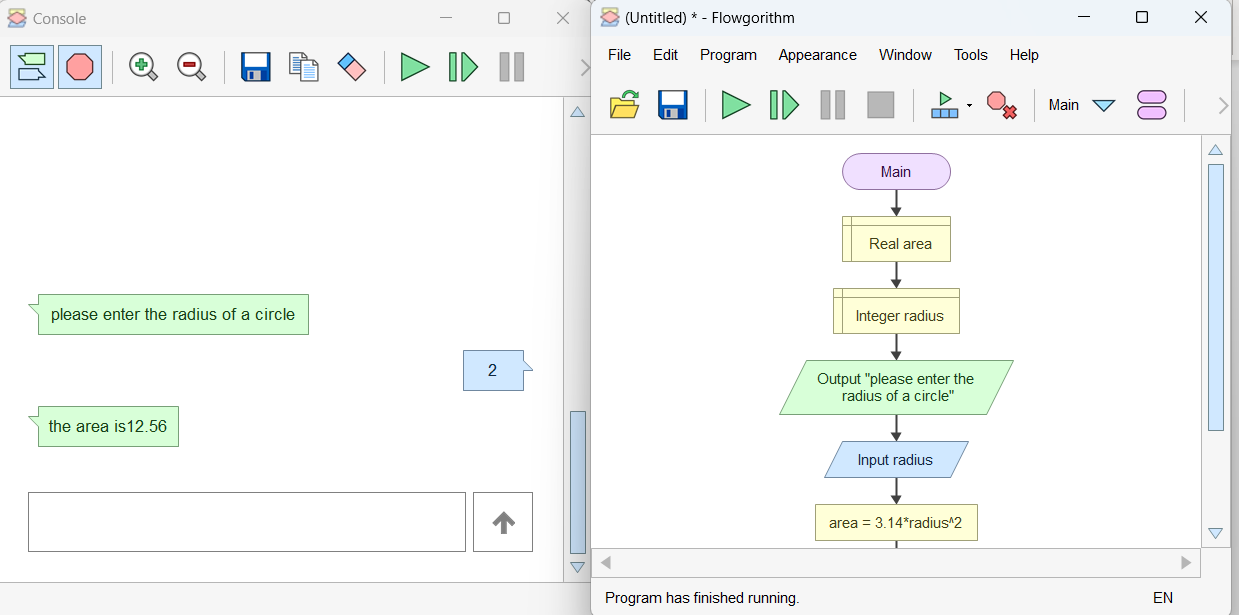
***Bài 8: Lập trình giao diện trong Python (35-40)***

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

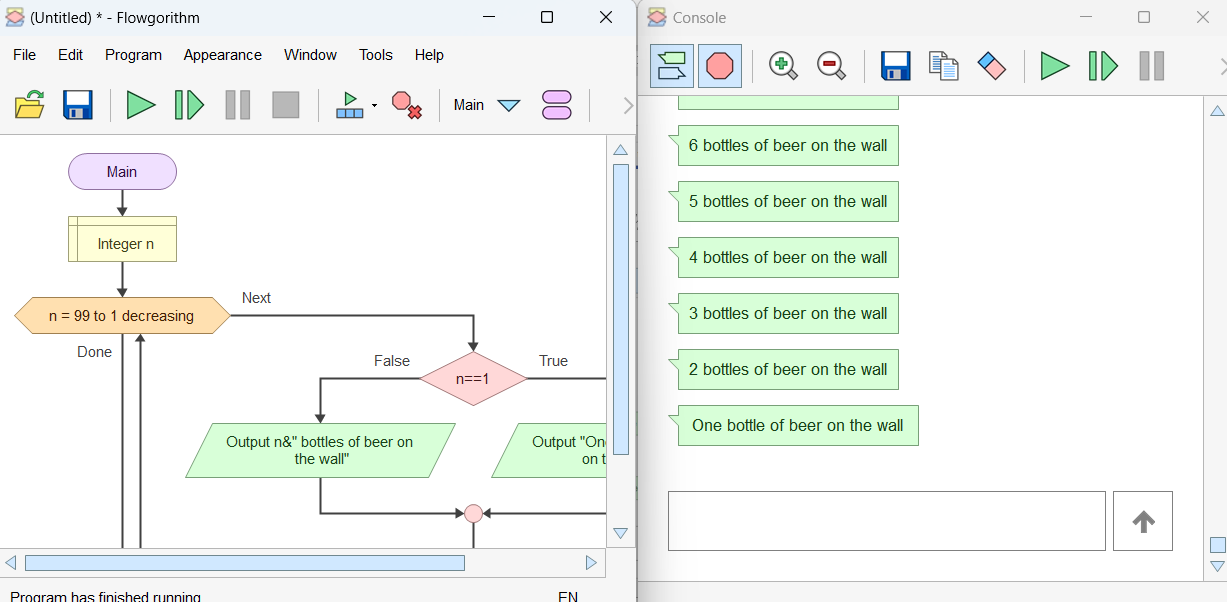
* 1. **Mục đích**
* Sử dụng phần mềm Flowgorithm trong thiết kế và biểu diễn thuật toán;
* Xây dựng thuật toán cho các bài toán cụ thể trên Flowgorithm
  1. **Các bước thực hiện và kết quả**

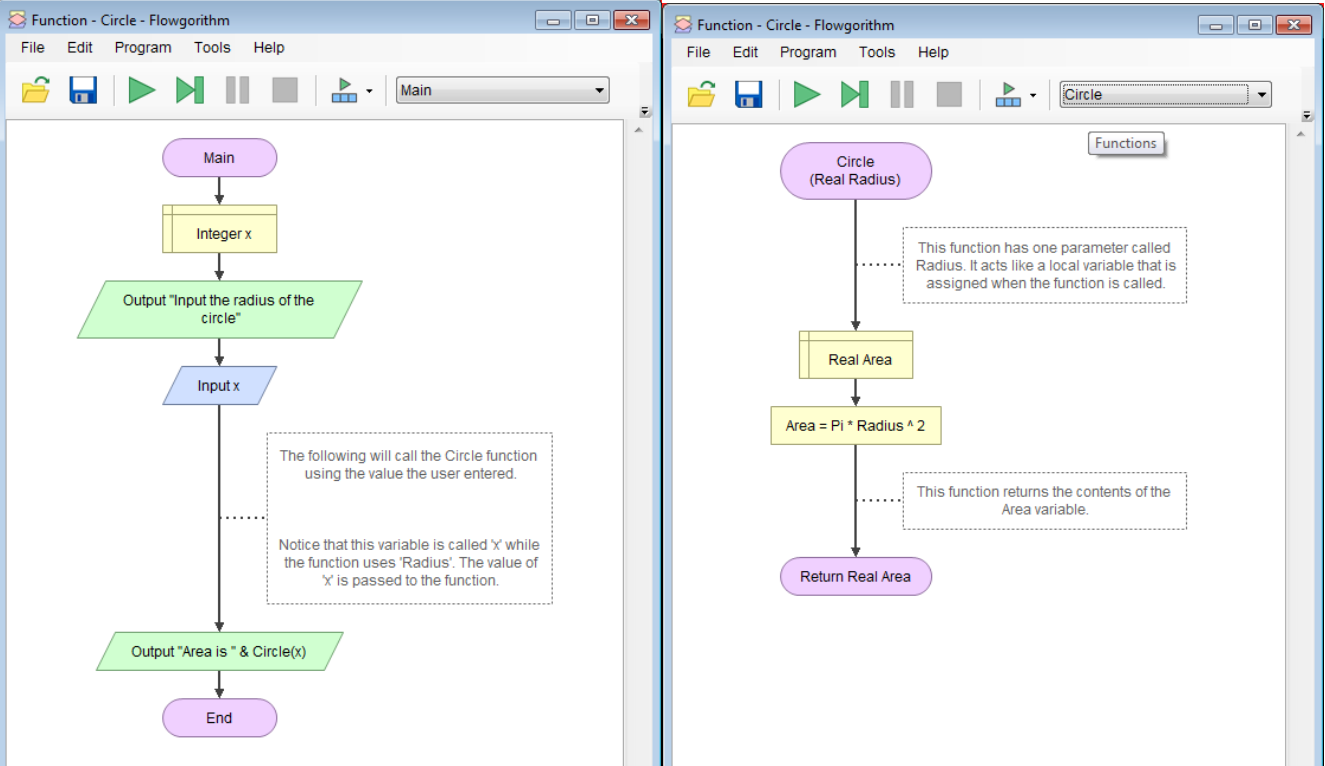


*a, Tính diện tích hình tròn*

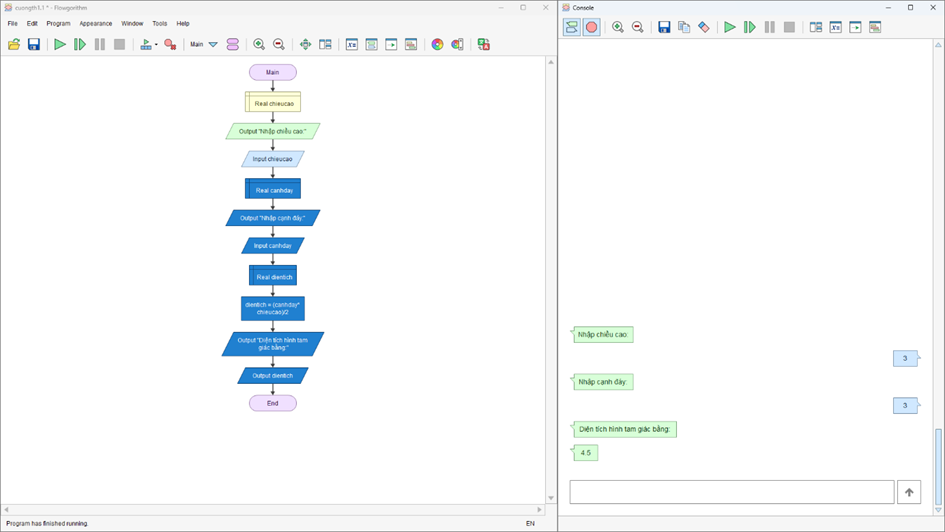


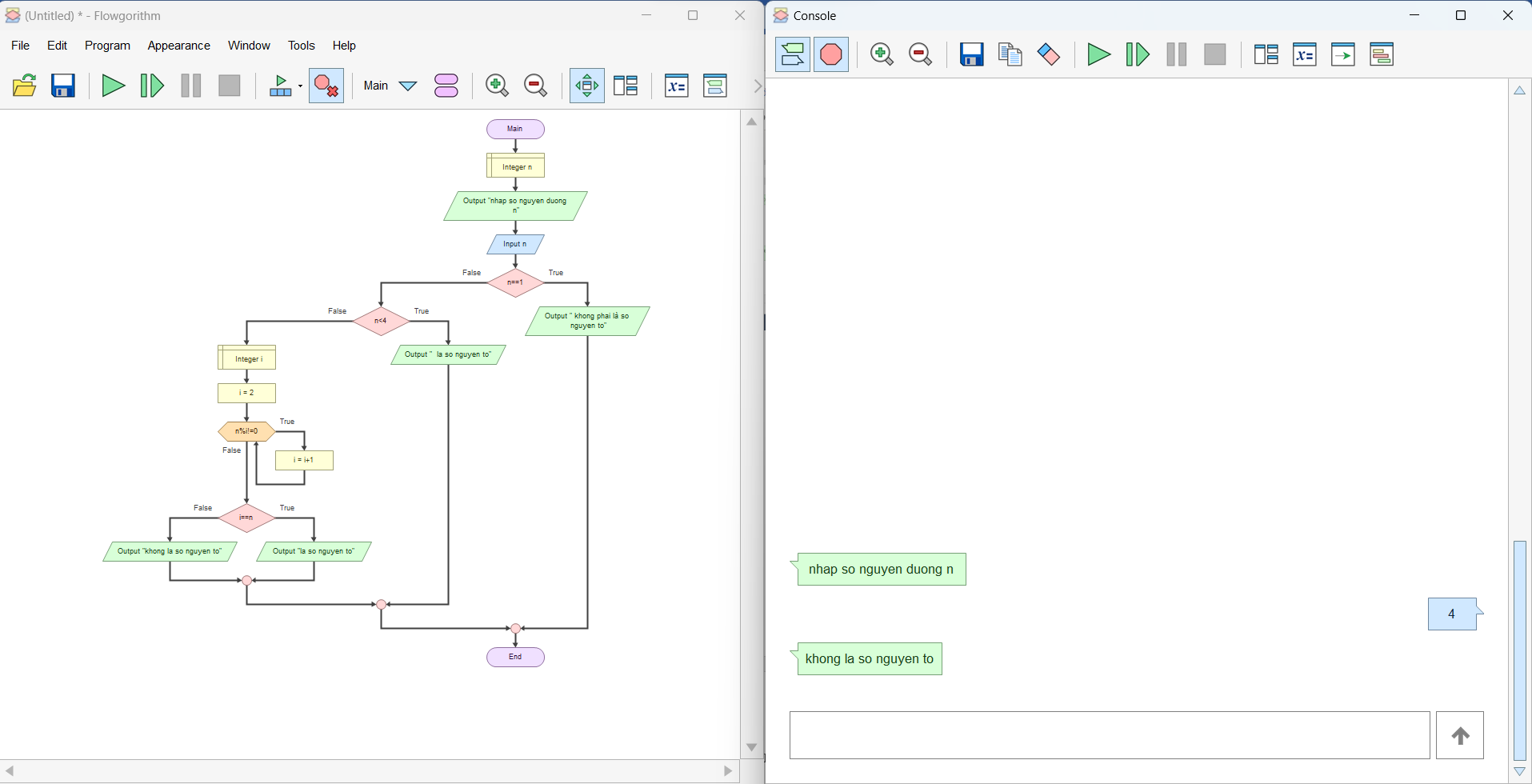
*b, Nhập và in số lượng chai sử dụng vong lặp for*

** c,Tính diện tích hình tròn sử dụng hàm ( chương trình con)

****

d, Viết chương trình nhập cạnh và chiều cao tương ứng của 1 tam giác và in ra màn hình diện tích của nó

****

e, Xây dựng thuật toán kiểm tra tính nguyên tố****

* 1. **Câu hỏi kiểm tra**
* Ý nghĩa các khối sử dụng trong chương chương
* Cách khai báo dữ liệu cho các biến ,các khối chức năng và vòng lặp
* Xây dựng hàm thực hiện chương trình con trong flowgorithm
  1. **Tài liệu tham khảo**

[**http://www.flowgorithm.org/documentation/ndex.htm**](http://www.flowgorithm.org/documentation/ndex.htm)

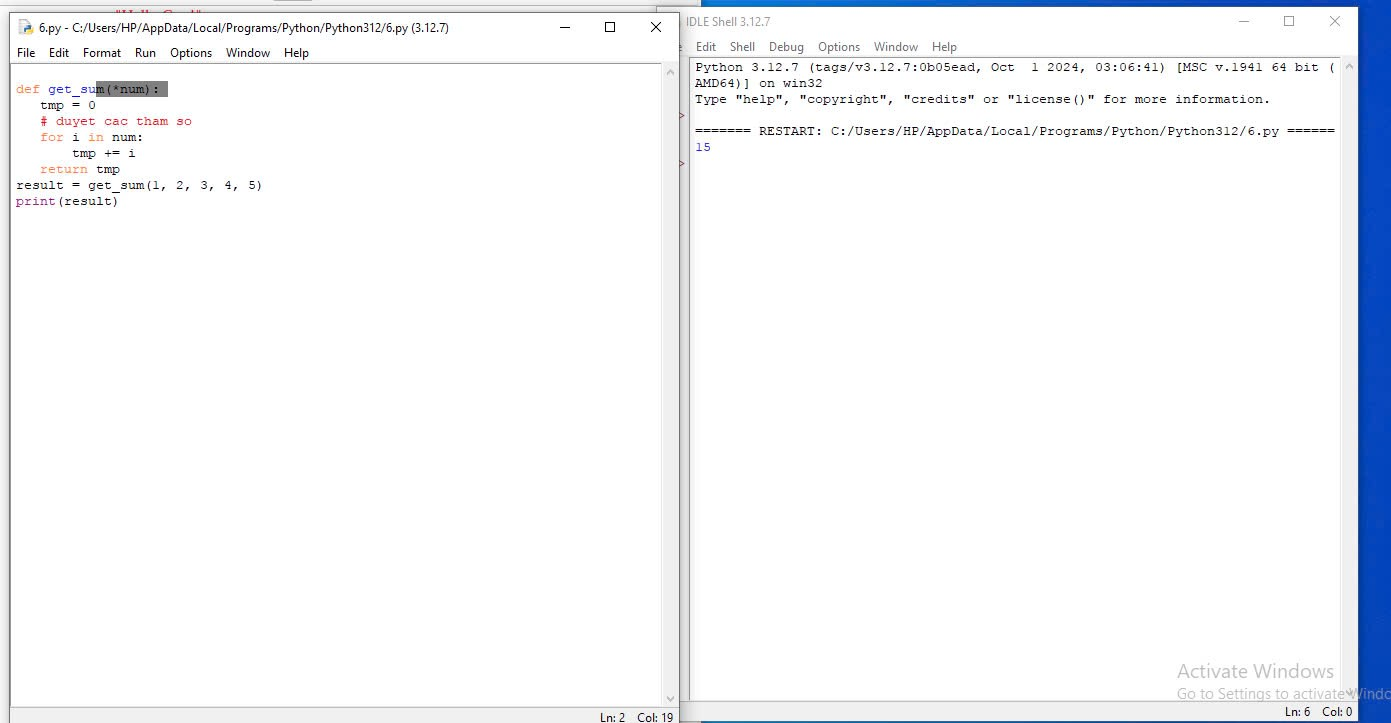
# **Bài 2. Các cú pháp, kiểu dữ liệu, lệnh điều khiển trong lập trình Python**

* 1. **Mục đích**

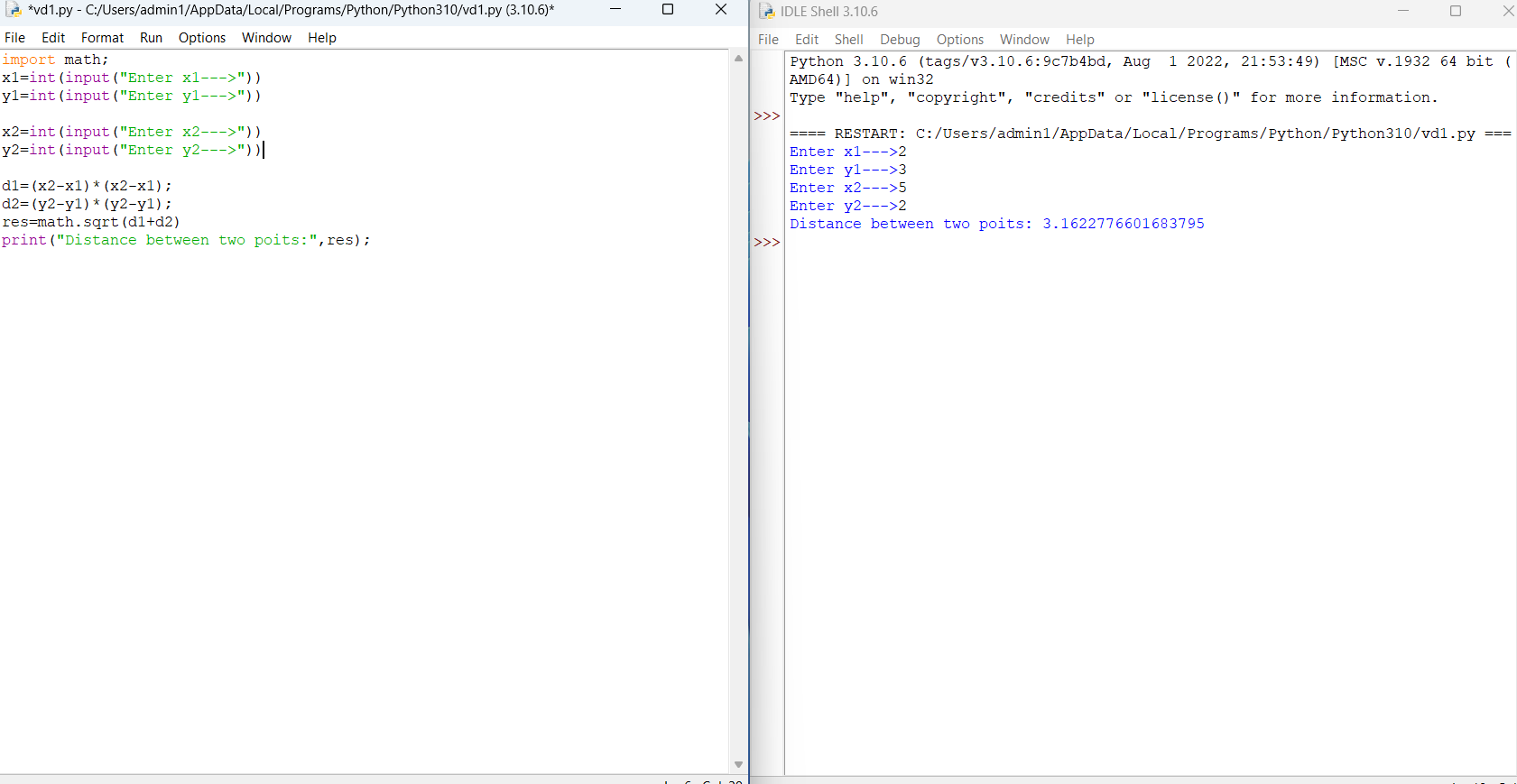
Giúp sinh viên nắm các kiến thức cơ bản như kiểu dữ liệu , các kiểu cú pháp, các lệnh điều khiển khi viết một chương trình bằng ngôn ngữ python.

* 1. **Các bước thực hiện và kết quả**

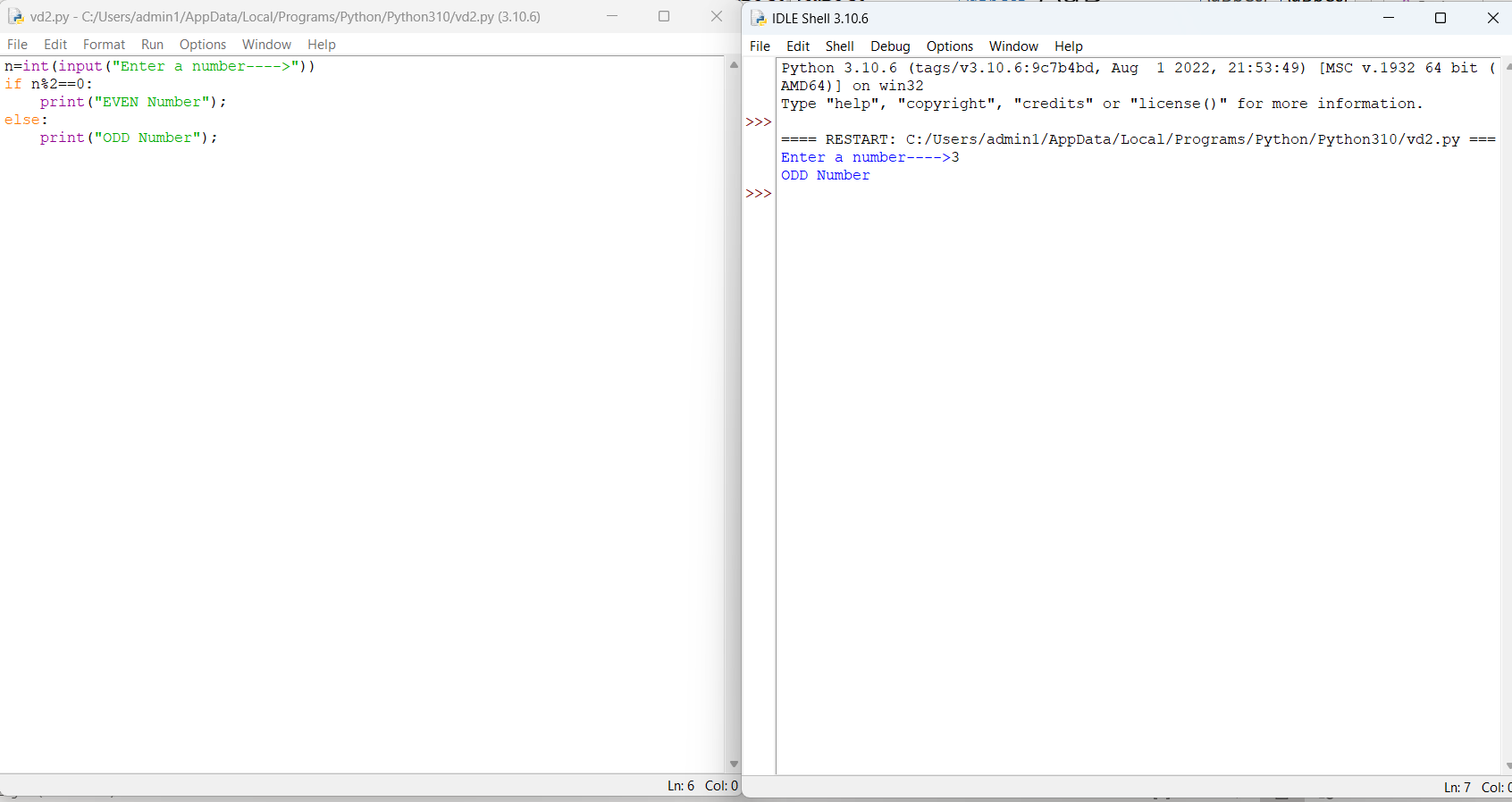
1. Viết đoạn chương trình sau và sữa lỗi

****

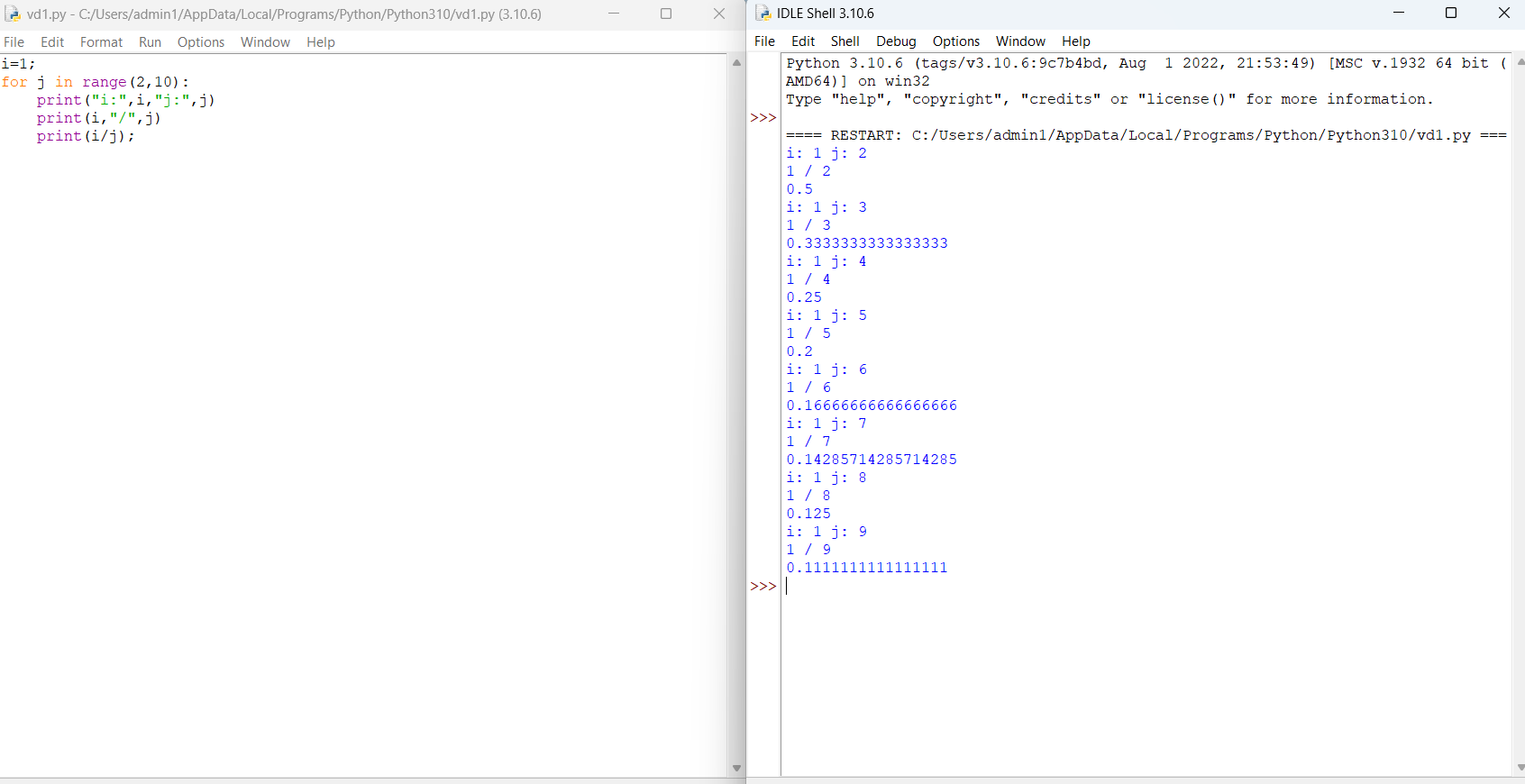
1. Viết chương trình nhập hai điểm và tính khoảng cách

****

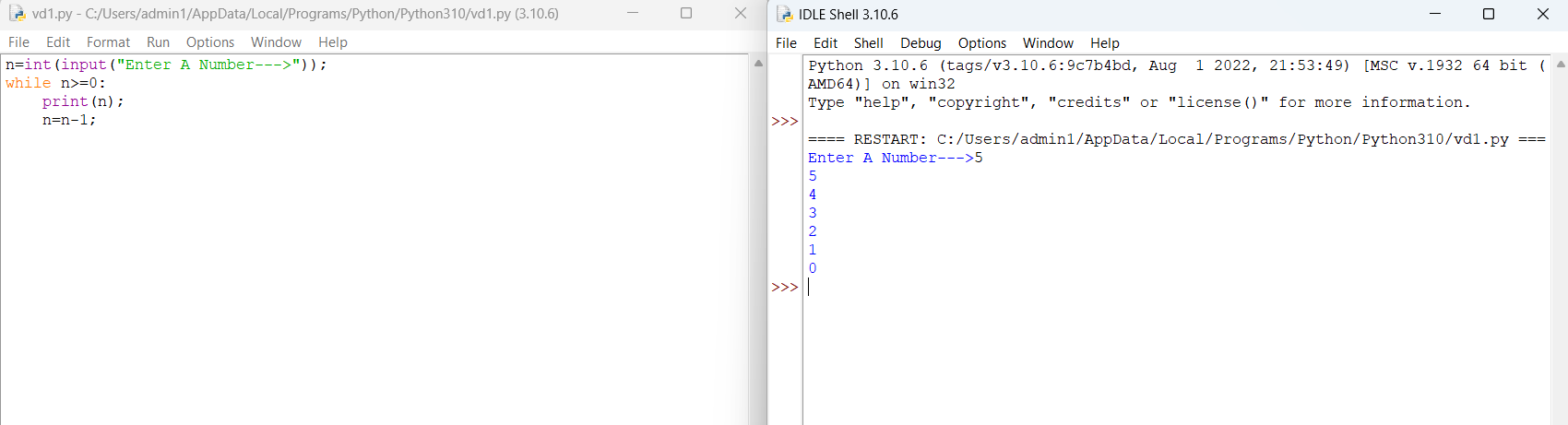
1. Viết chương trình nhập vào 1 số và kiểm tra số đó chẵn hay lẻ, in thông báo ra màn hình

****

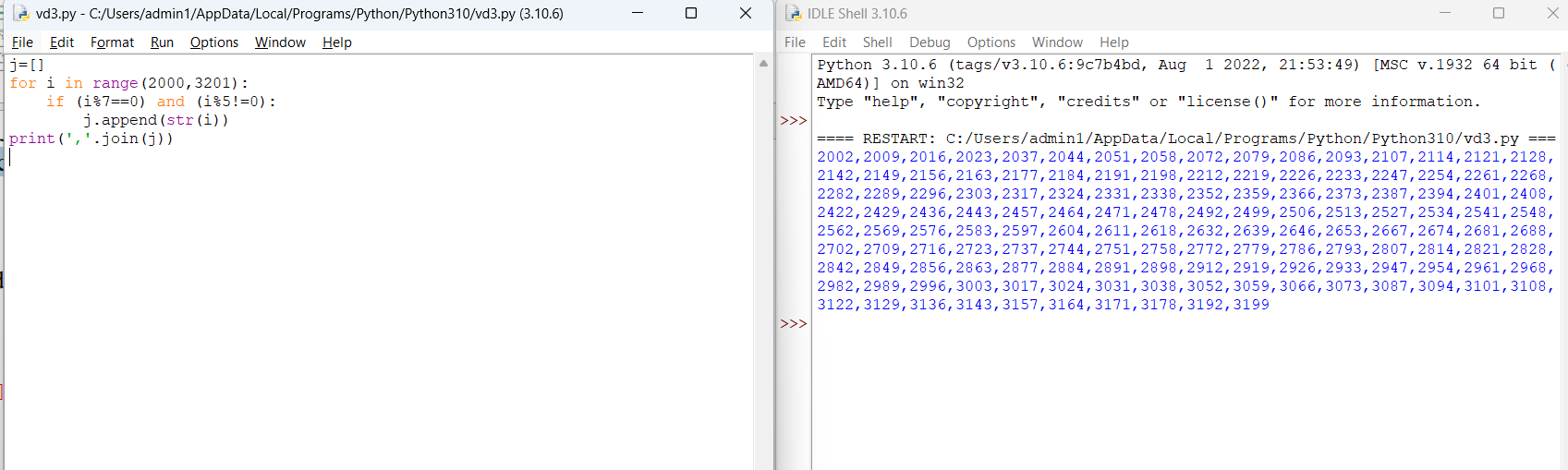
1. Viết chương trình in ra màn hình số nghịch đảo và kết quả dưới dạng thập phân của một dãy số tự nhiên trong khoảng (a,b)

****

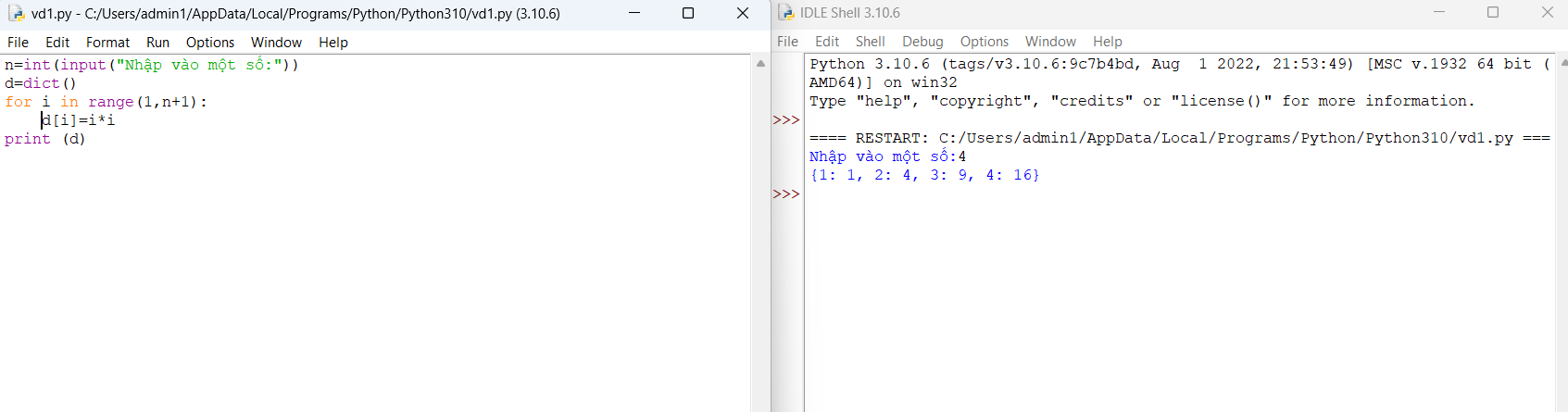
1. Viết chương trình nhập vào một số tự nhiên n>0, in vào màn hình các số tự nhiên giảm dần từ n đến 0, mỗi kí tự in trên 1 hàng

****

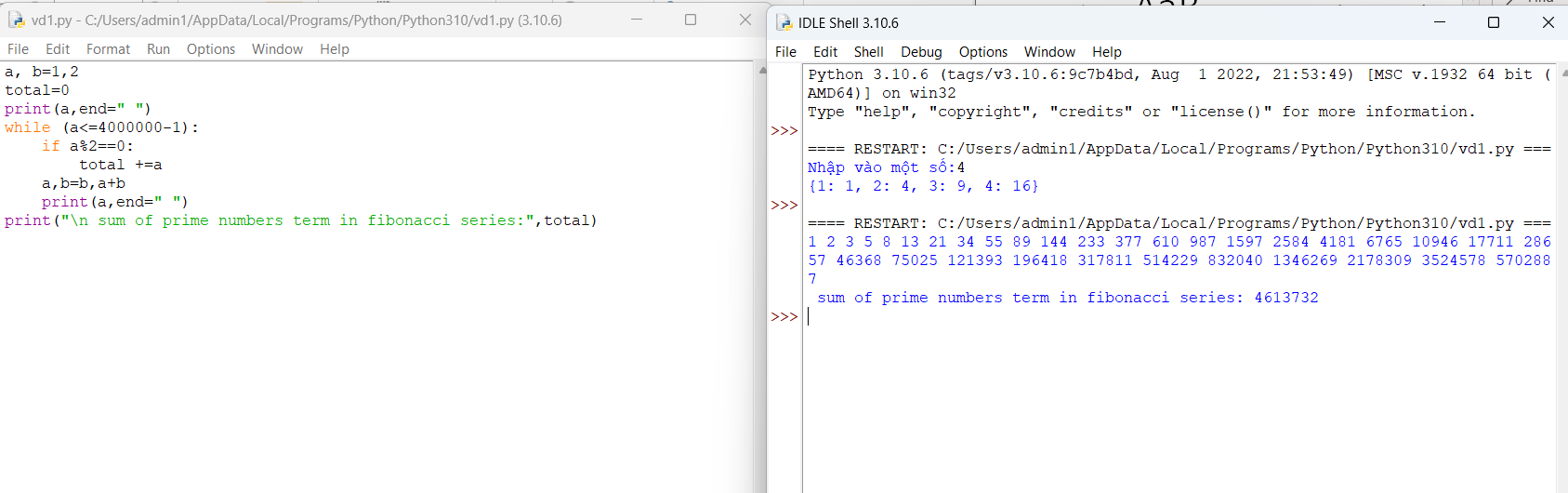
1. Viết chương trình tất cả số chia hết cho 7 nhưng không phải bội số của 5 nằm trong đoạn 2000 đến 3200( tính cả 2000 và 3200). Các số thu được sẽ được in thành chuỗi trên một dong , cách nhau bằng dấu phẩy

****

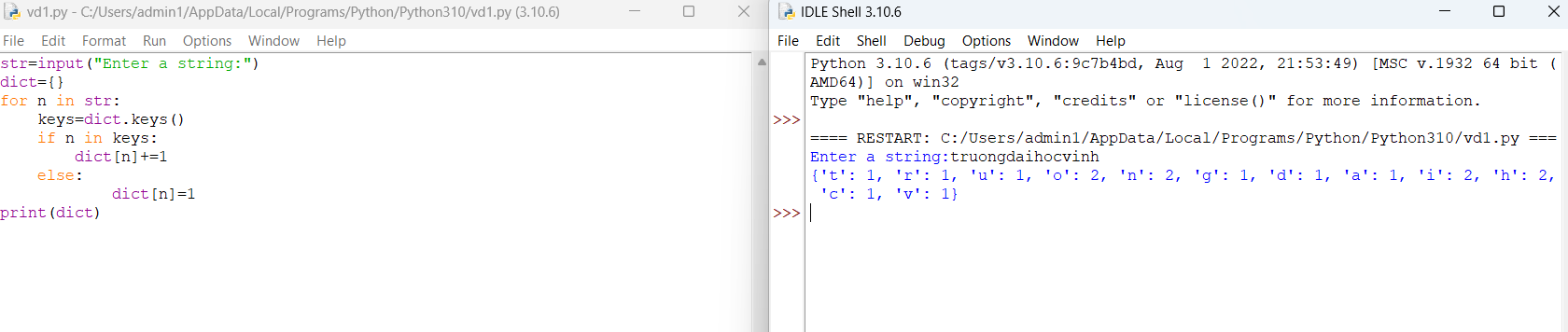
1. Với số nguyên n nhất định,hãy viết chương trình để tạo ra một dictionry chứa (i, i\*i) như là số nguyên từ 1 đến n ( bao gồm cả 1 và n ) sau đó in ra dictionry này

****

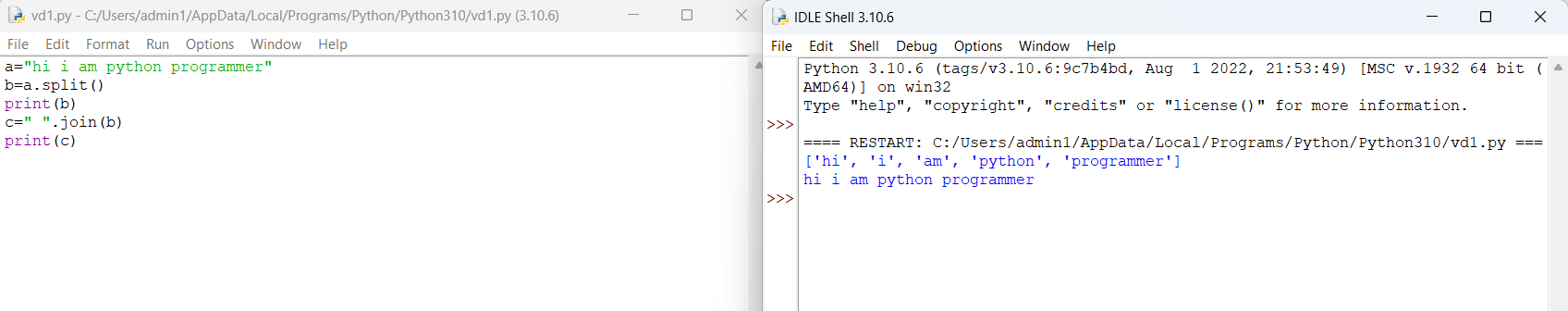
1. Viết chương trình in ra màn hình dãy số Fibonacci nhỏ hơn 4.000.000, tìm tổng các số chẵn trong dãy đã in

****

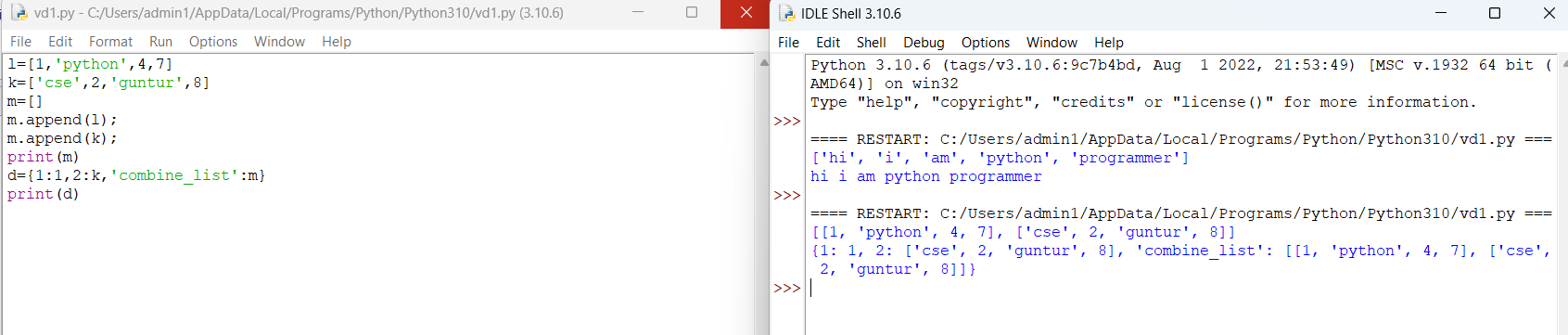
1. Viết chương trình đếm số ký tự trong 1 xâu ký tự nhập vào từ bàn phím, lưa các ký tự vào cấu trúc từ điển

****

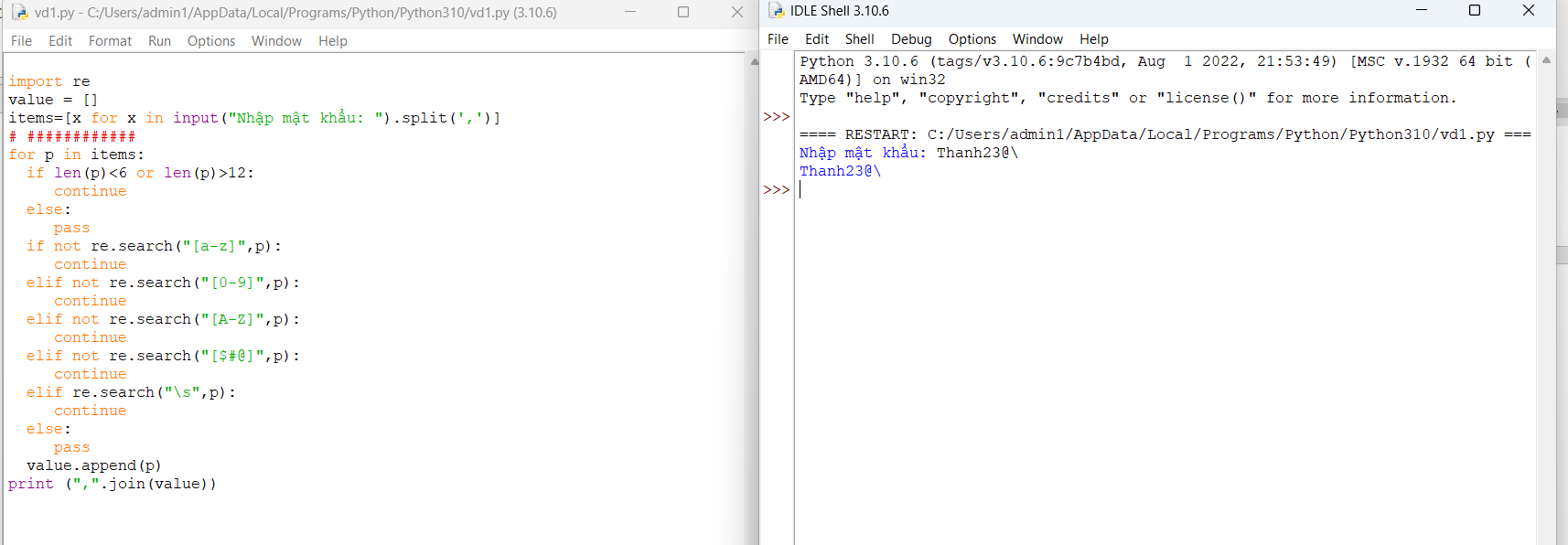
1. Viết chương trình sử dụng các phương thức split và join để tách nhập xâu ký tự

****

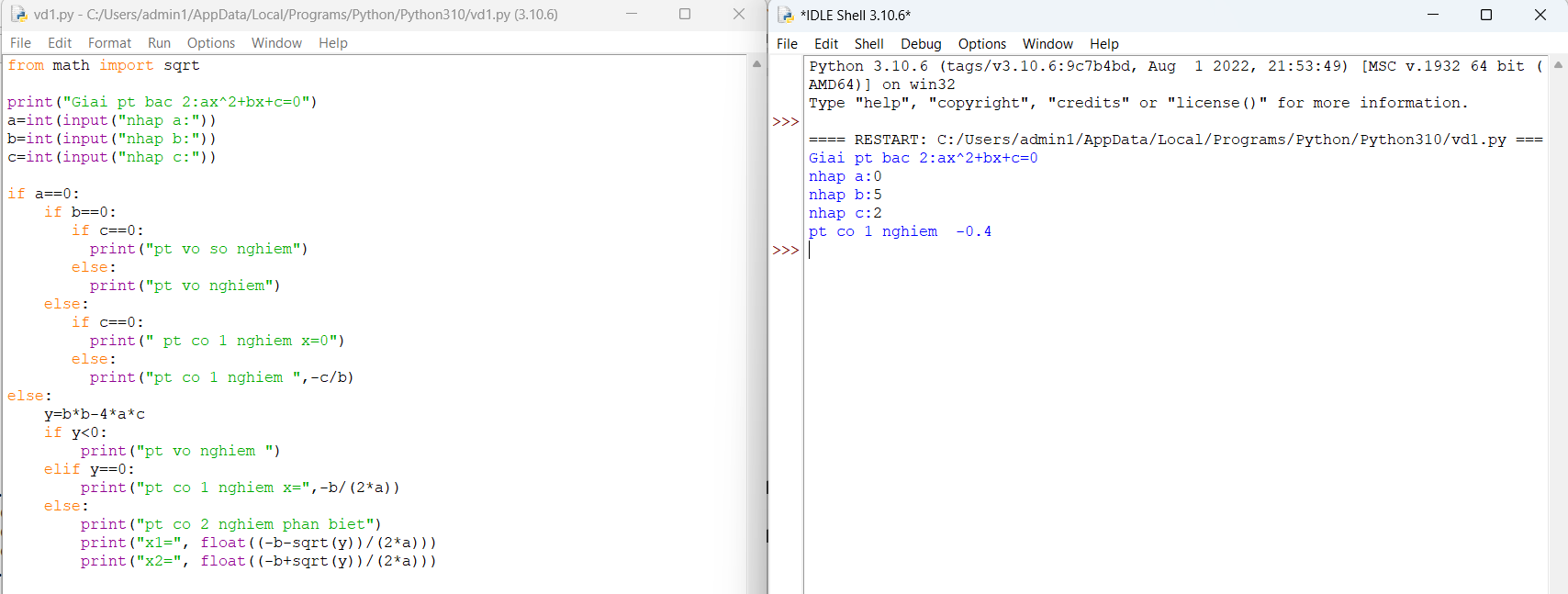
1. Viết chương trình kết nối các danh danh sách vào từ điển

****

1. 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.

****

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

****

* 1. **Câu hỏi kiểm tra**
* Các kiểu biến, khai báo và đặt tên biến trong python;
* Cấu trúc điều khiển trong python
* Giải thích các cú pháp lệnh và các phương thức sử dụng trong các chương trình
  1. **Tài liệu tham khảo**

Allen B.Downey, Think Python, o’Reilly Msdia, Inc, 2015.

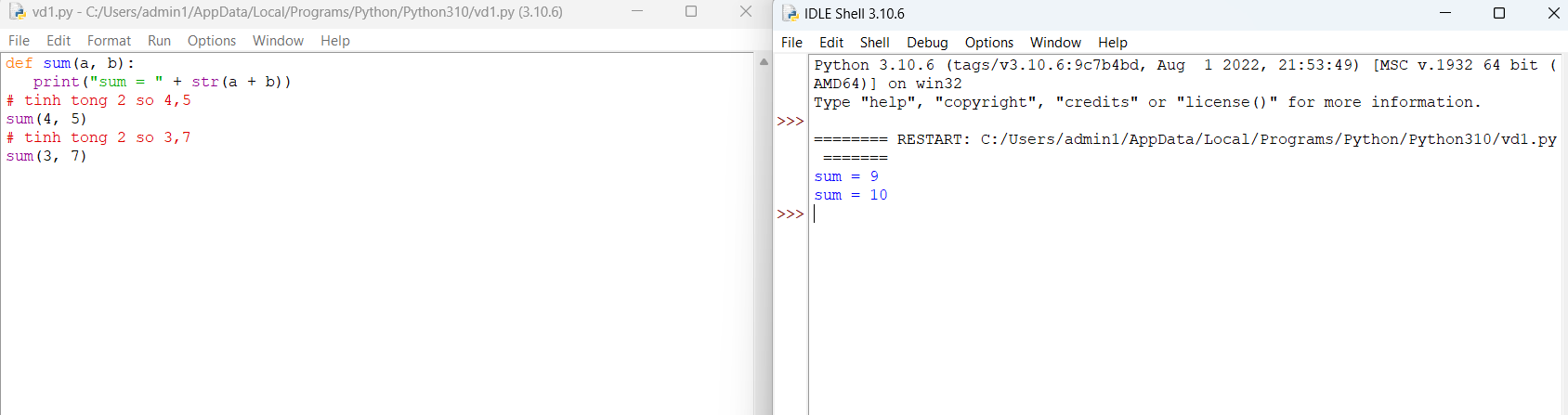
**Bài 3. Lập trình hàm trong Python**

* 1. **Mục đích**

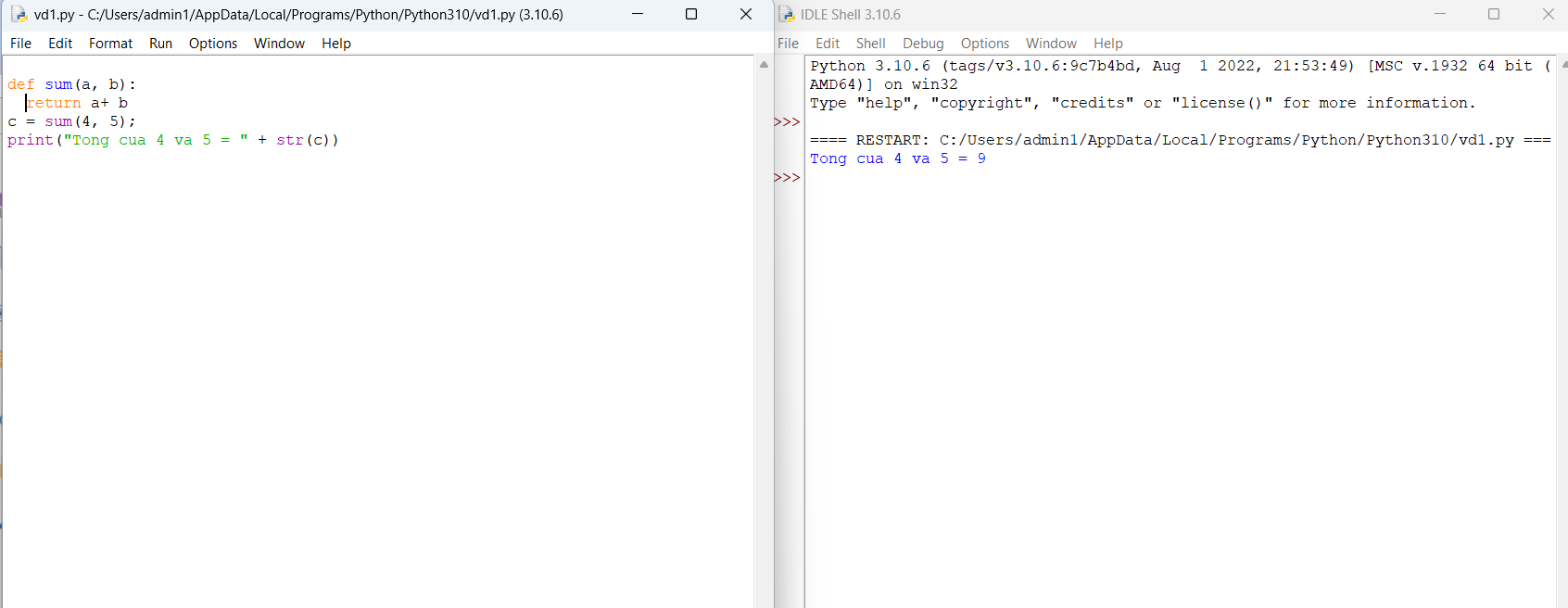
Giúp sinh viên nắm bắt các kiến thức trong lập trình python sử dụng hàm.

* 1. **Các bước thực hiện và kết quả.**

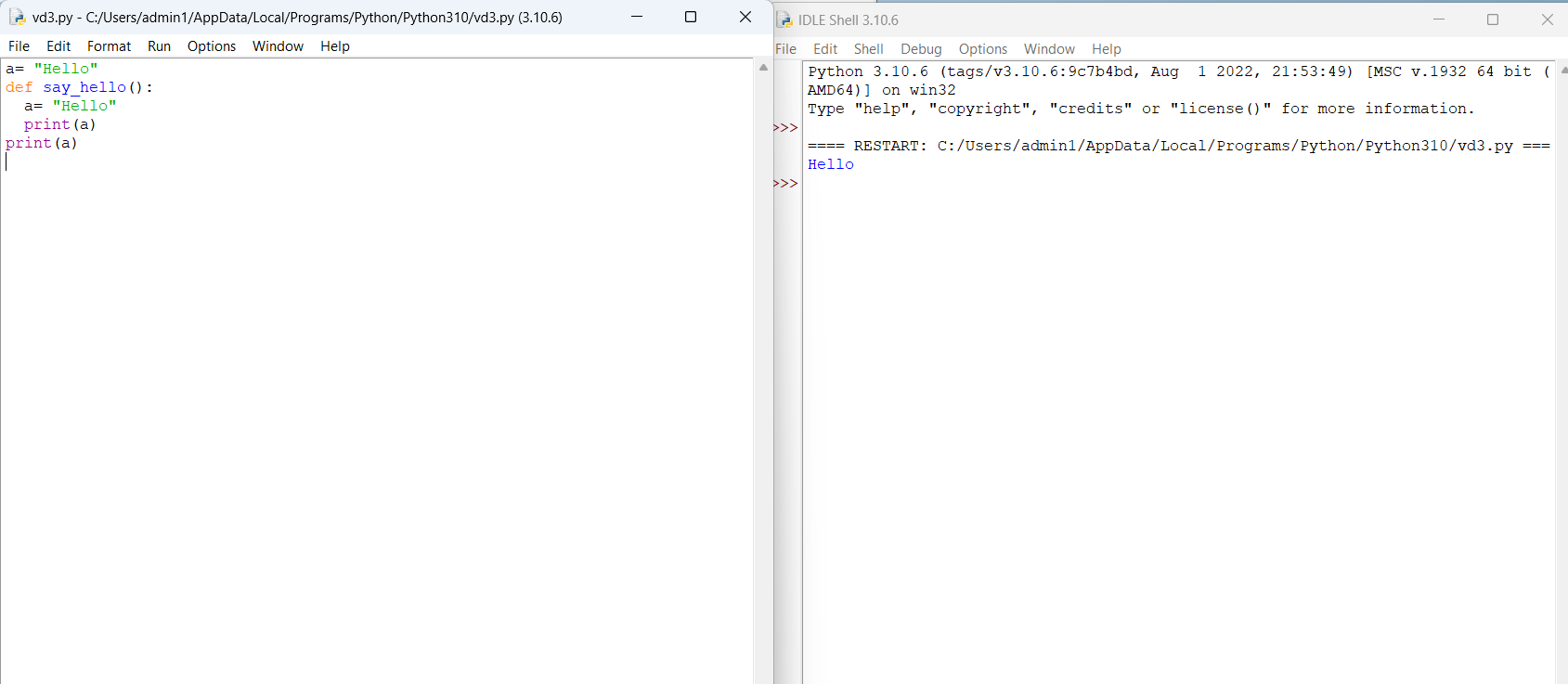
1. Viết hàm sum( ) tính tổng hai số

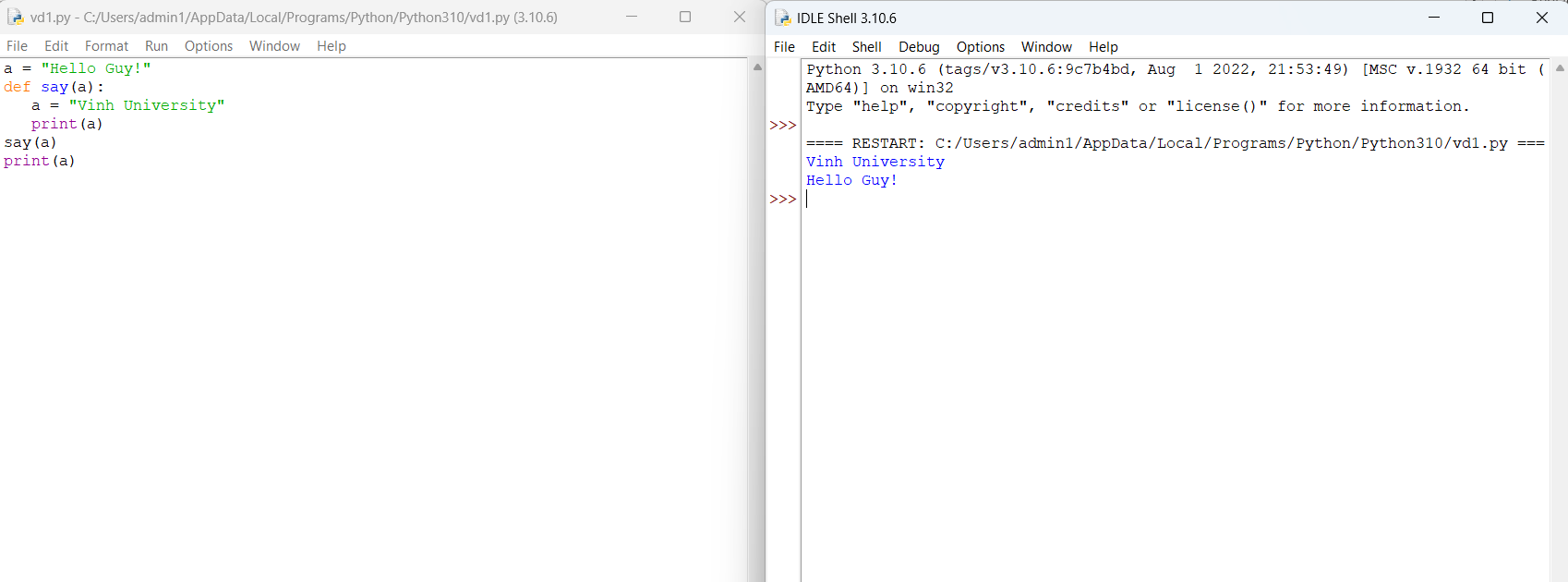


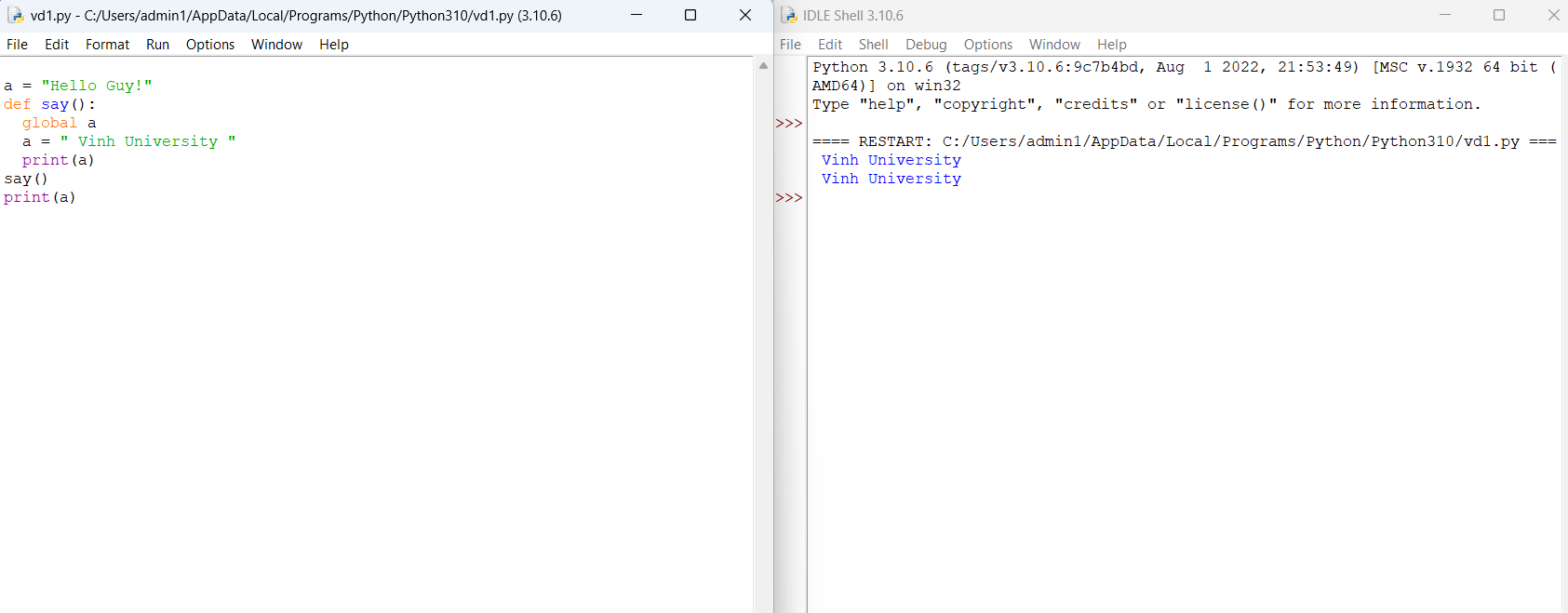
1. Viết hàm sum( ) với kết quả trả về

****

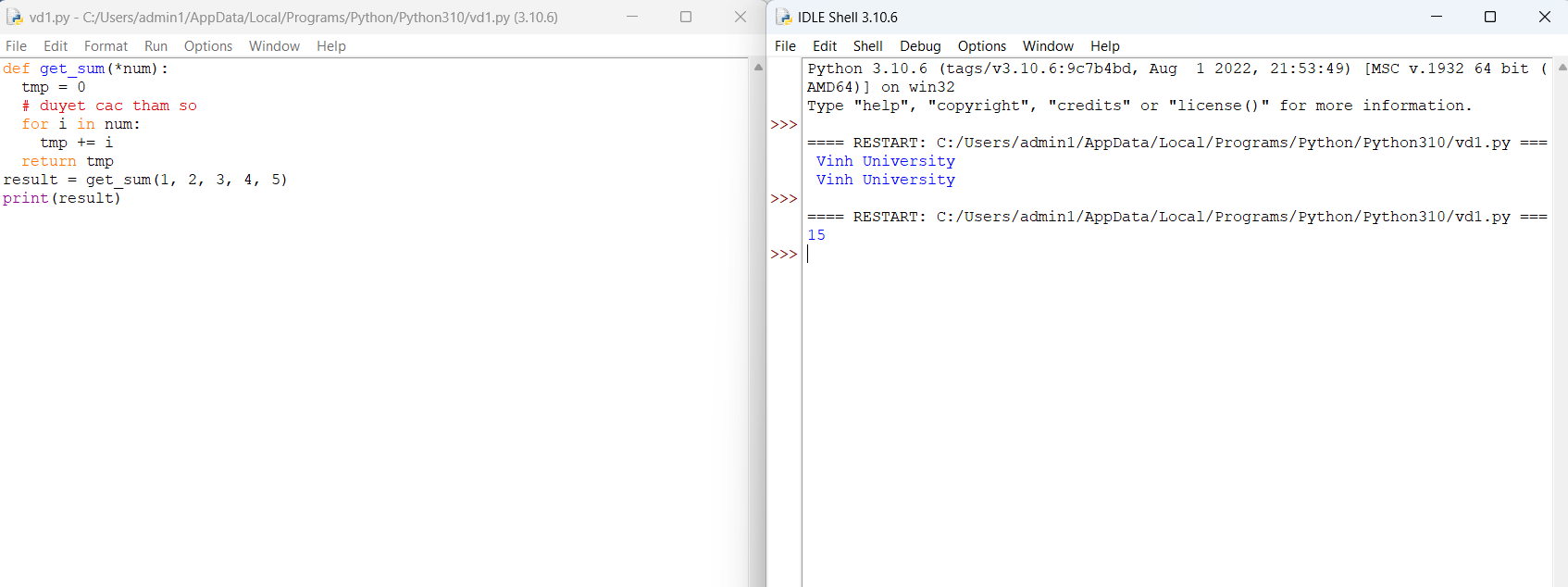
1. Tìm và sữa lỗi chương trình

****

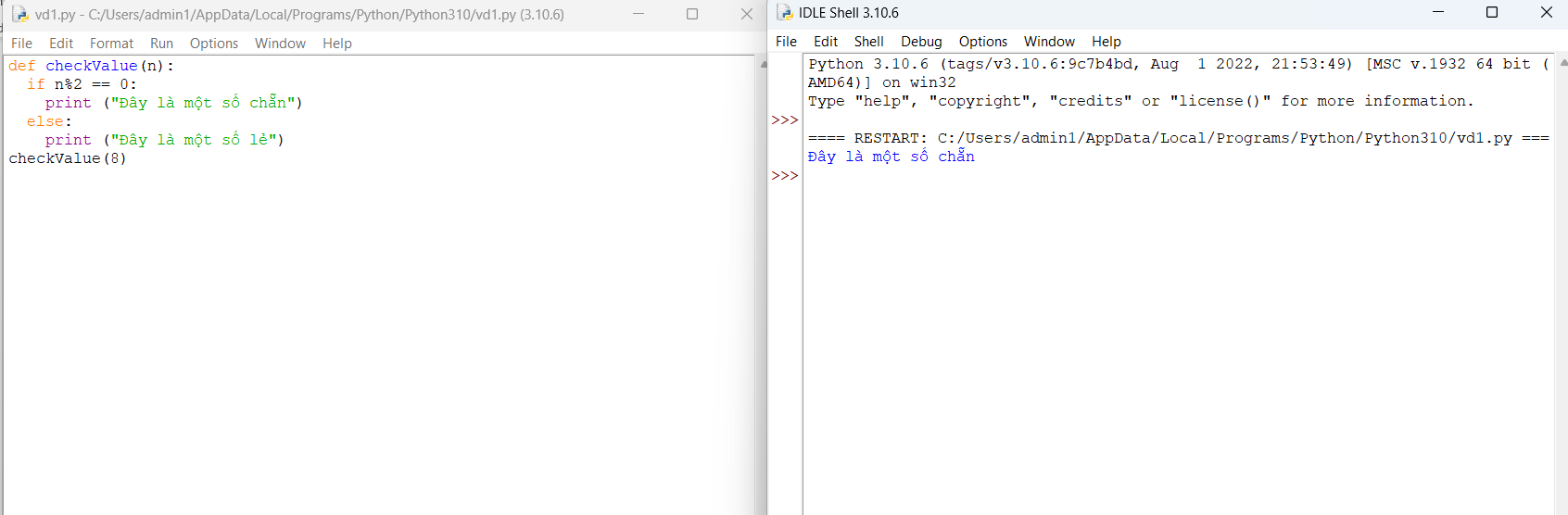
1. Viết chương trình có phạm vi biến như sau
2. Viết chương trình sau và xem sự thay đổi của biến



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



1. 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.



1. 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:

UP 5

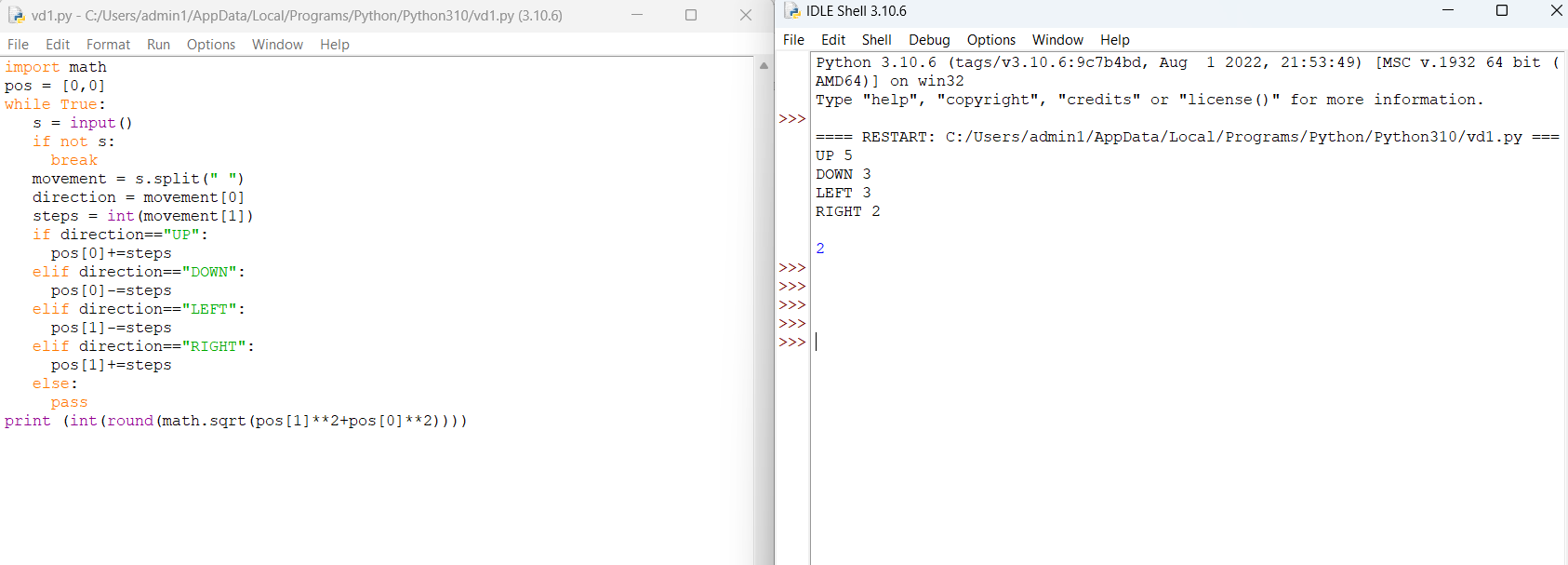
DOWN 3

LEFT 3

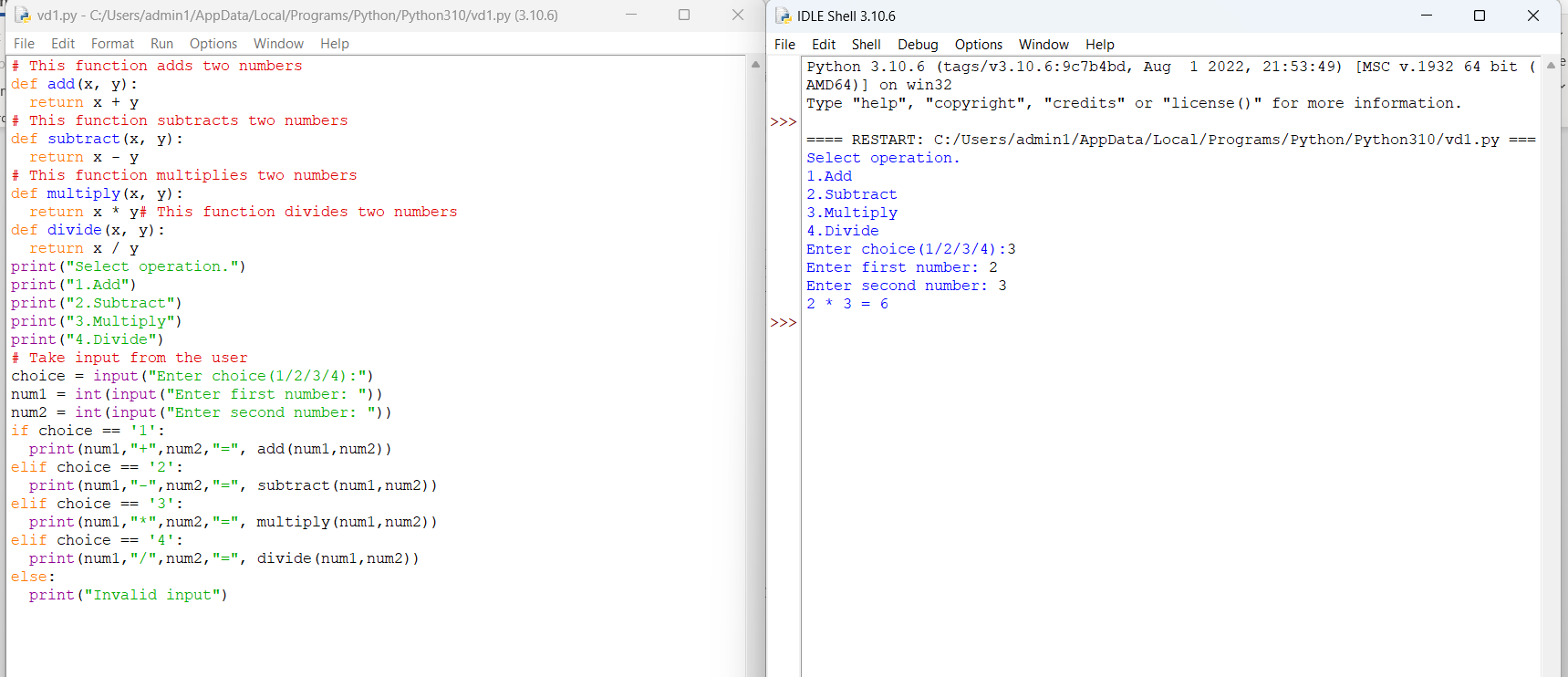
RIGHT 3

Các con số sau phía sau hướng di chuyển chính là số bước đi. Hãy viết chương

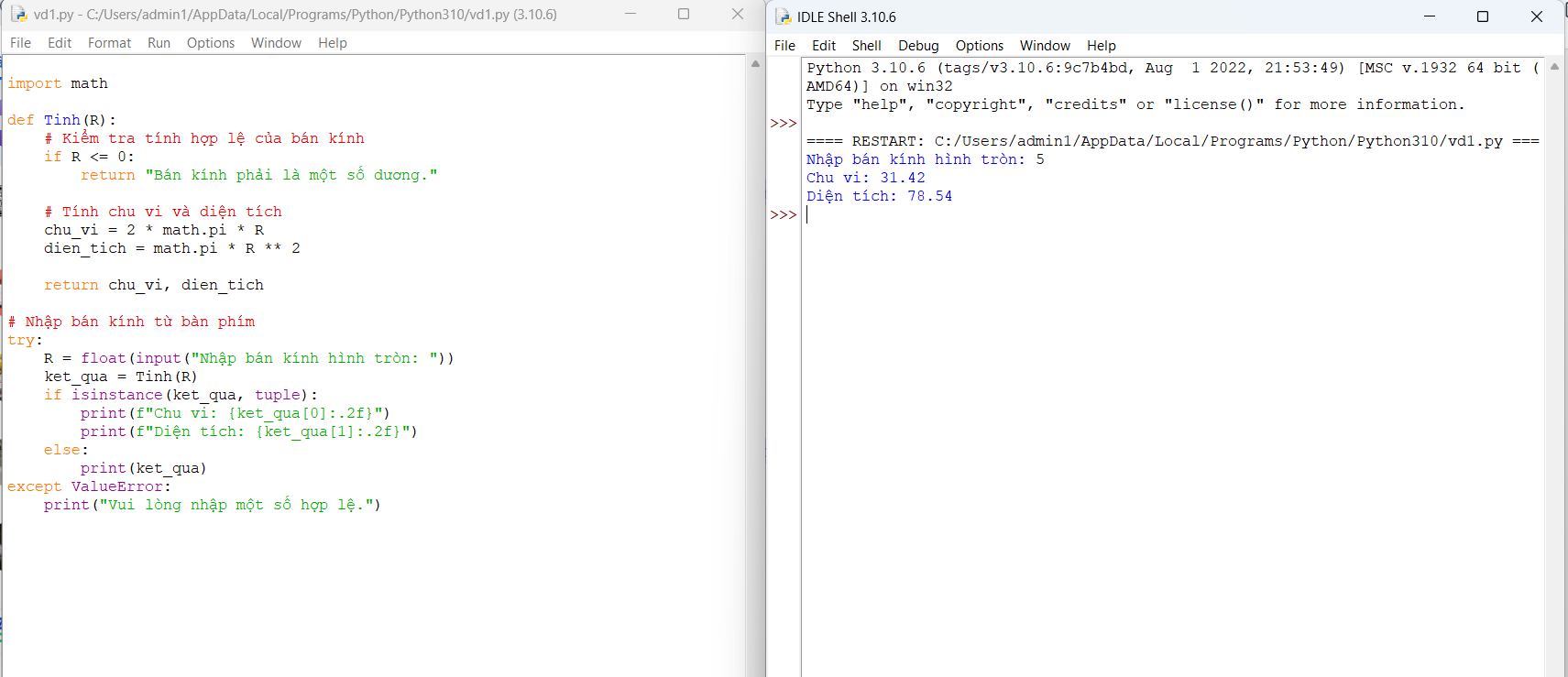
trình để tính toán khoảng cách từ vị trí hiện tại đến vị trí đầu tiên, sau khi robot đã di chuyển một quãng đường. Nếu khoảng cách là một số thập phân chỉ cần in số nguyên gần nhất.



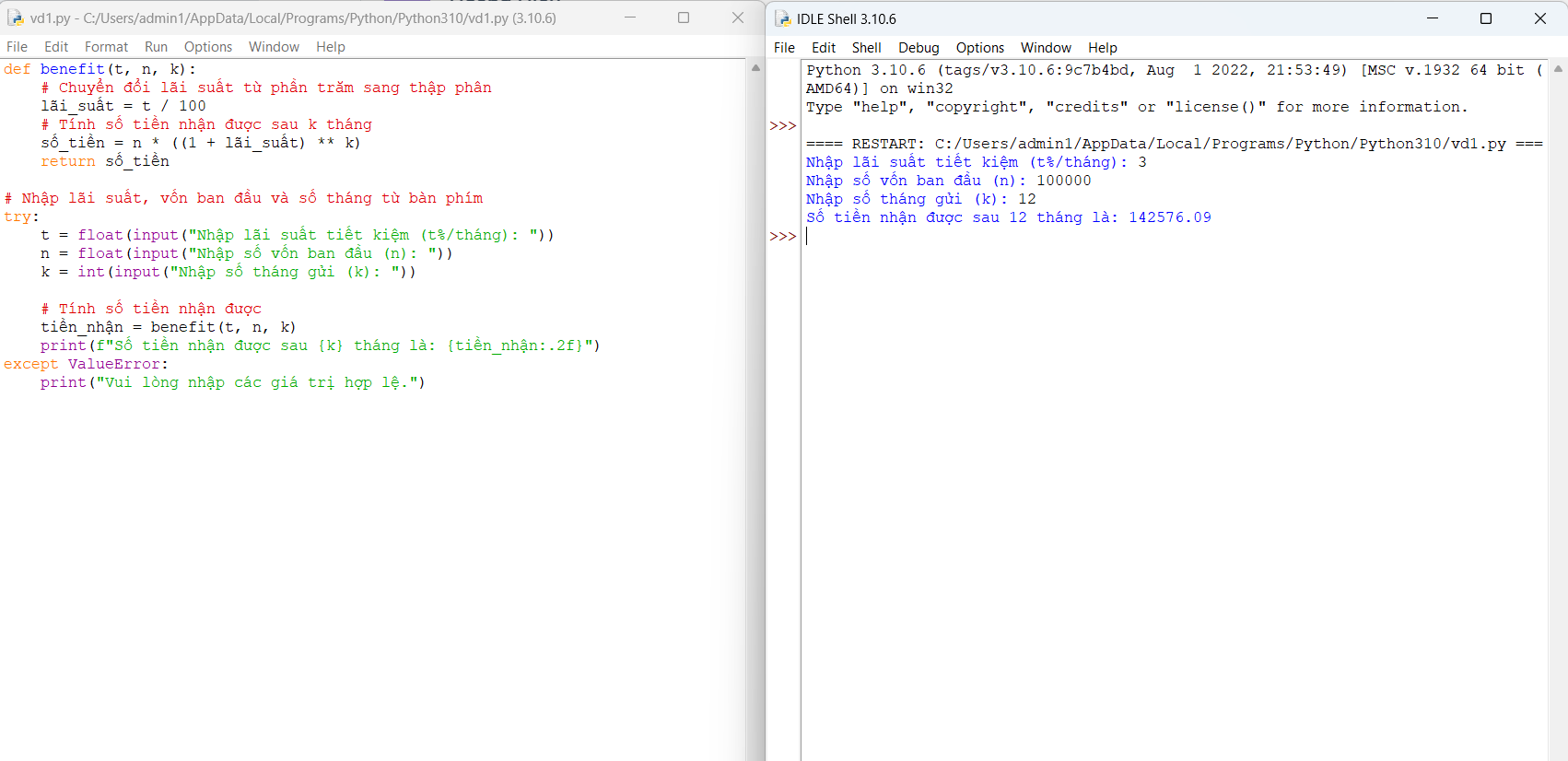
1. Chương trình máy tính thực hiện các phép tính đơn giản



1. Viết hàm “def Tinh(R):” tính chu vi và diện tích hình tròn, với bán kính R được nhập từ bàn phím, và kiểm tra giá trị bán kính đầu vào là hợp lệ.



1. 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 sử dụng cấu trúc hàm def benefit(t,n,k):



* 1. **Câu hỏi kiểm tra**

- Giải thích quy tắc khai báo hàm, gọi hàm, giá trị trả về, tham số truyền vào, tham số mặc định, phạm vi của biến trong python, sử dụng các hàm có sẵn trong các thư viện của python trong các chương trình đã thực thi

**1.6. Tài liệu tham khảo**

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

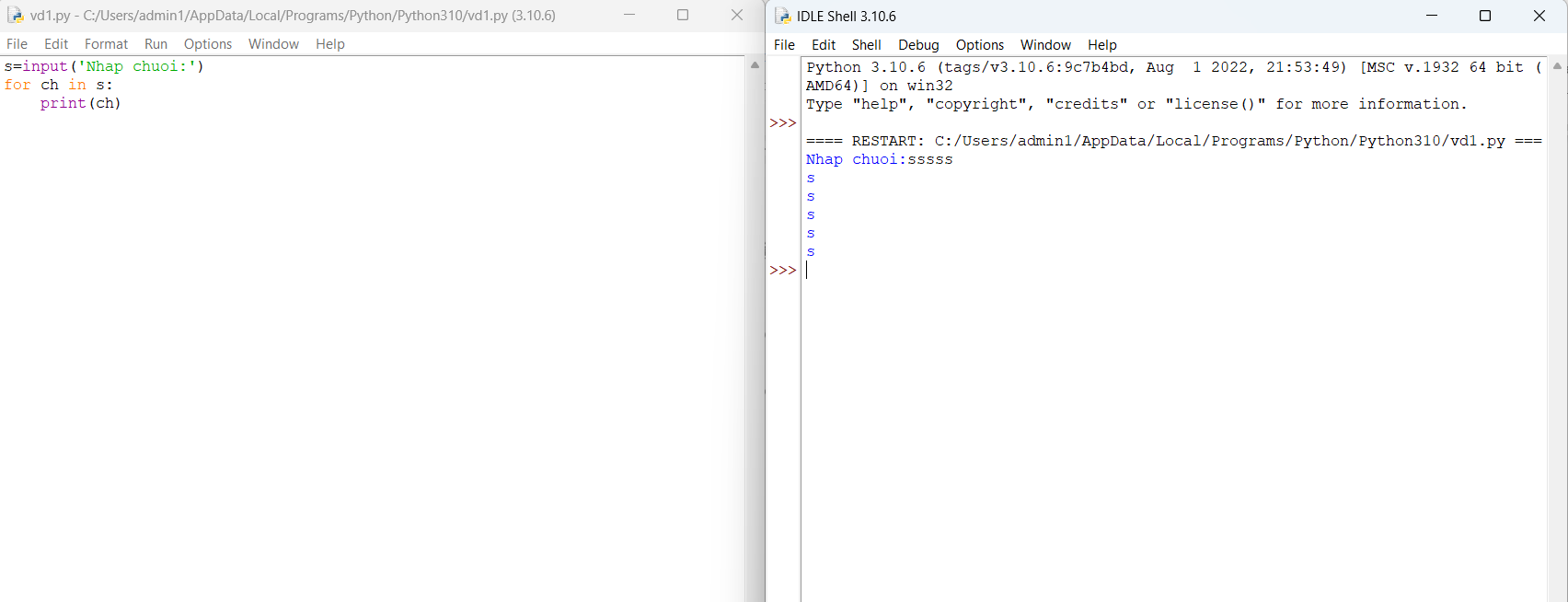
**Bài 4. Các kiểu dữ liệu có cấu trúc trong Python**

* 1. **Mục đích**

Giúp sinh viên nắm bắt các kiểu dữ liệu có cấu trúc trong lập trình python.

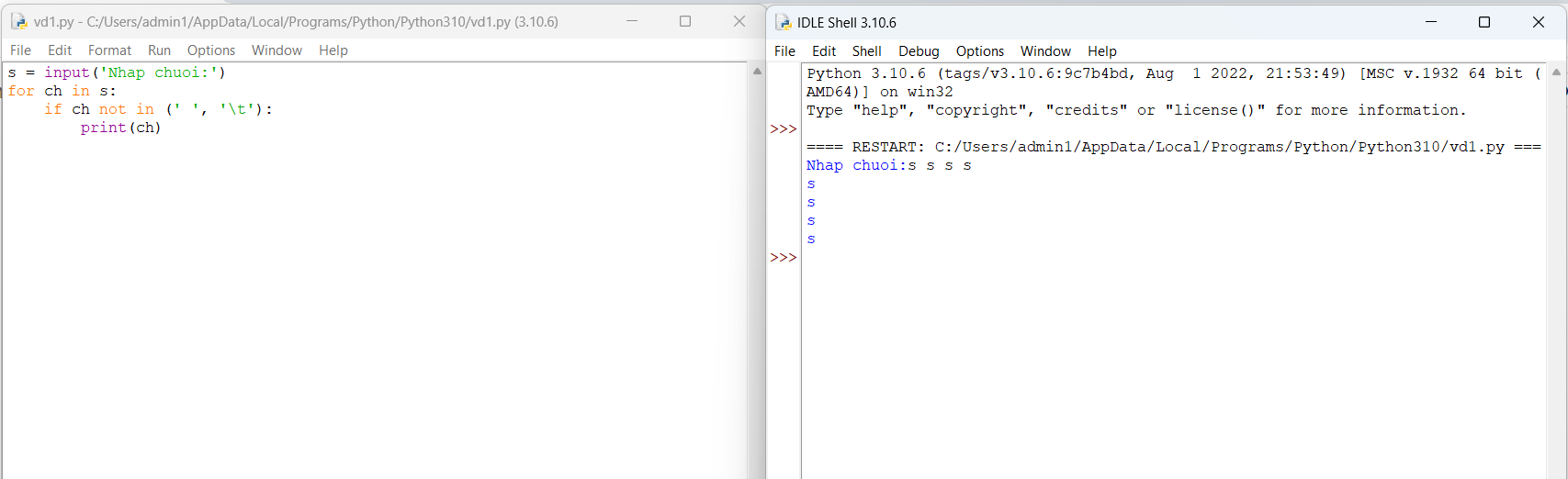
* 1. **Các bước tiến hành và kết quả**

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

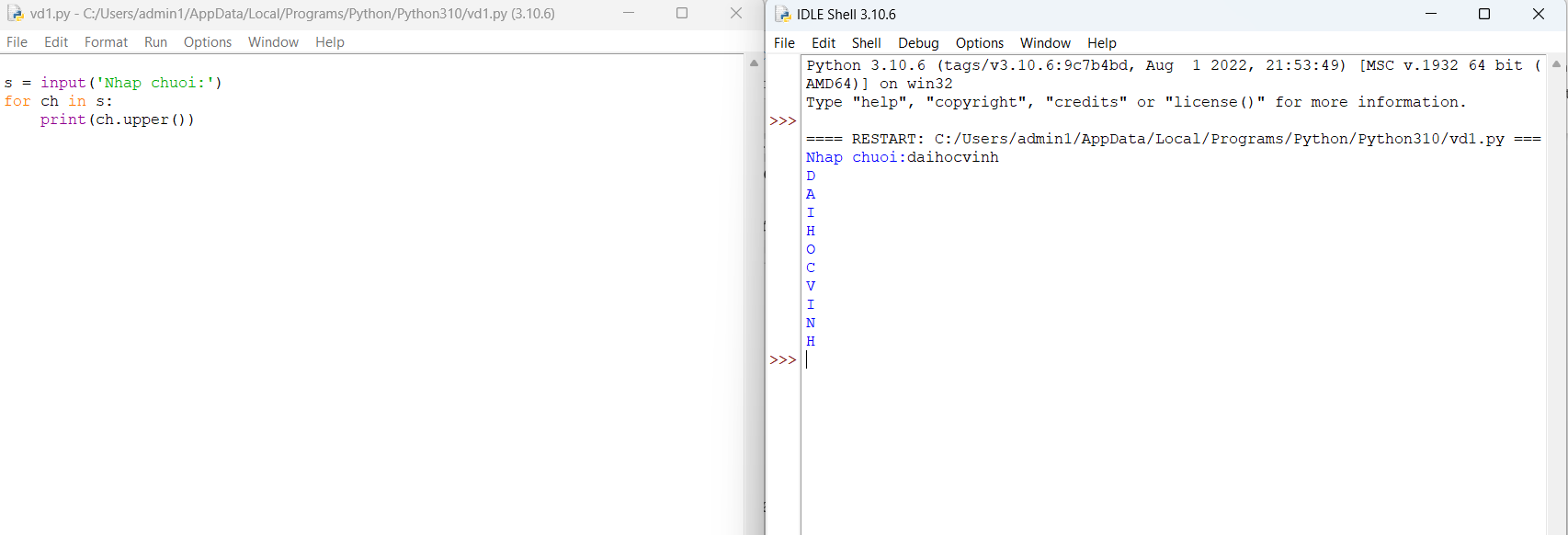


1. Chỉnh sửa ví dụ trên: hãy bỏ qua không in ra những kí tự “không nhìn thấy” (dấu

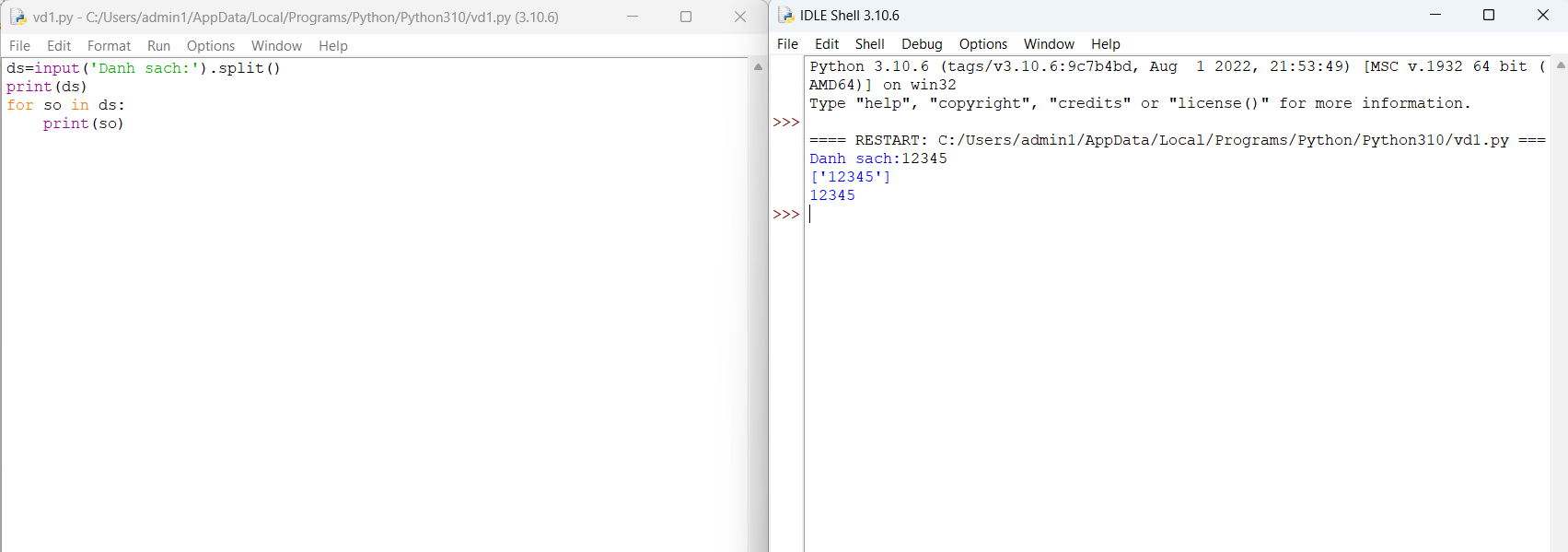
space và dấu tab).



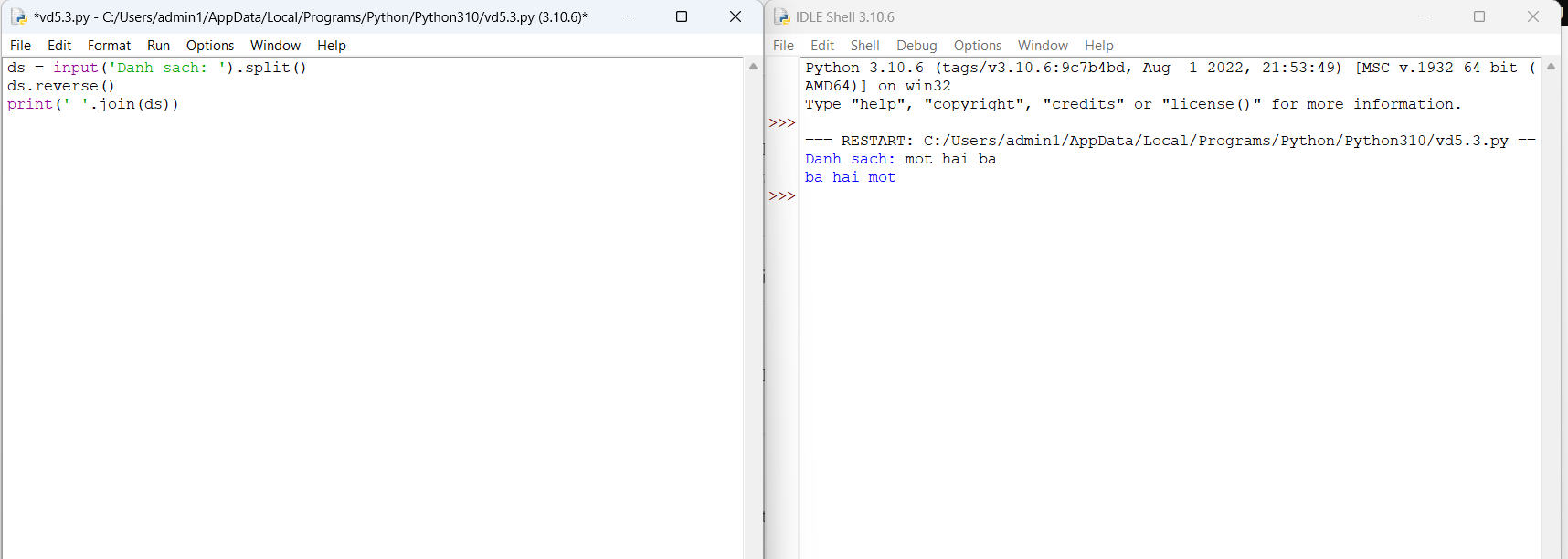
1. Chỉnh sửa ví dụ ở bài 1: hãy các kí tự ở dạng IN HOA.



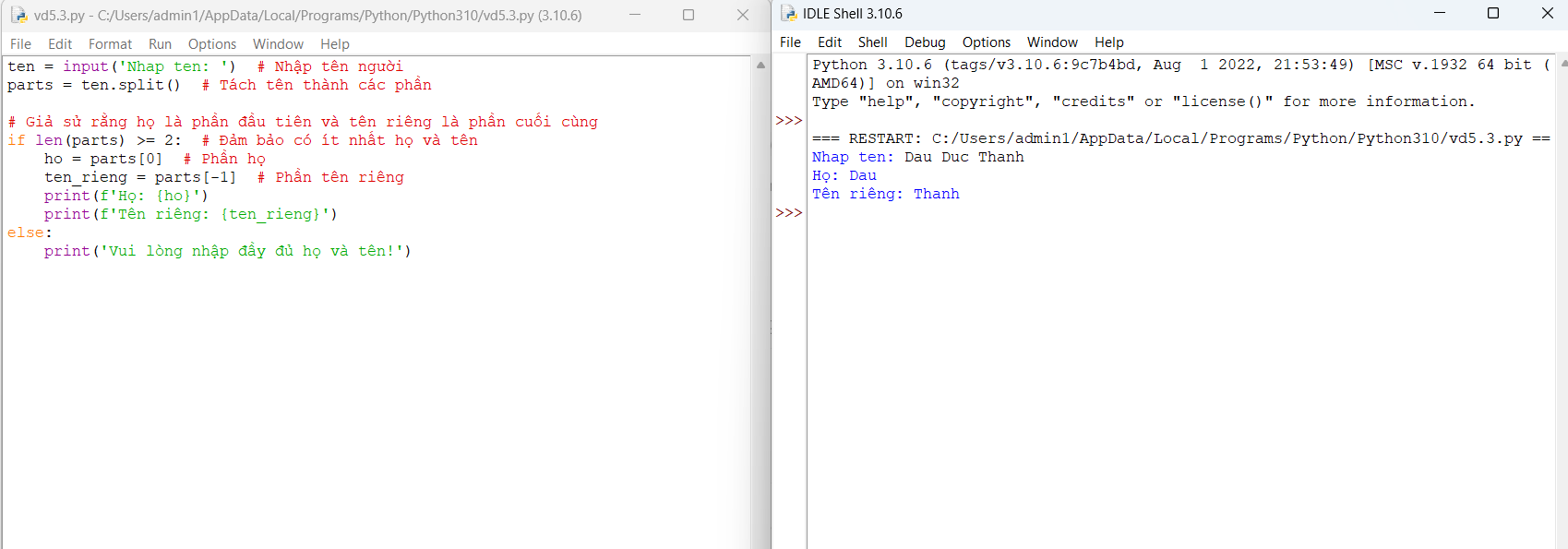
1. Nhập một danh sách trên một dòng, mỗi phần tử cách nhau bởi dấu trống hoặc tab, sau đó in ra dãy vừa nhập ra màn hình



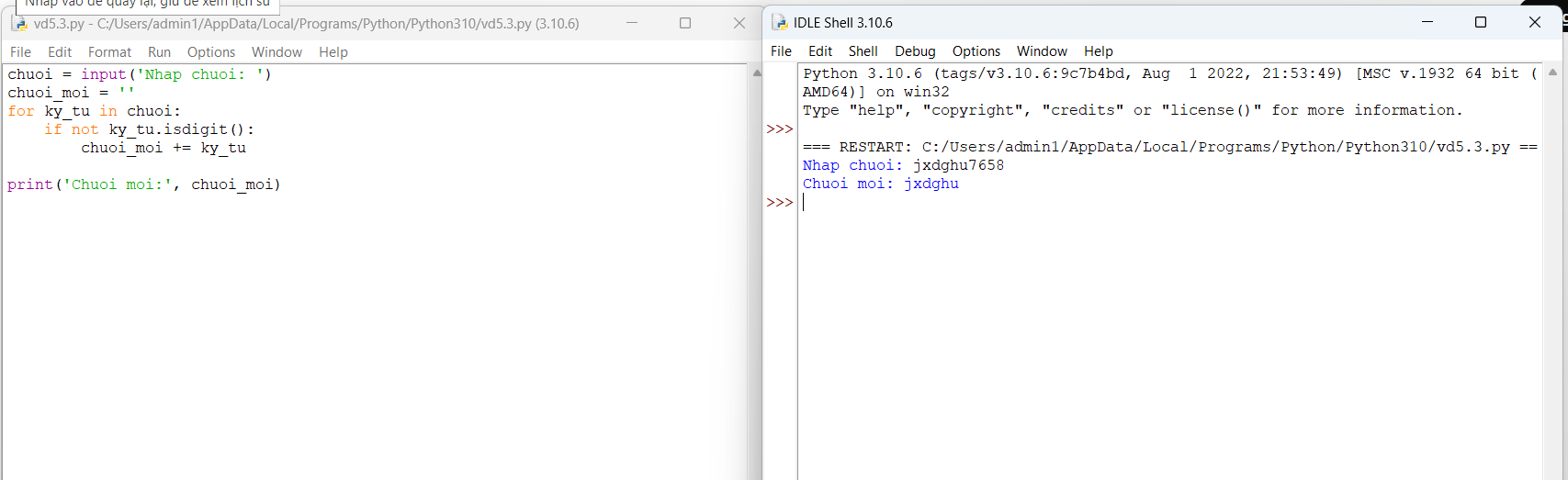
1. Chỉnh sửa ví dụ ở bài 4: nhập 1 danh sách các từ từ bàn phím, in ra các từ đó theo thứ tự ngược lại thứ tự vừa nhập (ví dụ nhập “mot hai ba” thì in ra “ba hai mot”)



1. 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).

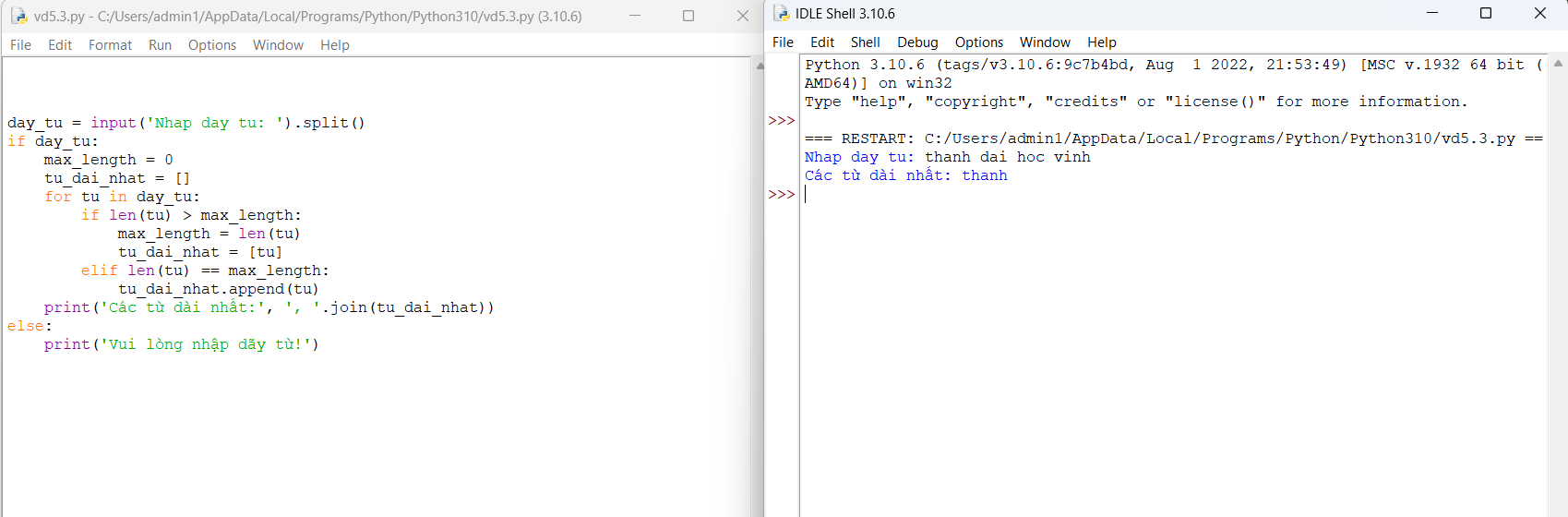


1. Nhập một chuỗi từ bàn phím, hãy loại bỏ tất cả các chữ số khỏi chuỗi và in lại nội dung chuỗi mới ra màn hình.

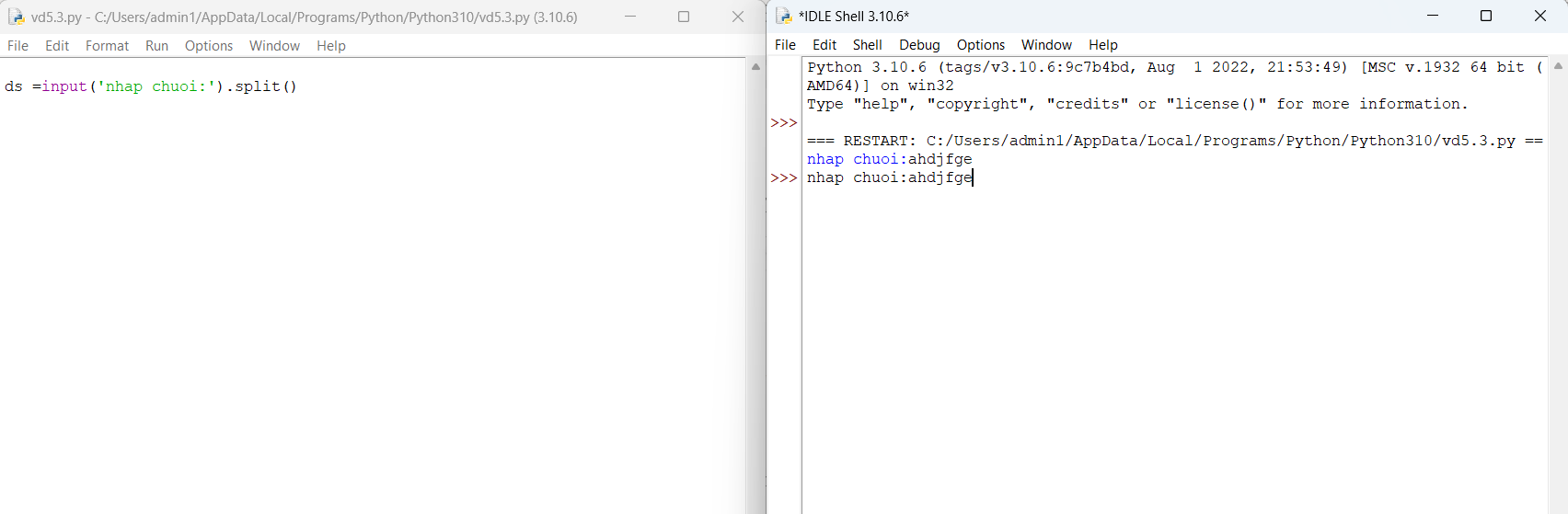


1. Nhập một dãy các từ từ bàn phím, hãy in ra từ dài nhất trong dãy vừa nhập, in ra

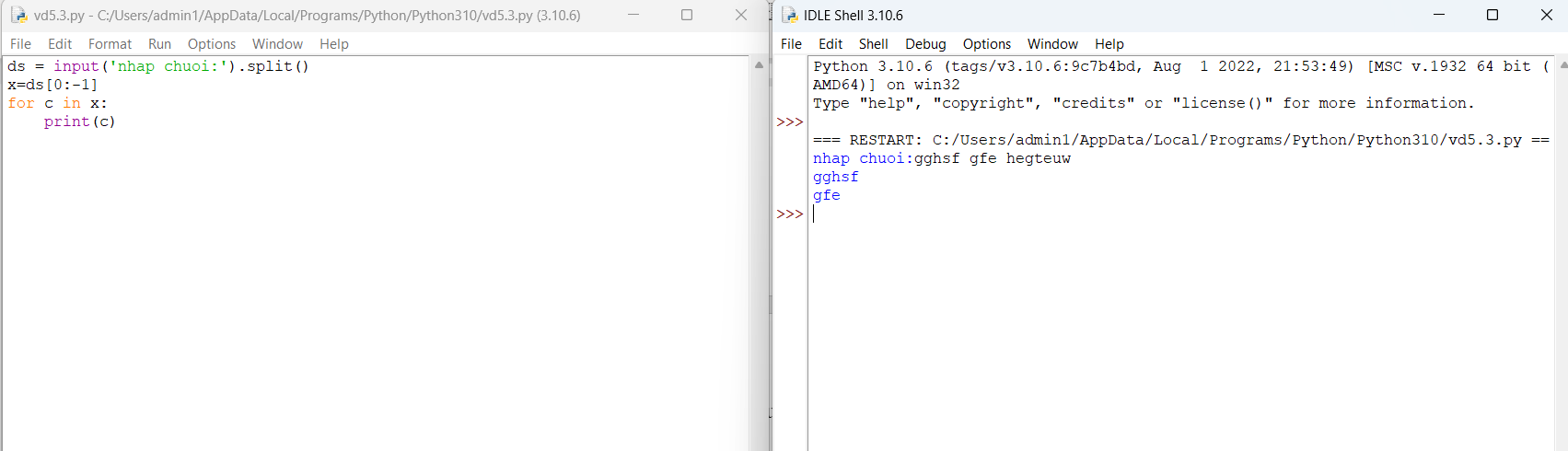
mọi từ có cùng độ dài nhất.



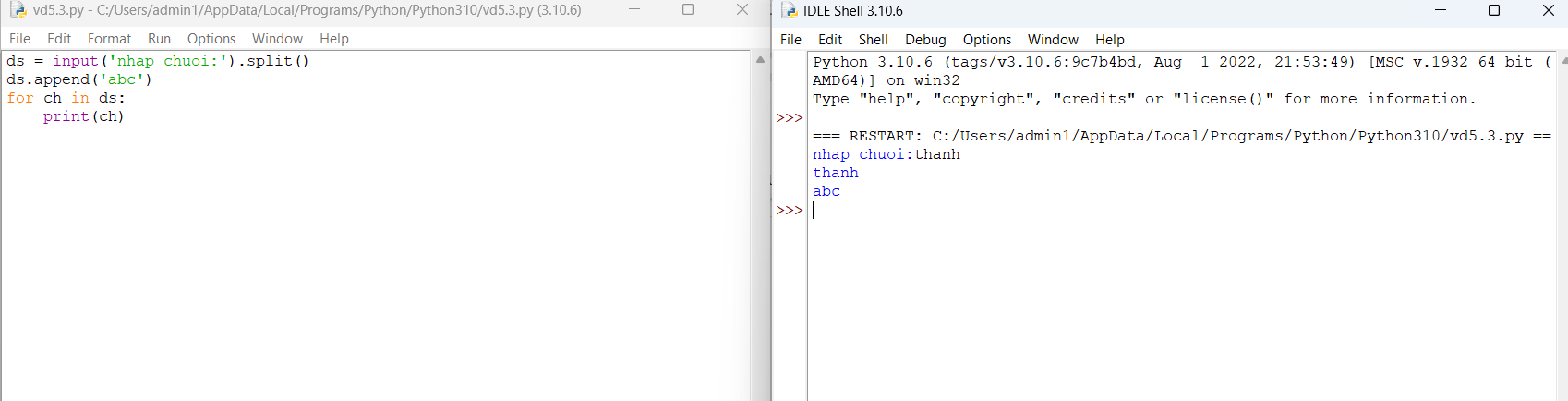
1. Nhập một list từ bàn phím



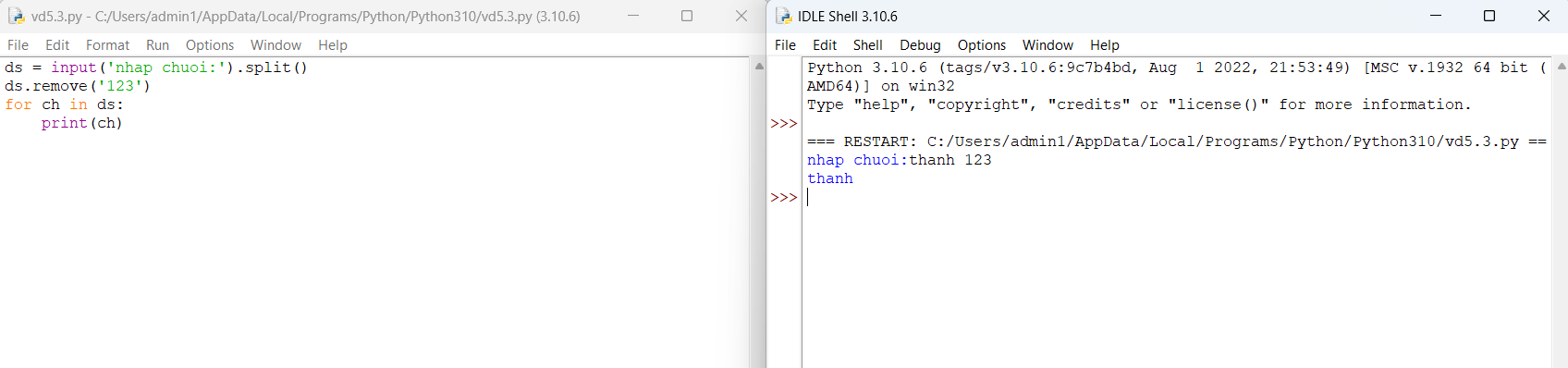
1. Cắt list: lấy list nhưng bỏ phần tử đầu và cuối



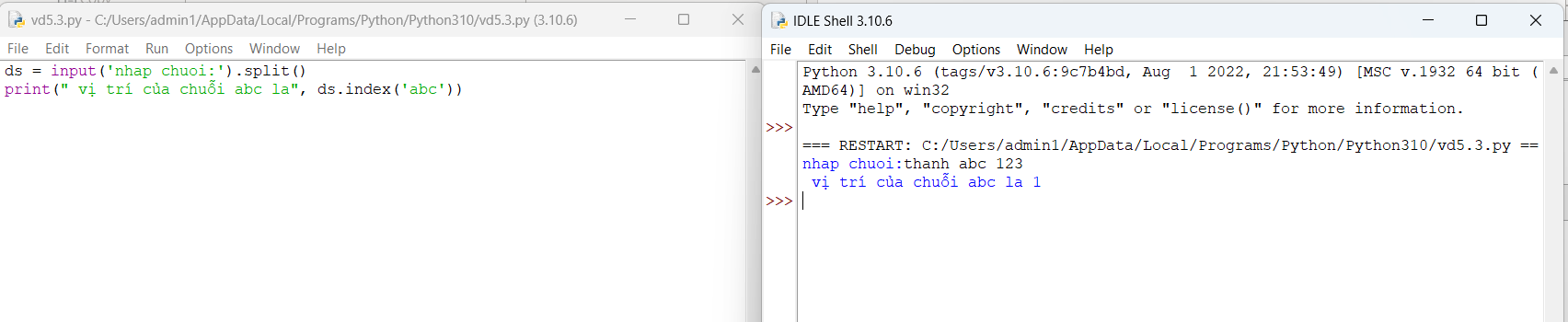
1. Thêm phần tử vào list



1. Bỏ phần tử khỏi list



1. Tìm kiếm phần tử trong list

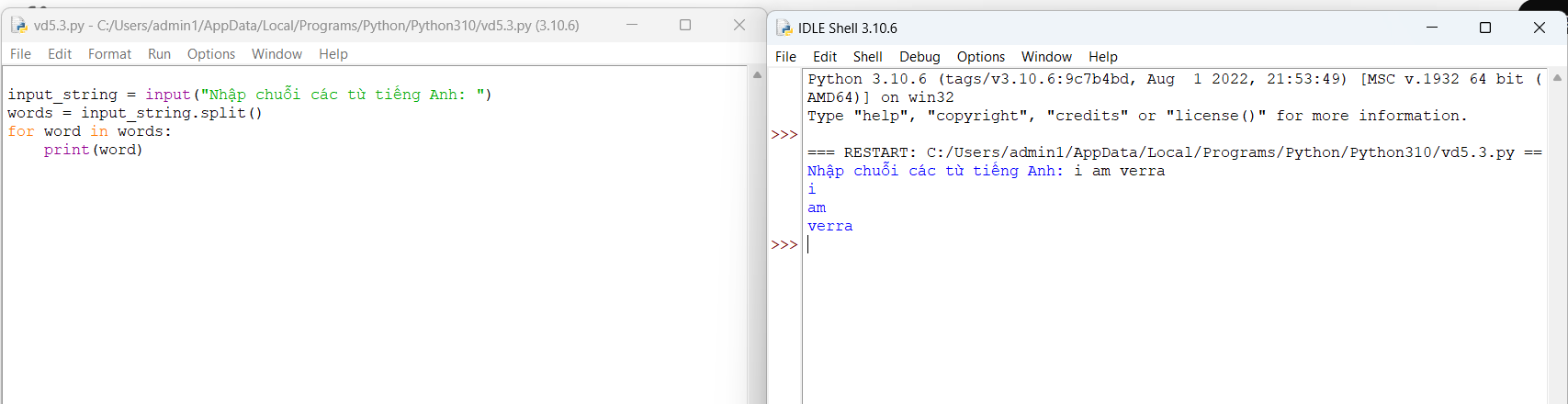


1. Sắp xếp các phần tử trong list



1. Người dùng nhập từ bàn phím liên tiếp các từ tiếng Anh viết tách nhau bởi dấu

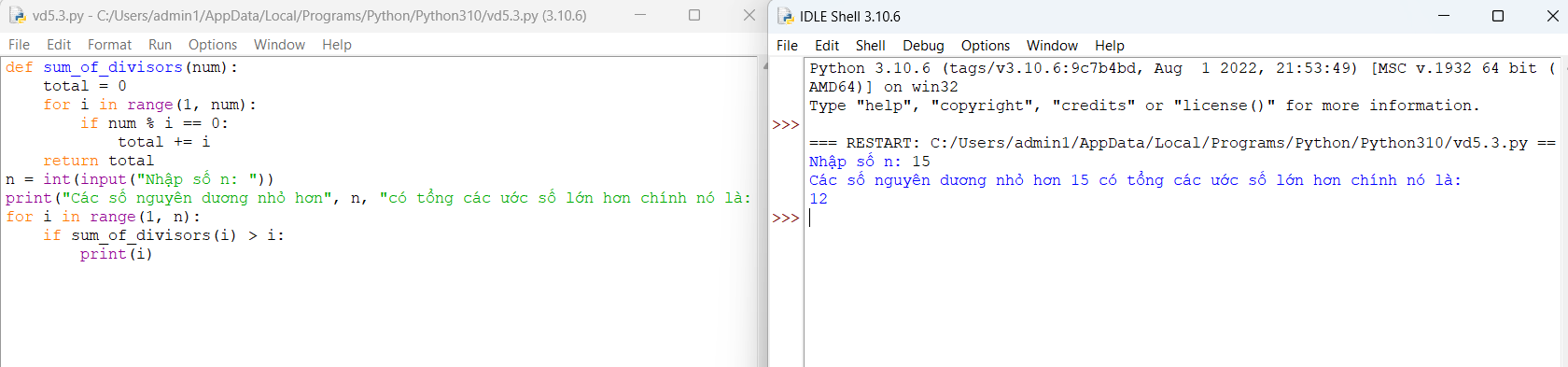
cách. Hãy nhập chuỗi đầu vào và tách thành các từ sau đó in ra màn hình các từ đó



1. 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

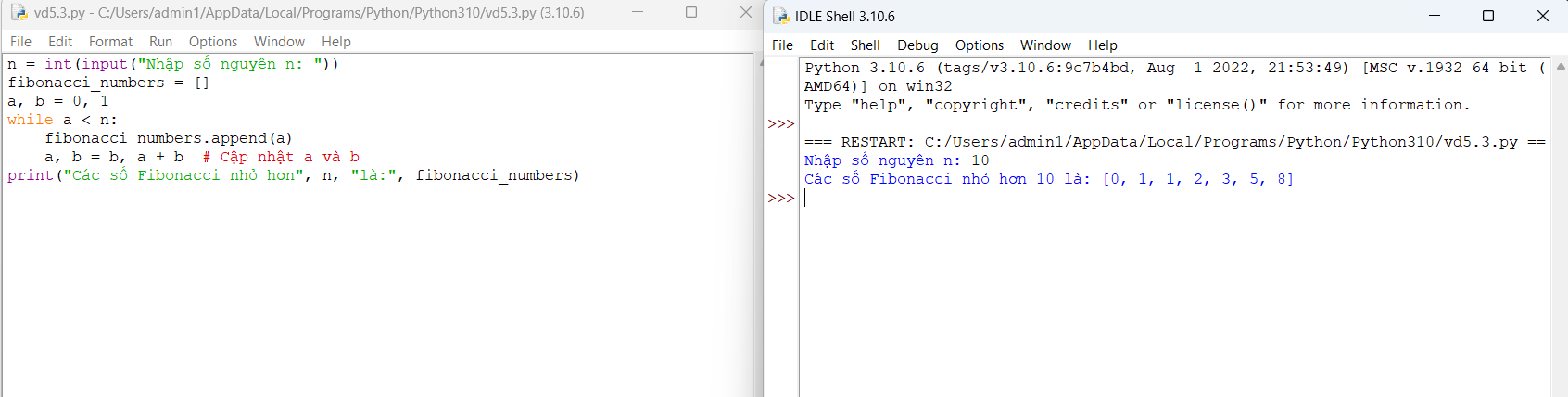


1. Nhập số n, in ra màn hình các số nguyên dương nhỏ hơn n có tổng các ước số lớn hơn chính nó.

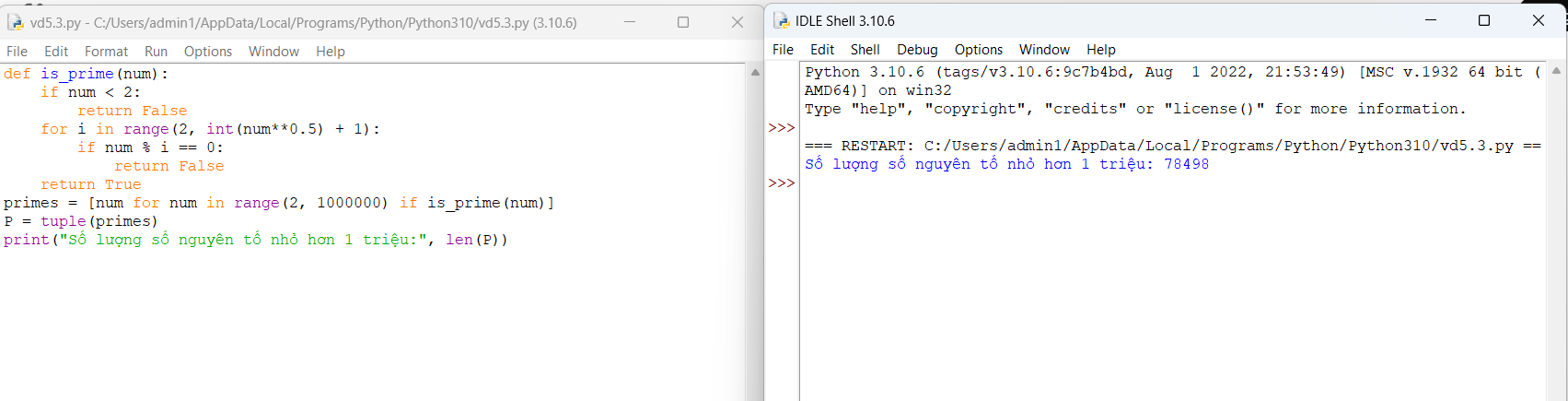


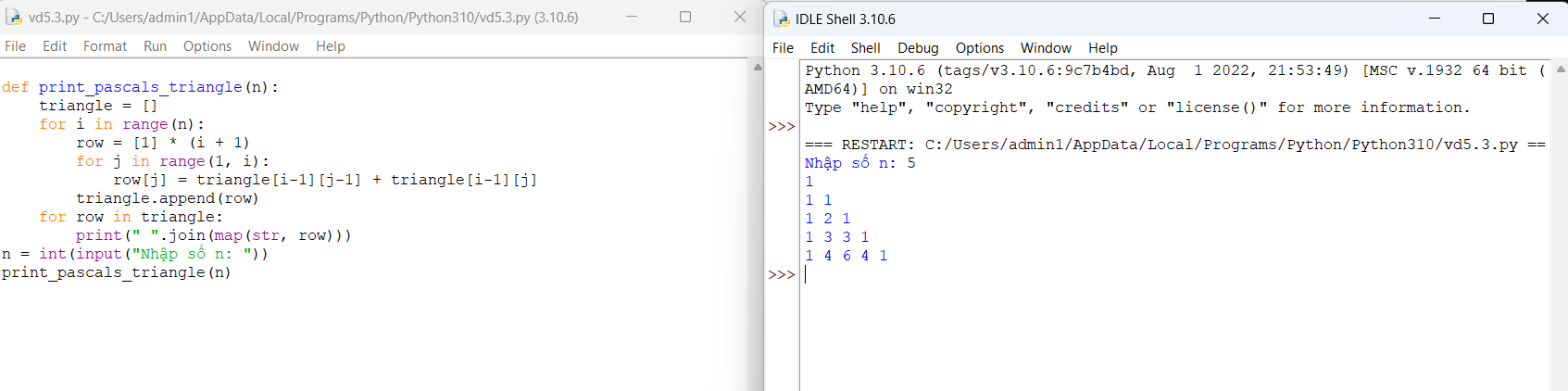
1. Hãy nhập số nguyên n, tạo một list gồm các số fibonacci nhỏ hơn n và in ra màn

hình



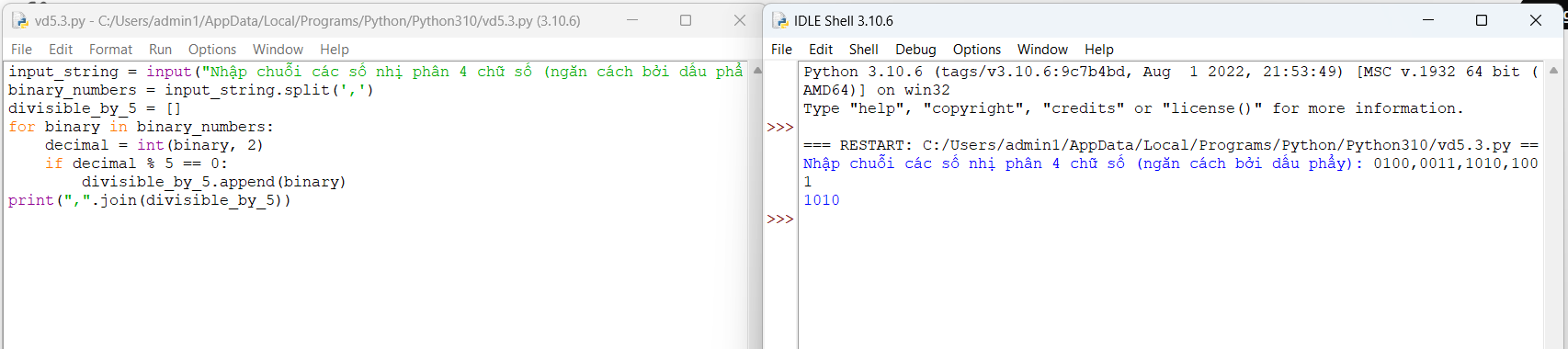
1. Hãy tạo ra tuple P gồm các số nguyên tố nhỏ hơn 1 triệu



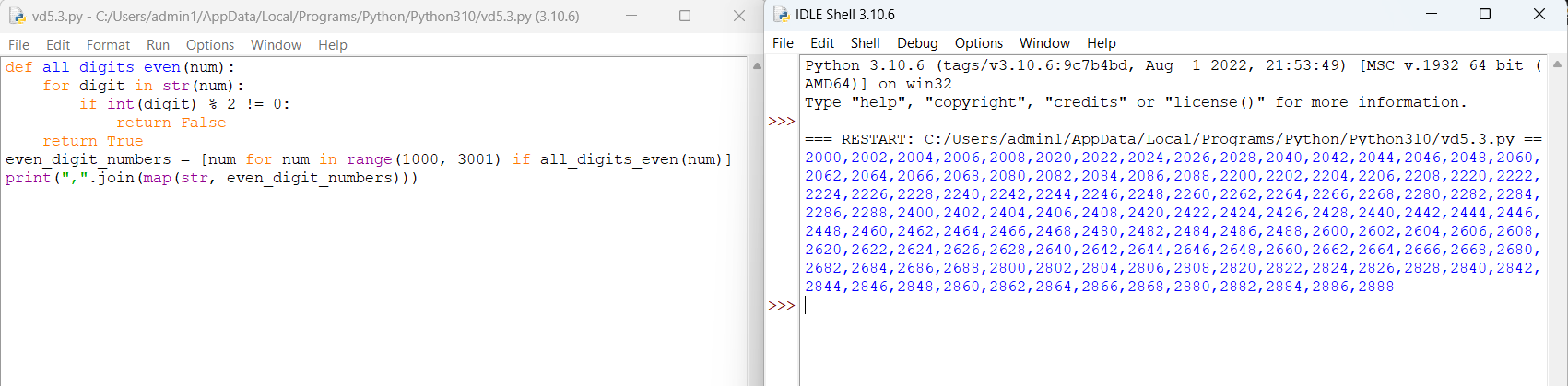
1. Nhập n, in n dòng đầu tiên của tam giác pascal
2. Viết một chương trình chấp nhận đầu vào là chuỗi các số nhị phân 4 chữ số, phân tách bởi dấu phẩy, kiểm tra xem chúng có chia hết cho 5 không. Sau đó in các số chia hết cho 5 thành dãy phân tách bởi dấu phẩy.

Ví dụ đầu vào là: 0100,0011,1010,1001

Đầu ra sẽ là: 1010



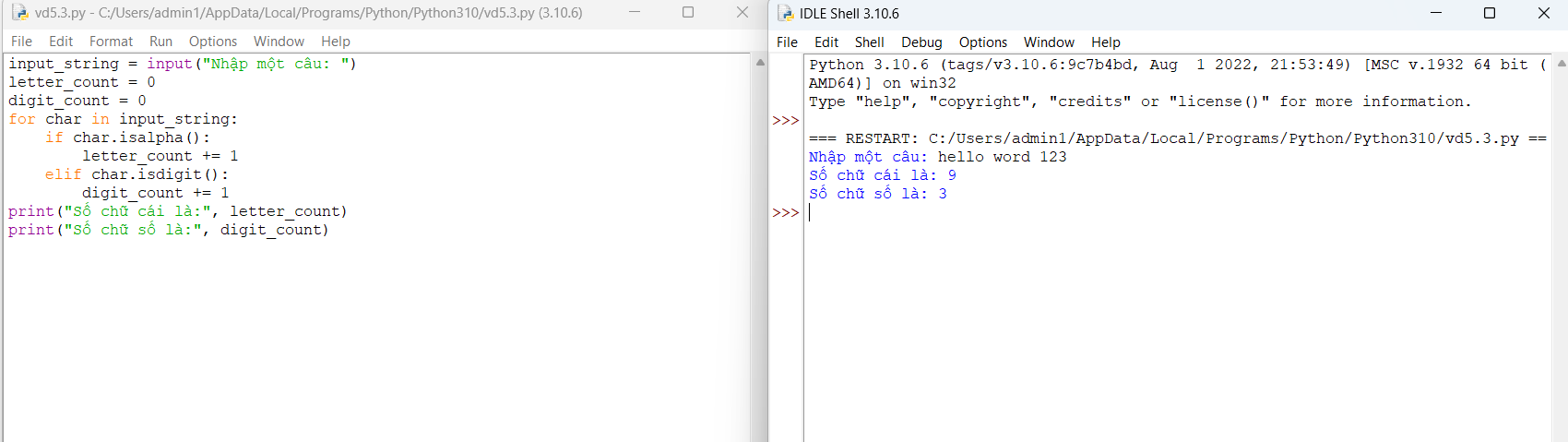
1. Viết một chương trình tìm tất cả các số trong đoạn 1000 và 3000 (tính cả 2 số này) sao cho tất cả các chữ số trong số đó là số chẵn. In các số tìm được thành chuỗi cách nhau bởi dấu phẩy, trên một dòng.



1. Viết một chương trình chấp nhận đầu vào là một câu, đếm số chữ cái và chữ số trong câu đó. Giả sử đầu vào sau được cấp cho chương trình: hello world! 123

Thì đầu ra sẽ là:Số chữ cái là: 10

Số chữ số là: 3



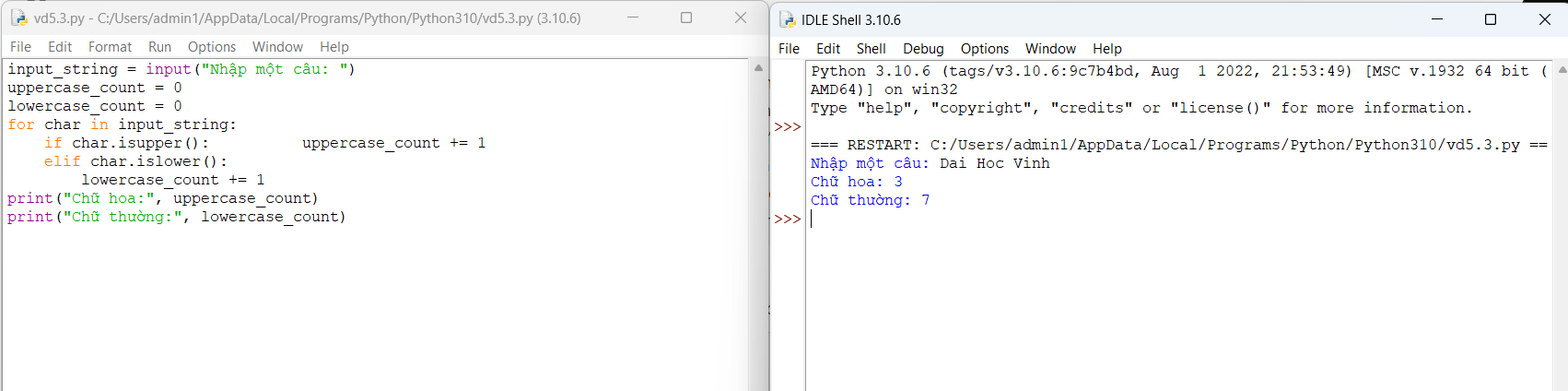
1. Viết một chương trình chấp nhận đầu vào là một câu, đếm chữ hoa, chữ thường.

Giả sử đầu vào là: Dai Hoc Vinh

Thì đầu ra là:

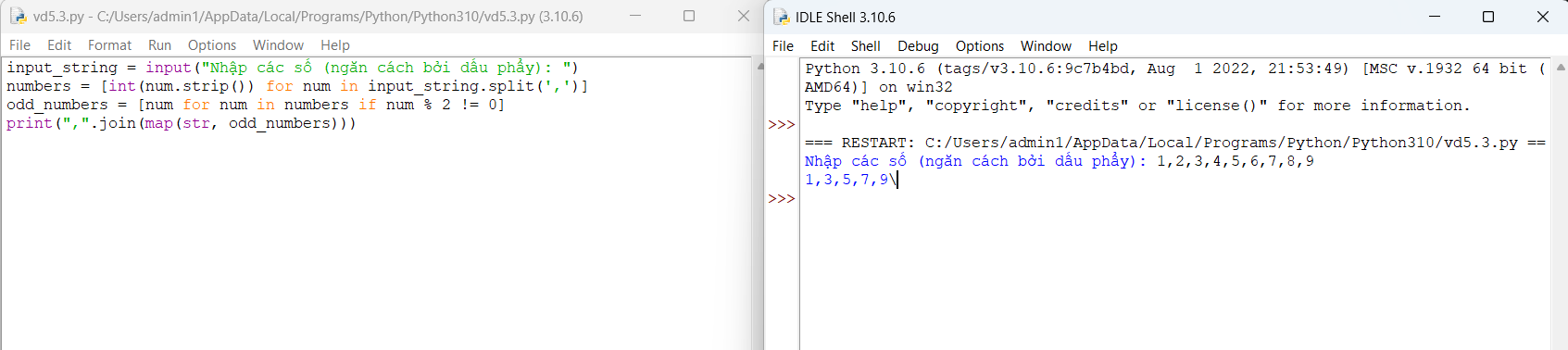
Chữ hoa: 3

Chữ thường: 7



1. Sử dụng một danh sách để lọc các số lẻ từ danh sách được người dùng nhập vào.

Giả sử đầu vào là: 1,2,3,4,5,6,7,8,9 thì đầu ra phải là: 1,3,5,7,9



1. Viết chương trình tính số tiền thực của một tài khoản ngân hàng dựa trên nhật ký

giao dịch được nhập vào từ giao diện điều khiển.

Định dạng nhật ký được hiển thị như sau:

D 100

W 200

(D là tiền gửi, W là tiền rút ra).

Giả sử đầu vào được cung cấp là:

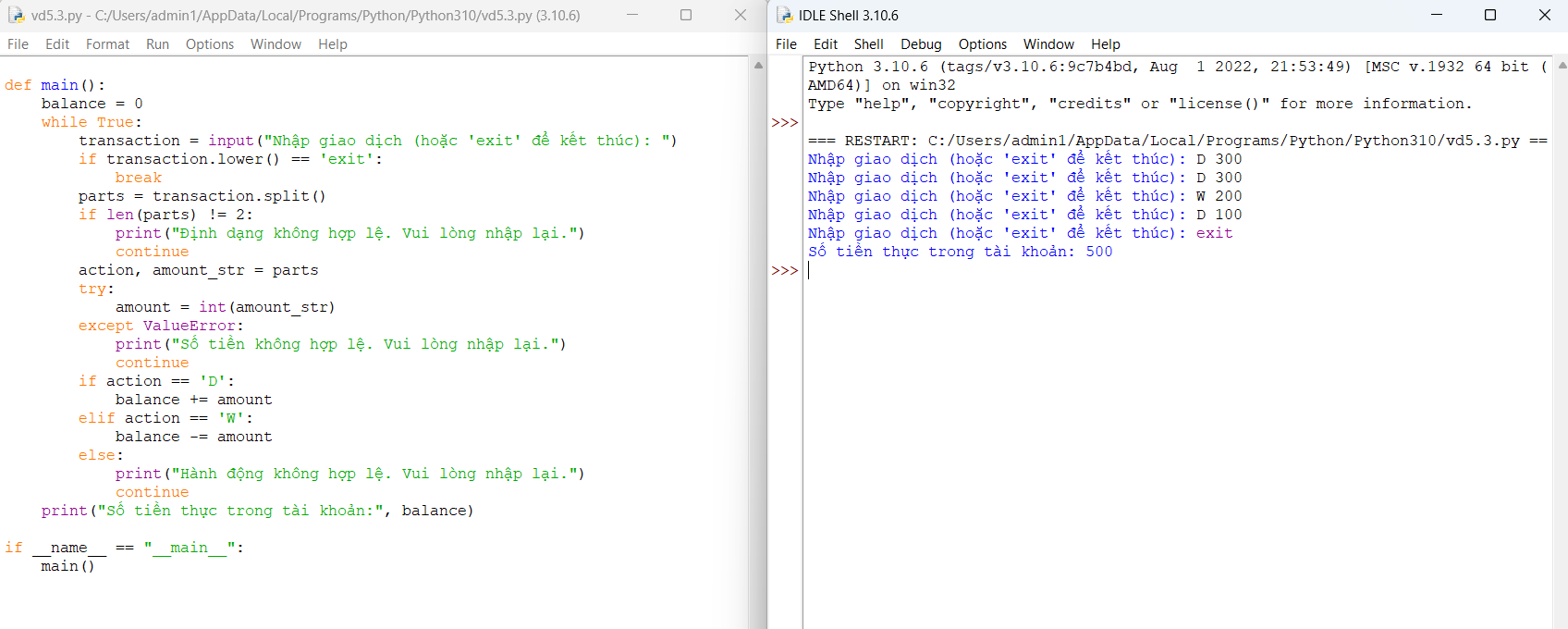
D 300

D 300

W 200

D 100

Thì đầu ra sẽ là:500



**1.5 . Câu hỏi kiểm tra**

- Giải thích chương trình đã viết

- Các hàm và các phương thức xây dựng sẵn cho các kiểu dữ liệu có cấu trúc

- Định nghĩa các kiểu dữ liệu có cấu trúc, phân biệt sự khác nhau, giải thích

**1.6. Tài liệu tham khảo**

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

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

* 1. **Mục đích**

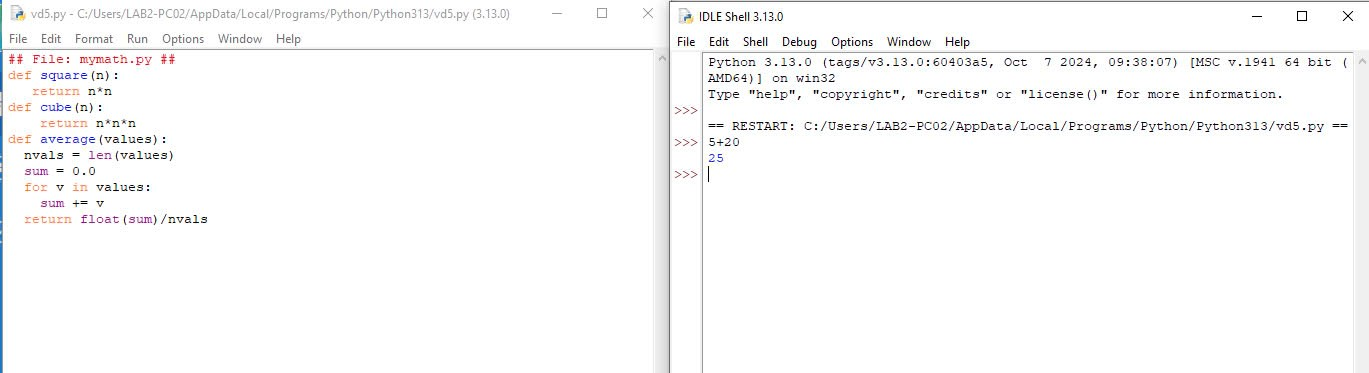
Giúp sinh viên nắm bắt việc thiết kế module trong lập trình python, sử dụng

module thue viện numpy trong các ứng dụng.

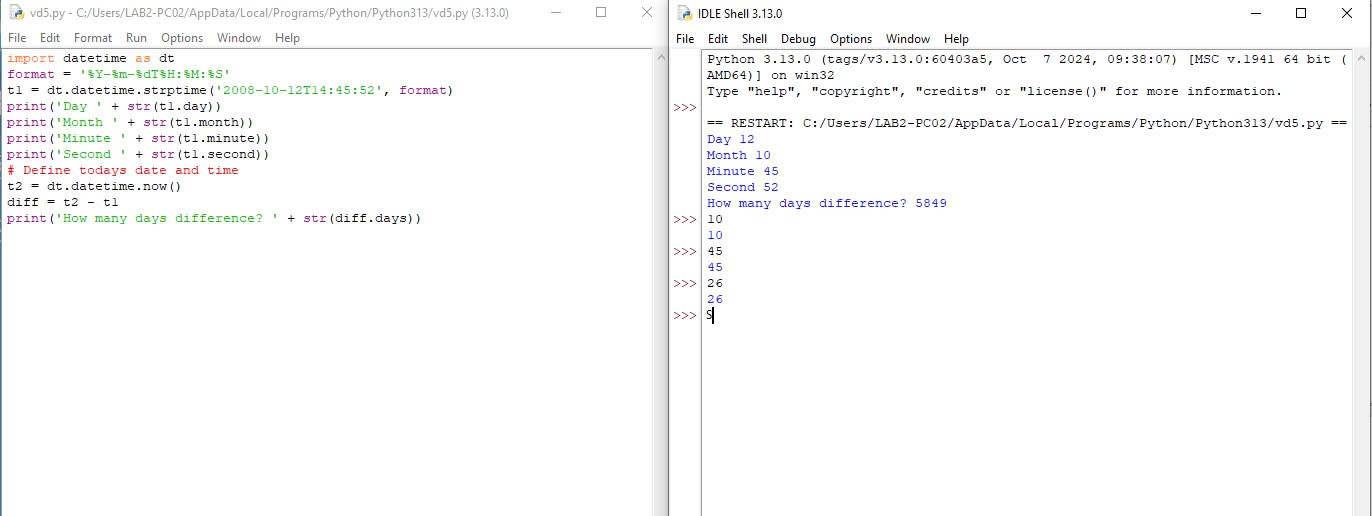
* 1. **Các bước tiến hành và kết quả**

1. Sử dụng module. Định nghĩa một module toán học gọi là mymath và sử dụng

module này từ một tập lệnh riêng biệt.



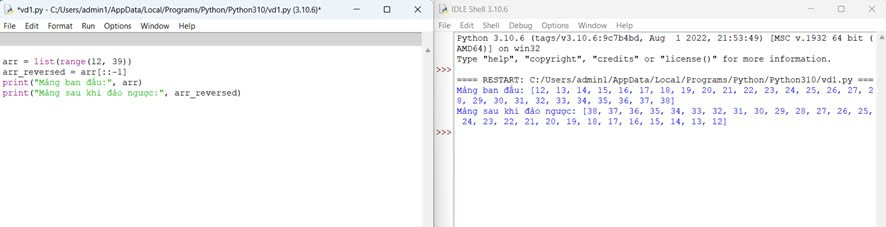
1. Sử dụng thư viện tiêu chuẩn của python (datetime)



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



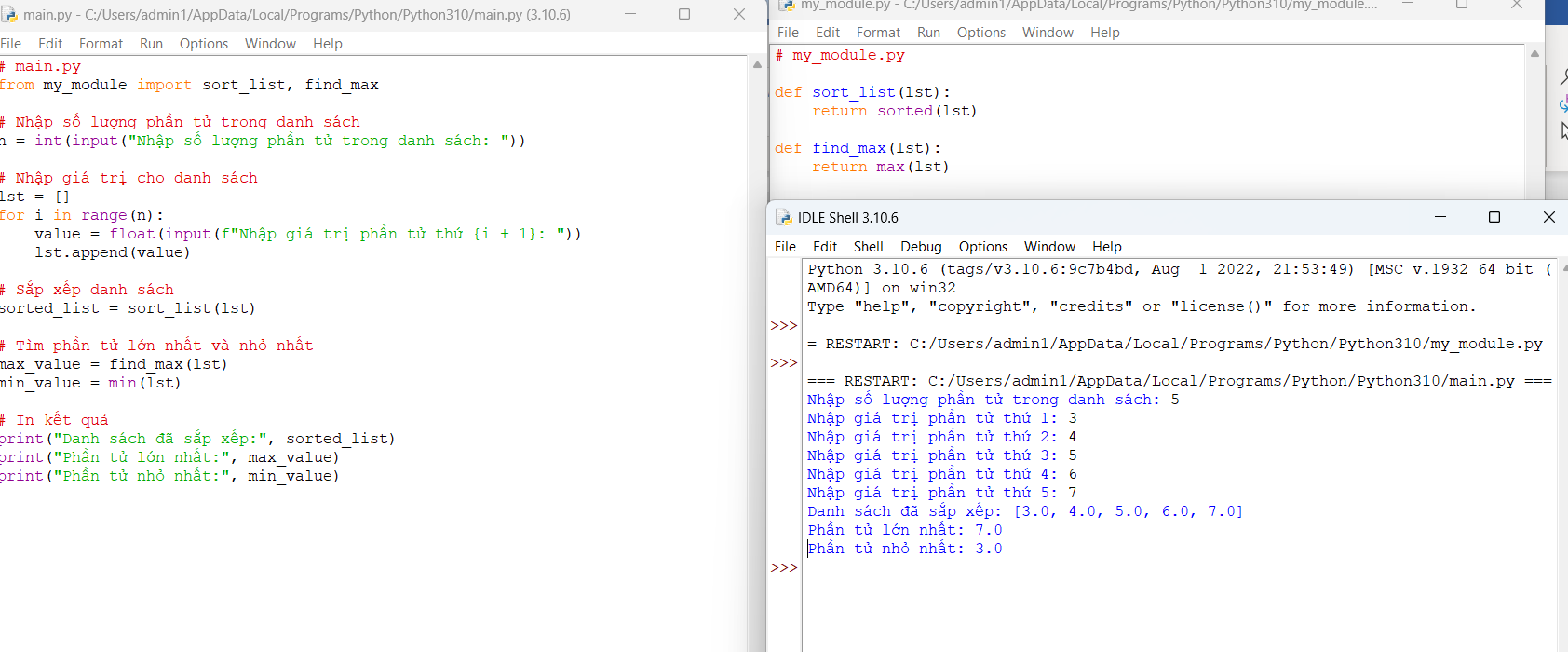
1. Viết chương trình để tạo một mảng với các giá trị nằm trong khoảng từ 12 đến 38 và đảo ngược mảng đã tạo (phần tử đầu tiên trở thành cuối cùng)



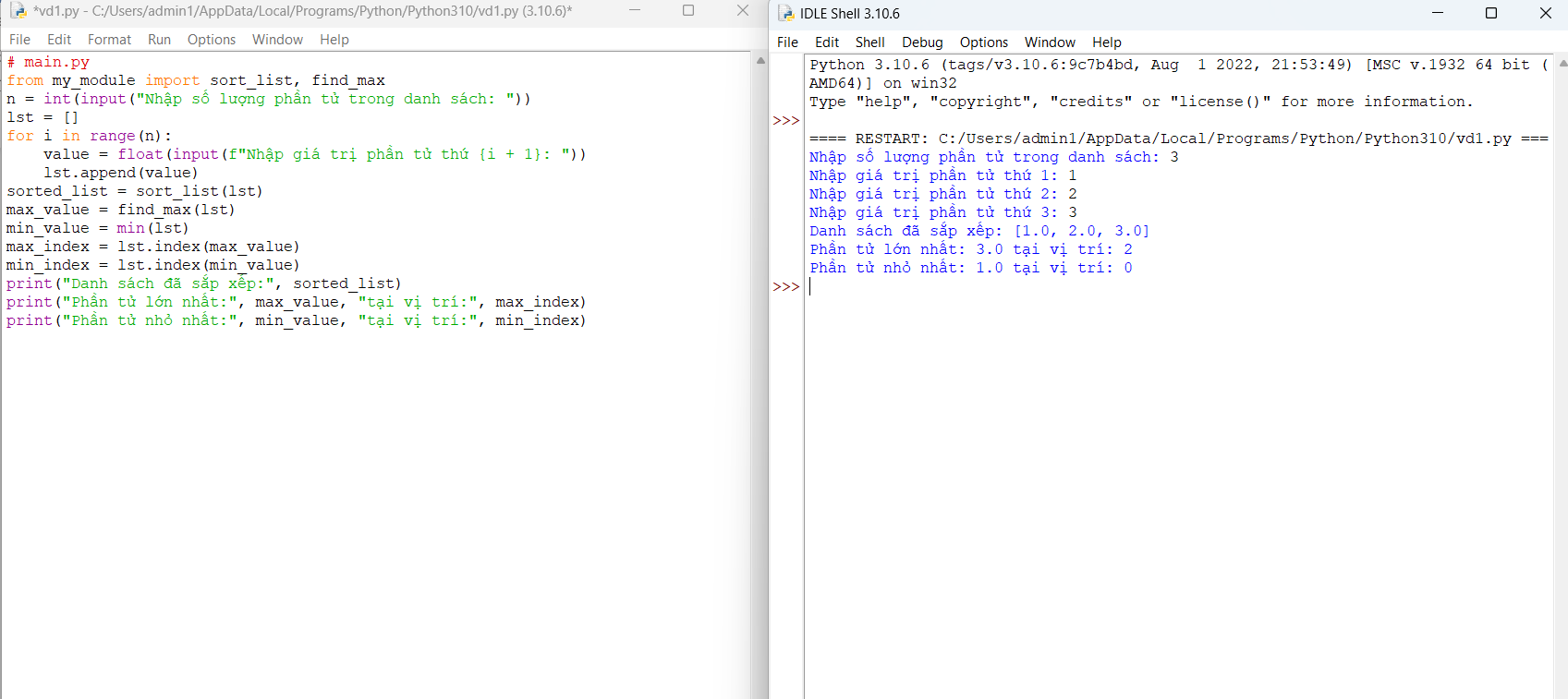
1. Viết chương trình tìm phần tử lớn nhất và nhỏ nhất của một danh sách

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

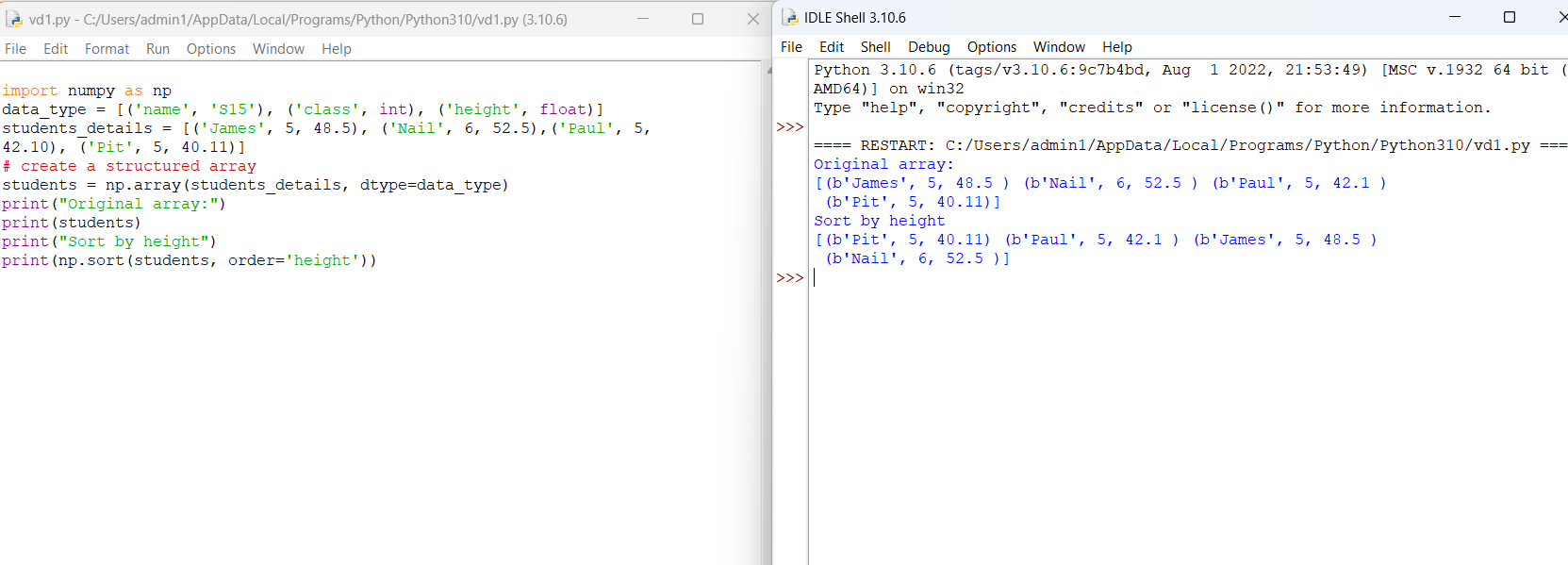
- Phương thức sắp xếp và tìm phần tử lớn nhất được viết thành module



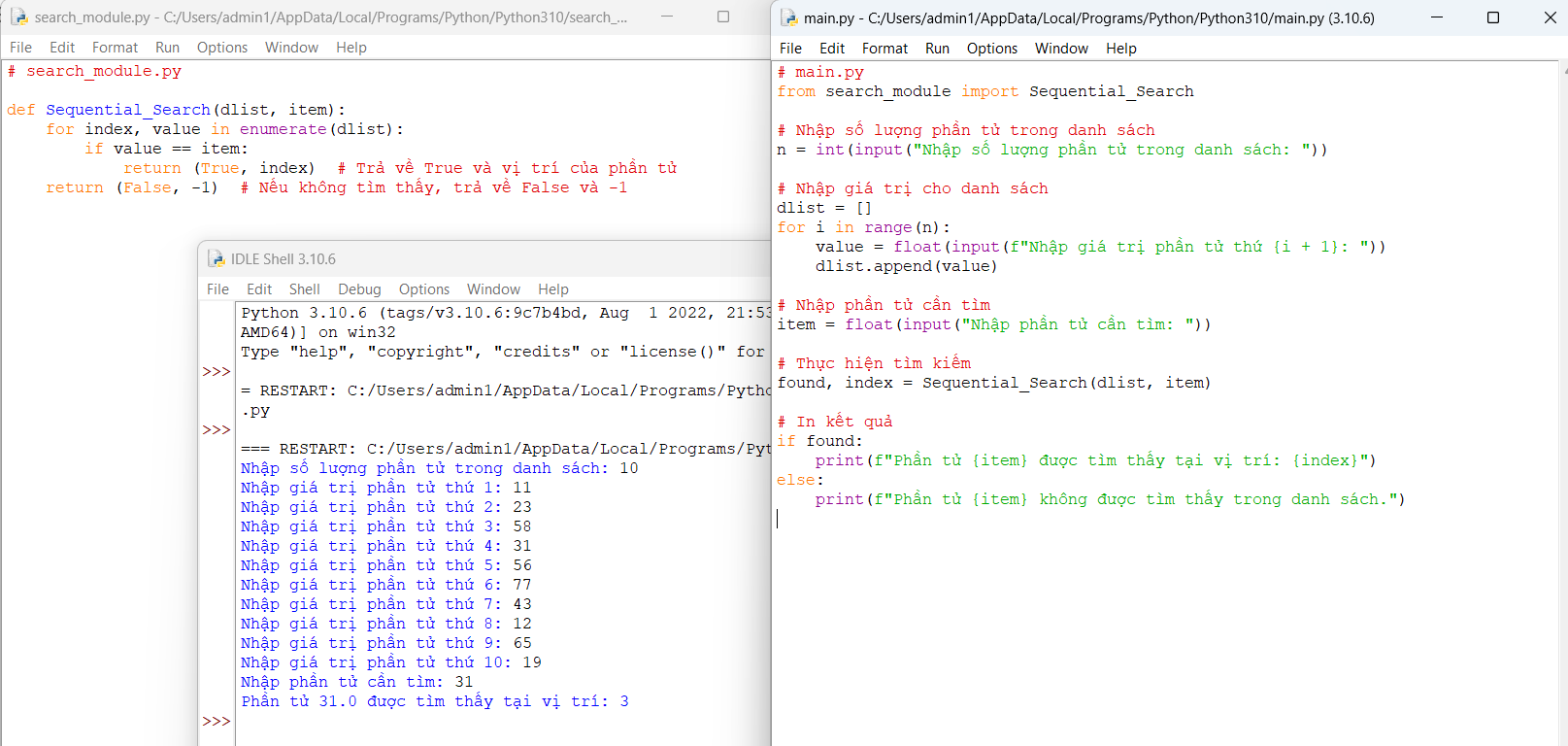
1. In ra vị trí phần tử lớn nhất và nhỏ nhất tìm được ở bài tập trên.



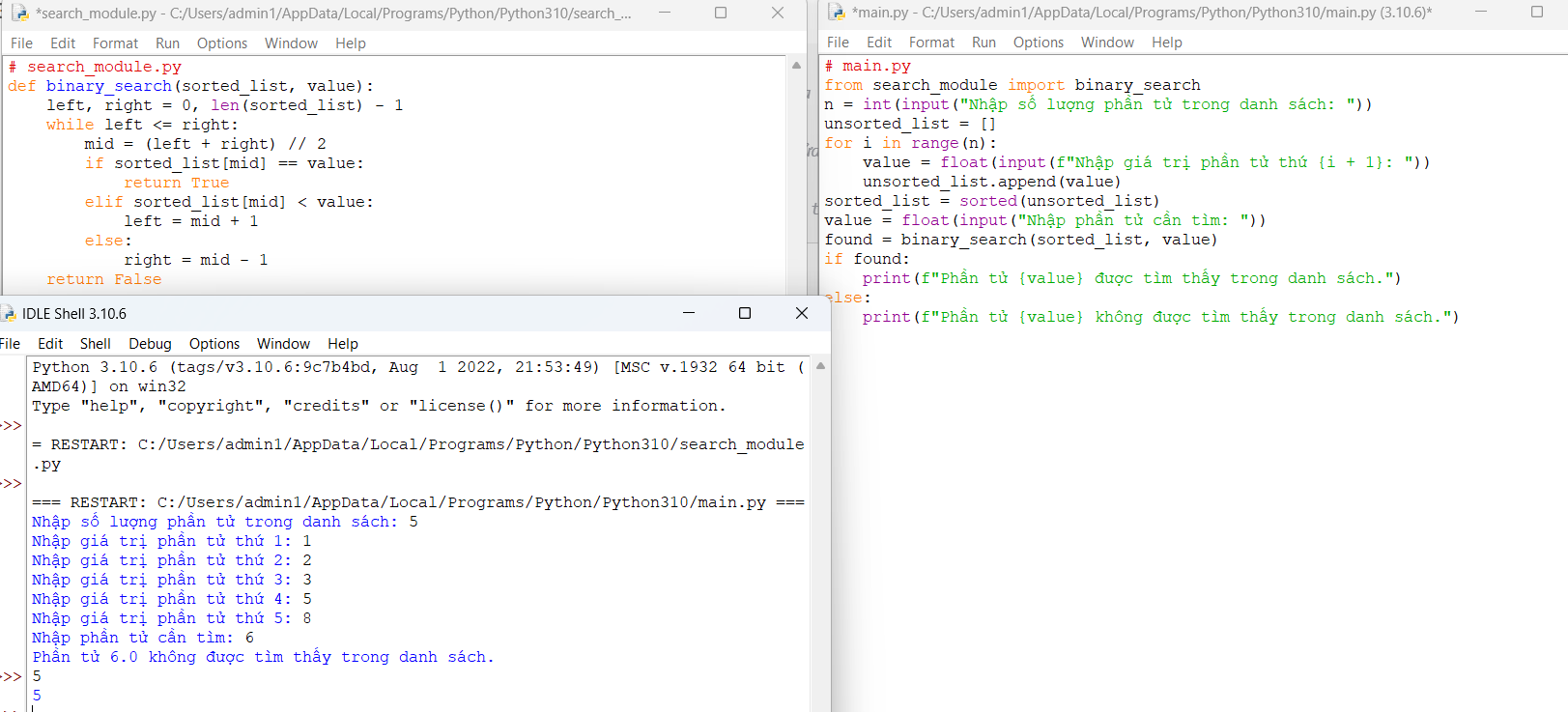
1. Viết chương trình sử dụng thư viện NumPy để tạo một mảng có cấu trúc từ tên sinh viên, chiều cao, lớp và các kiểu dữ liệu của họ. Bây giờ sắp xếp các mảng theo chiều cao.



1. 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. Sequential\_Search([11,23,58,31,56,77,43,12,65,19],31) -> (True, 3)



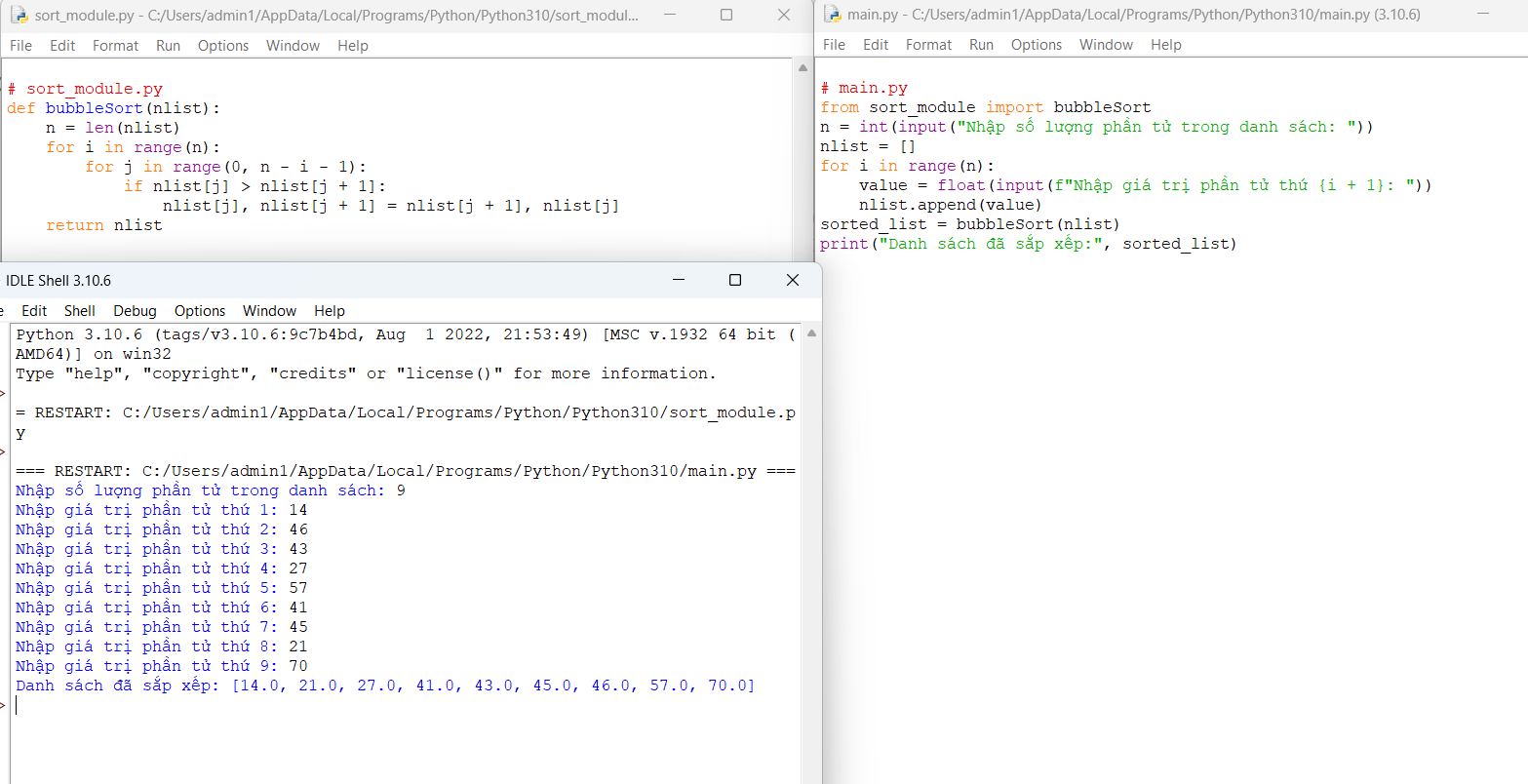
1. Xây dựng hàm *“binary\_search(list, value)”* (giải thuật tìm kiếm nhị phân) dưới dạng module. Viết chương trình nhập một list n phần tử từ bàn phím và tìm kiếm phần tử *value* bất kỳ.  
   binary\_search([1,2,3,5,8], 6) -> False  
   binary\_search([1,2,3,5,8], 5) -> True



1. Xây dựng hàm “bubbleSort (nlist)” (giải thuật sắp xếp nổi bọt) dưới dạng module.Viết chương trình nhập một nlist n phần tử từ bàn phím và sắp xếp.

Sample Data: [14,46,43,27,57,41,45,21,70]

Expected Result: [14, 21, 27, 41, 43, 45, 46, 57, 70]



1. . 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 theo lớp, sau đó chiều cao nếu lớp bằng nhau.

Dữ liệu đầu vào:

[('James', 5, 48.5 ) ('Nail', 6, 52.5 ) ('Paul', 5, 42.1 ) ('Pit', 5, 40.11)]

Kết quả sắp xếp:

[('Pit', 5, 40.11) ('Paul', 5, 42.1 ) ('James', 5, 48.5 ) ('Nail', 6, 52.5 )]



1. Viết chương trình sử dụng thư viện NumPy để sắp xếp id sinh viên với chiều cao tăng dần của sinh viên từ id sinh viên và chiều cao đã cho. In các chỉ số nguyên mô tả thứ tự sắp xếp theo nhiều cột và dữ liệu được sắp xếp (sử dụng hàm lexsort())

Dữ liệu đầu vào:

student\_id = [1023, 5202, 6230, 1671, 1682, 5241, 4532]

student\_height = [40., 42., 45., 41., 38., 40., 42.0]

Kết quả:

Chỉ số:

[4 0 5 3 6 1 2]

Dữ liệu sắp xếp:

1682 38.0

1023 40.0

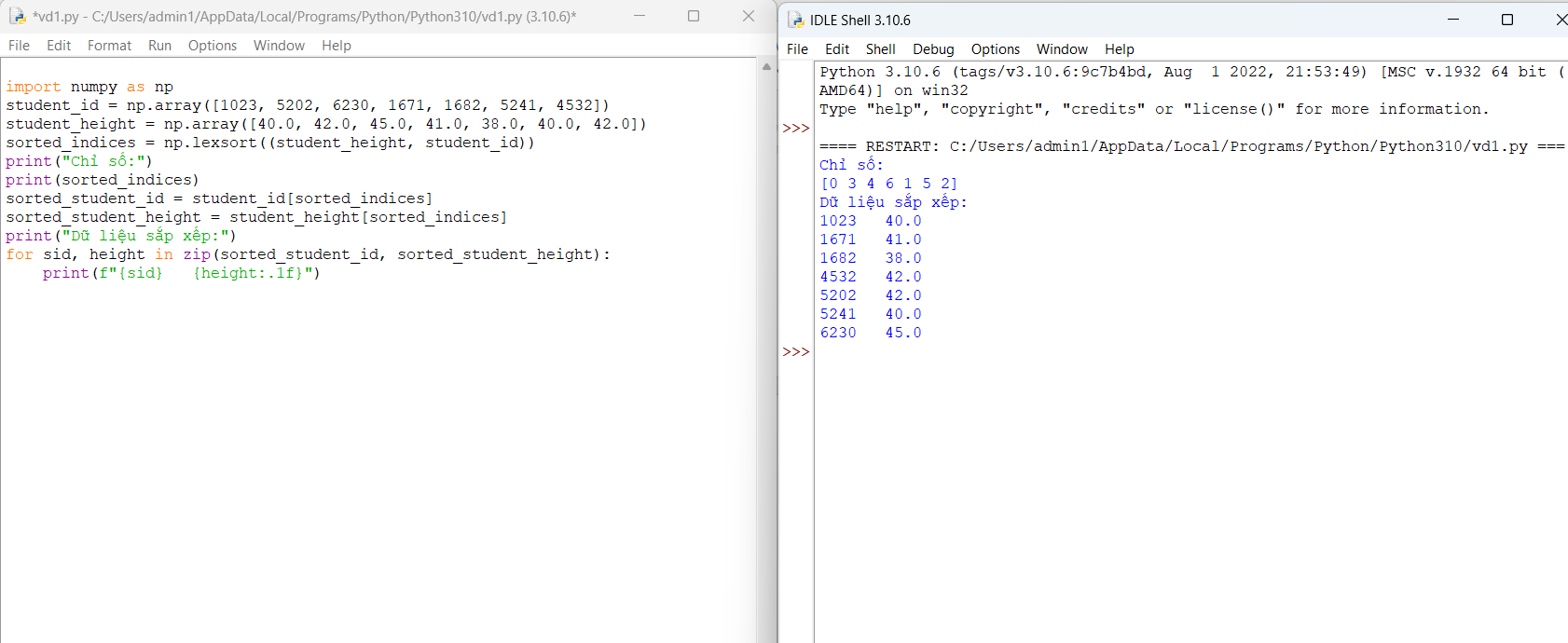
5241 40.0

1671 41.0

4532 42.0

5202 42.0

6230 45.0



**1.3. Câu hỏi kiểm tra**

- Giải thích chương trình đã viết

- Tạo và sử dụng module trong python

- Các thuật toán sắp xếp, tìm kiếm

- Thư viện numpy

**1.4. Tài liệu tham khảo**

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

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

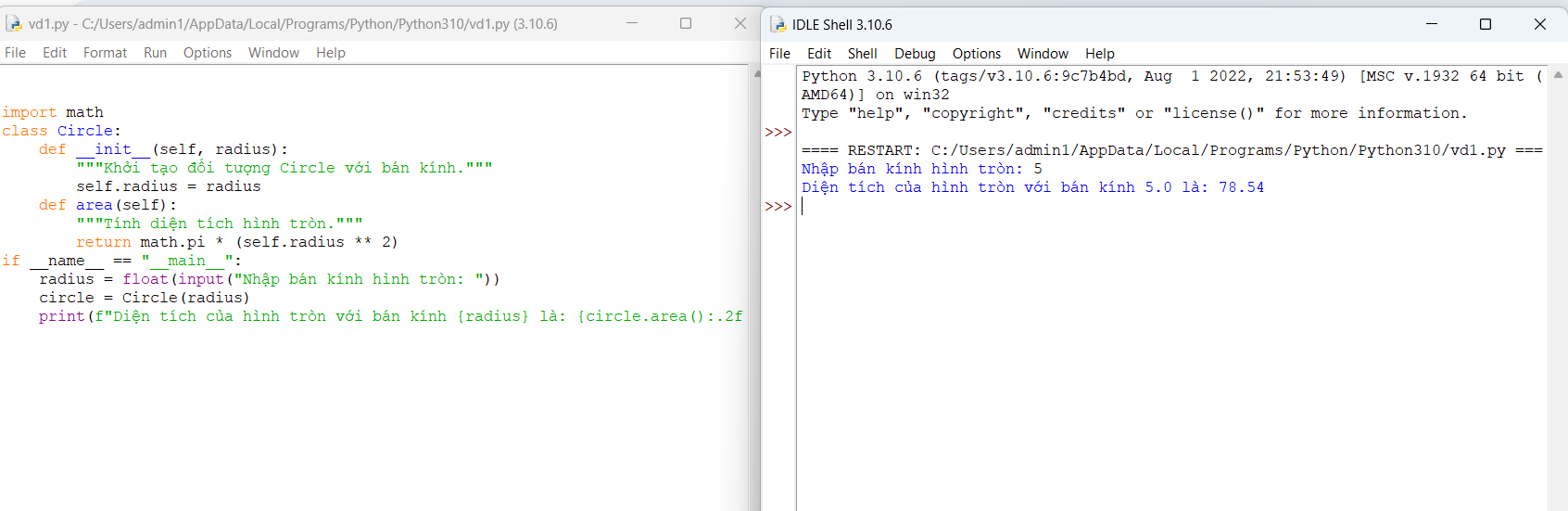
**1.1. Mục đích**

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

**1.2.Các bước tiến hành và kết quả**

1. Định nghĩa một class có tên là Circle có thể được xây dựng từ bán kính. Circl

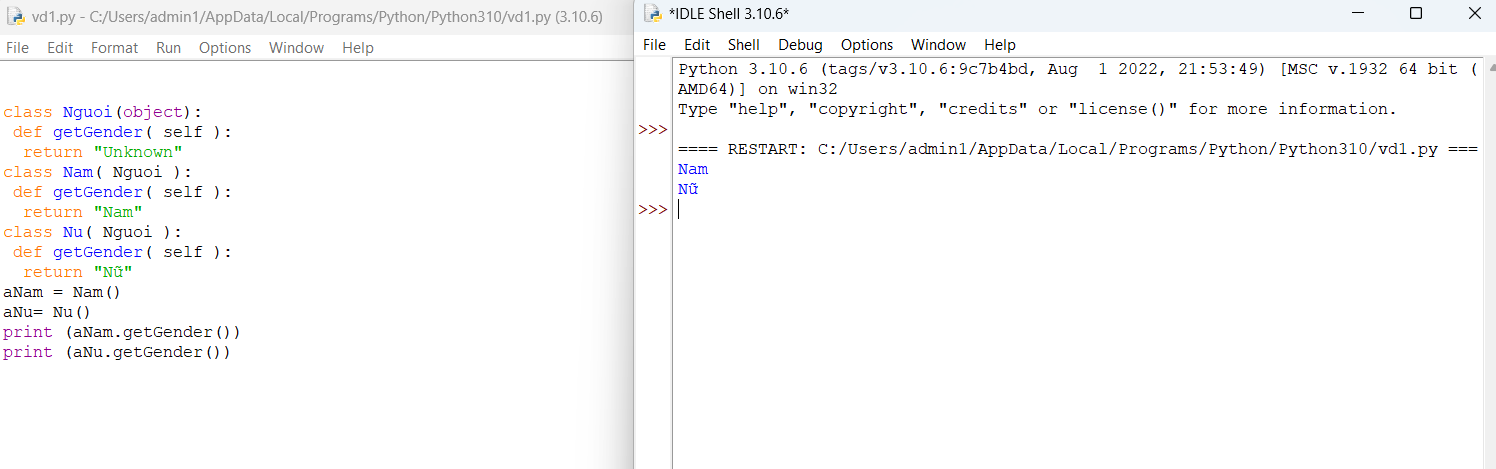
có một method có thể tính diện tích



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

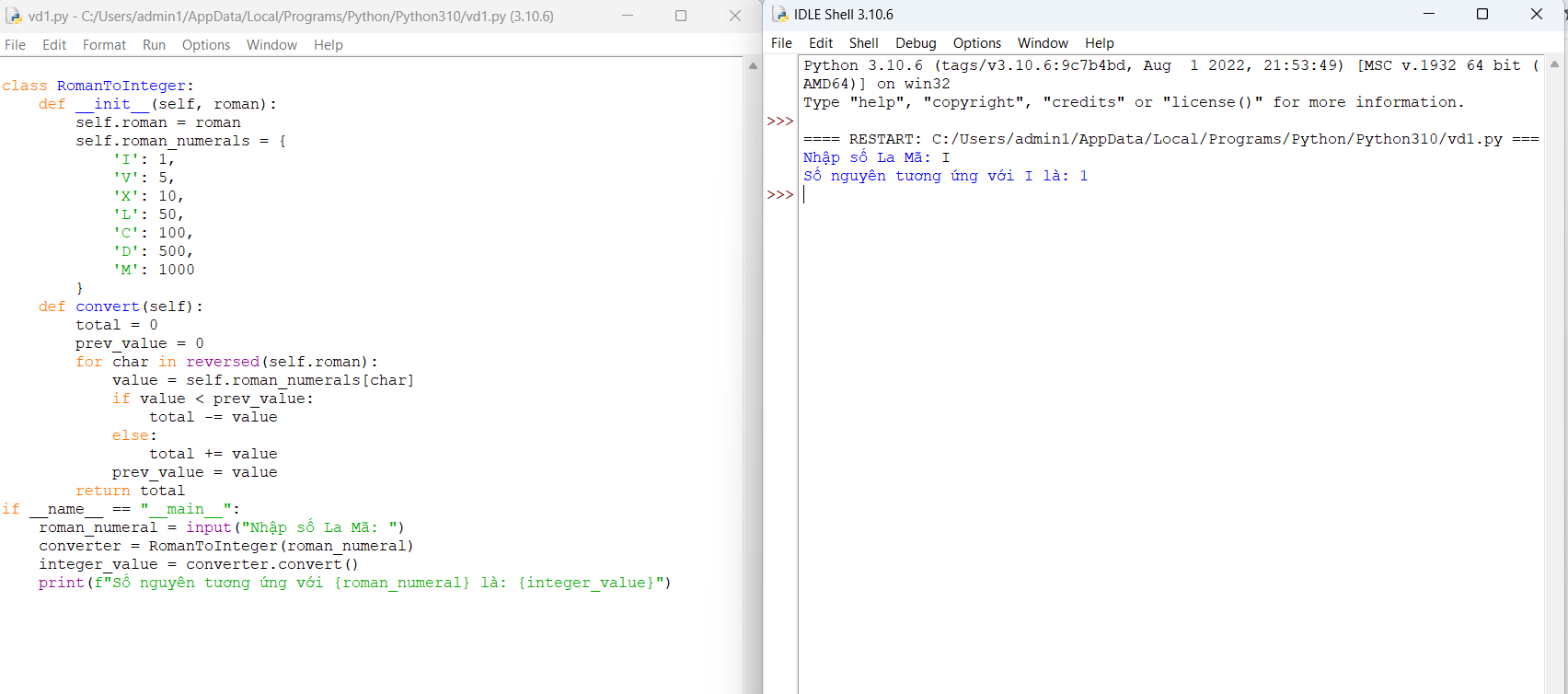
1. Định nghĩa class Nguoi và 2 class con của nó: Nam, Nu. Tất cả các class có

method "getGender" có thể in "Nam" cho class Nam và "Nữ" cho class Nu.



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

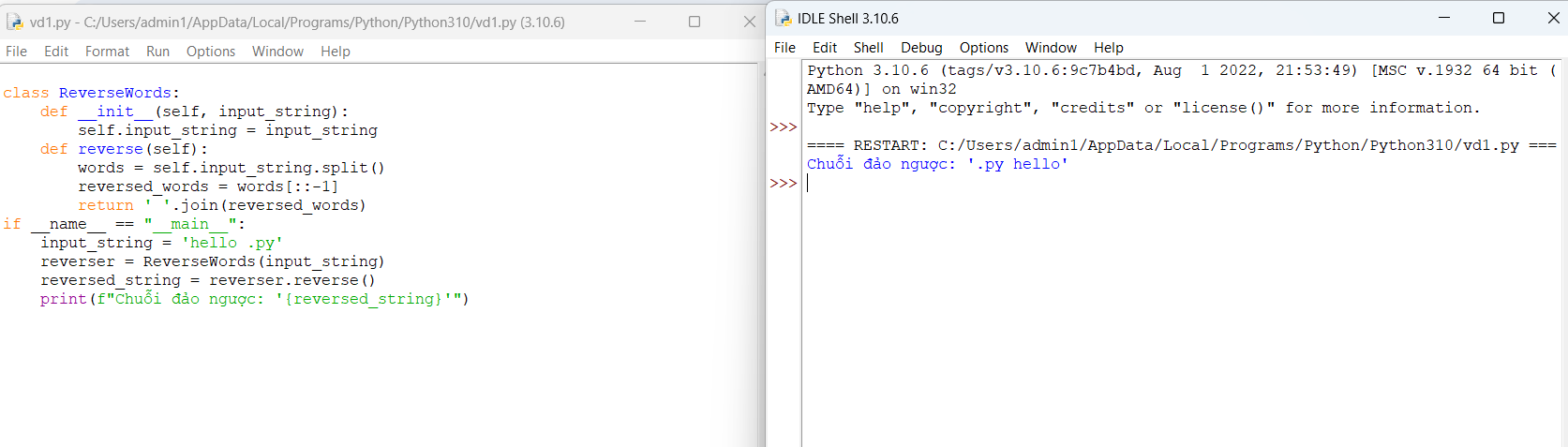
số nguyên



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

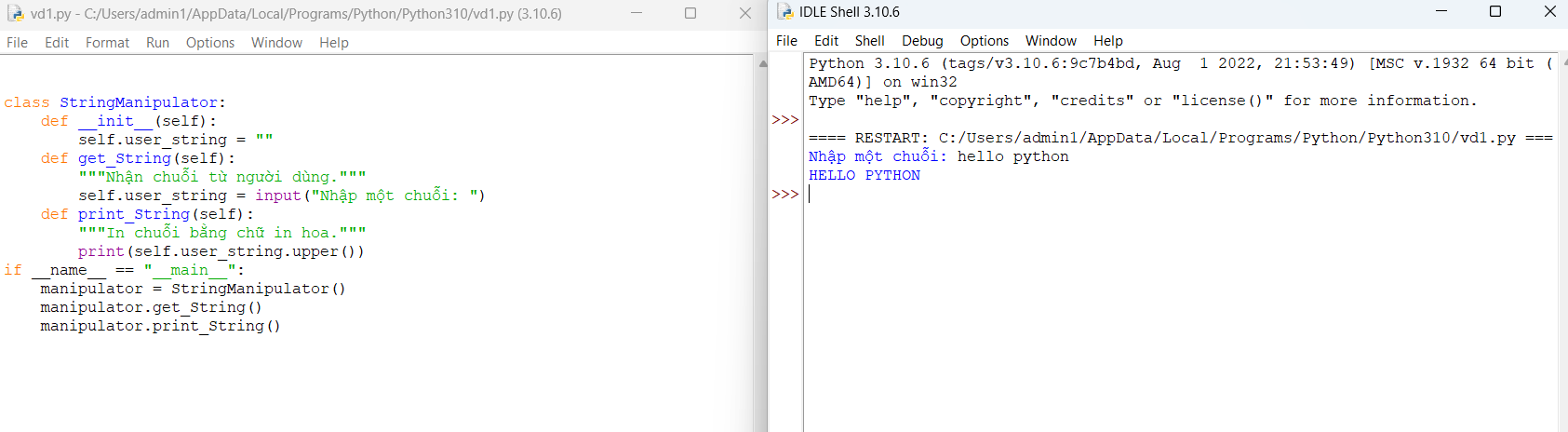
Dữ liệu vào : 'hello .py'

Đầu ra : '.py hello'



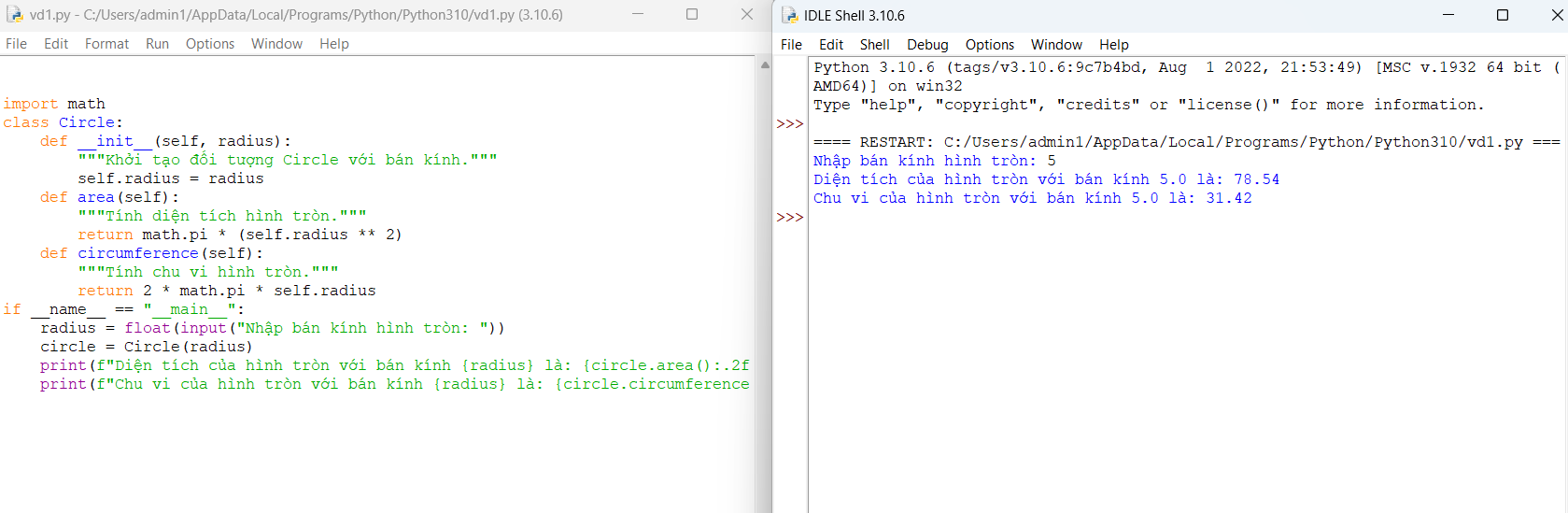
1. Viết một class Python có hai phương thức get\_String và print\_String. get\_String

chấp nhận một chuỗi từ người dùng và print\_String in chuỗi đó bằng chữ in hoa.

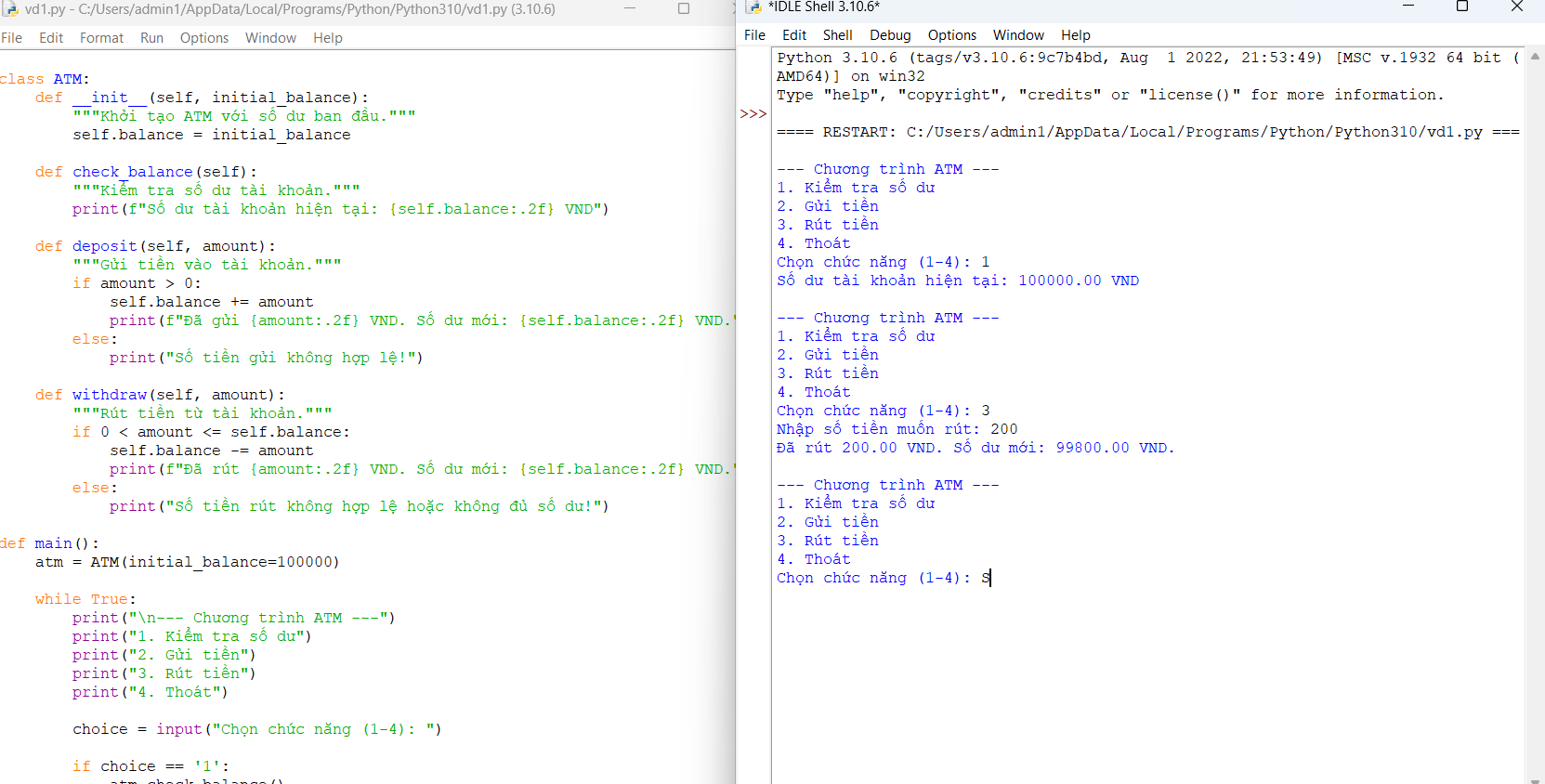


1. Viết một class Python có tên Circle được xây dựng theo bán kính và hai phương

thức sẽ tính diện tích và chu vi của hình tròn.



1. Chương trình ATM đơn giản



**1.3. Câu hỏi kiểm tra**

- Giải thích chương trình đã viết

- Các khái niệm về lớp , đối tượng, thuộc tính, phương thức

**1.4. Tài liệu tham khảo**

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

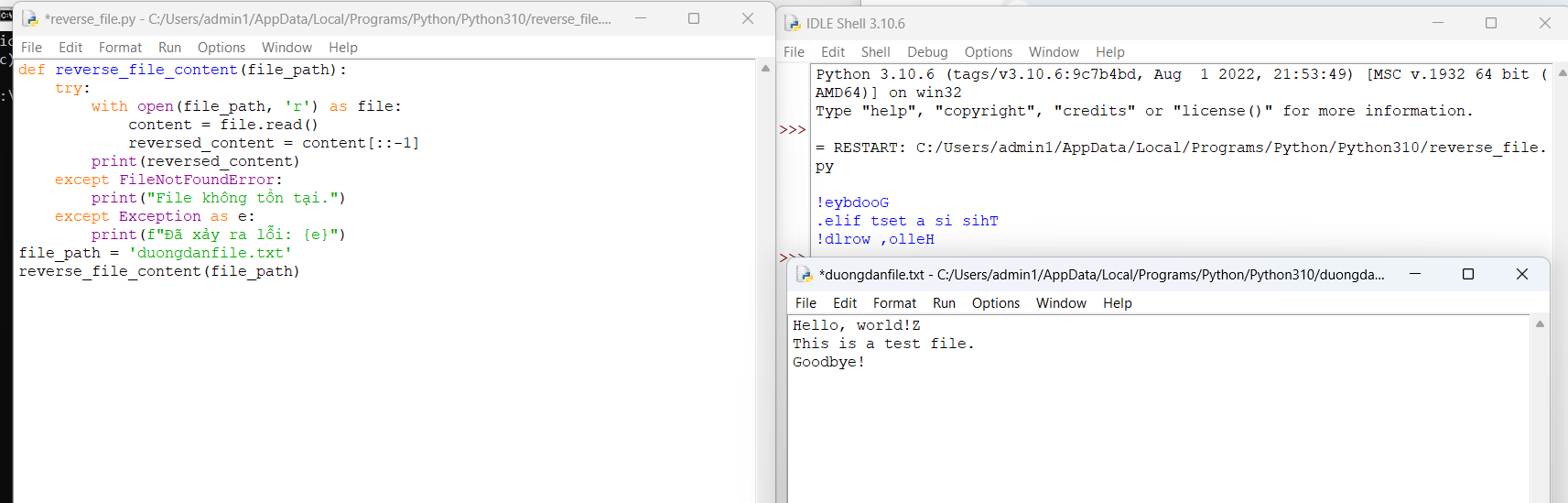
**Bài 7: Thao tác trên tập tin và thư mục trong Python**

* 1. **Mục đích**

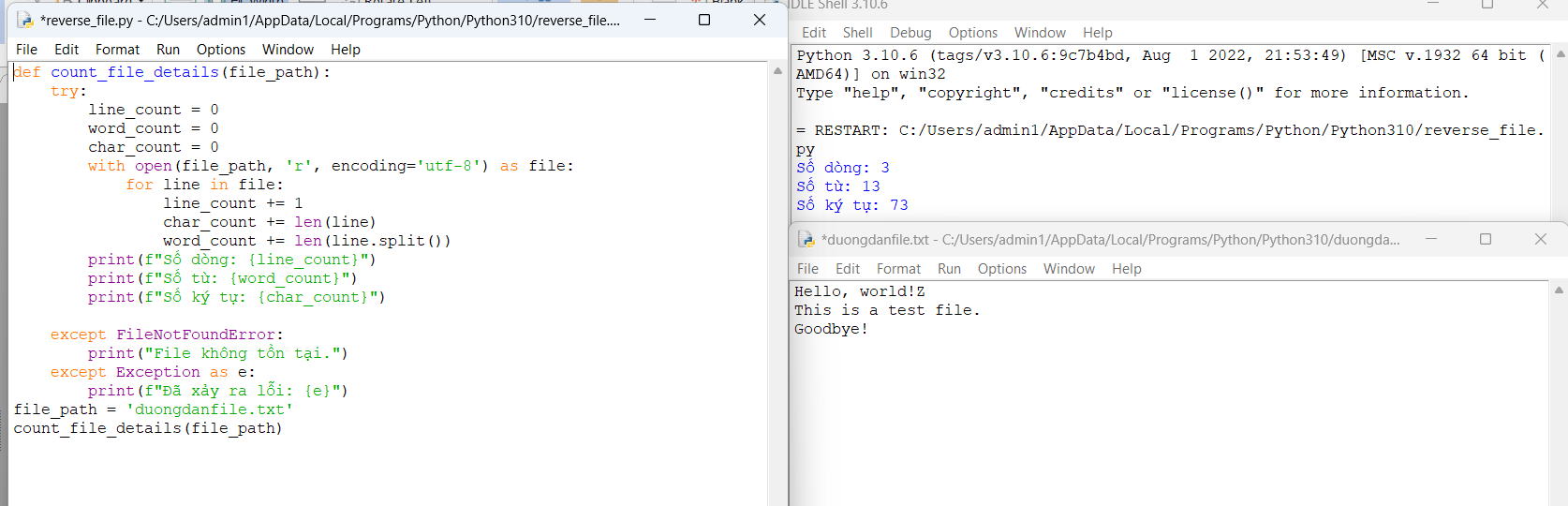
Giúp sinh viên có thể thao tác với các file văn bản sử dụng python.

**1.2. Các bước tiến hành và kết quả**

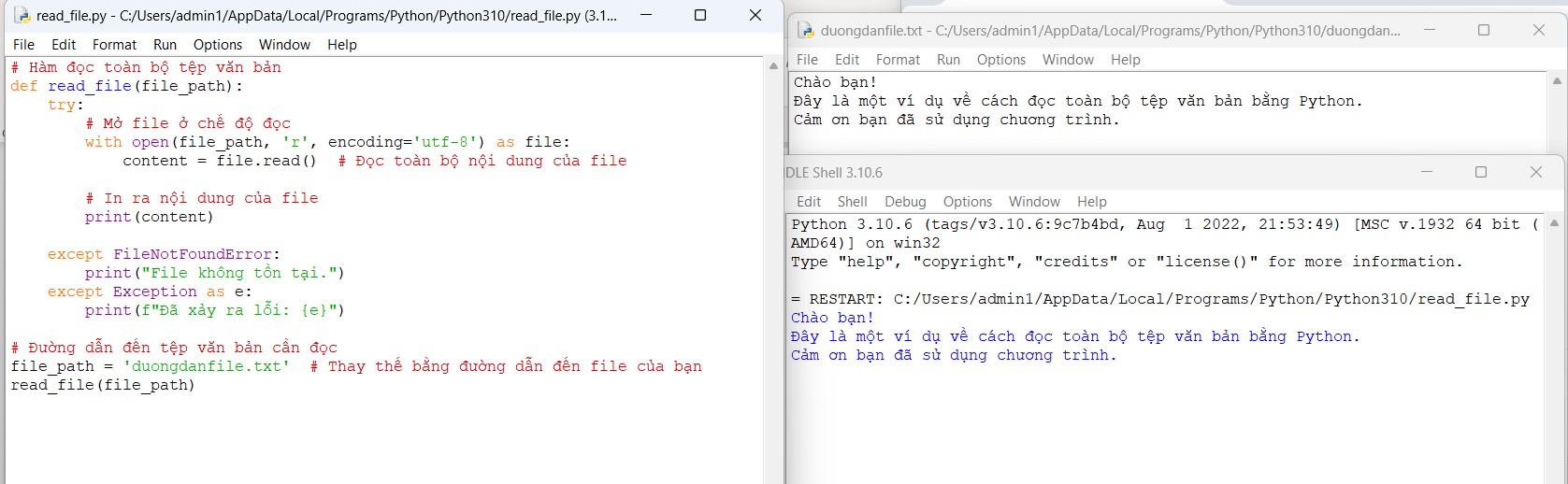
1. Chương trình đọc file và in đảo ngược kết quả

****

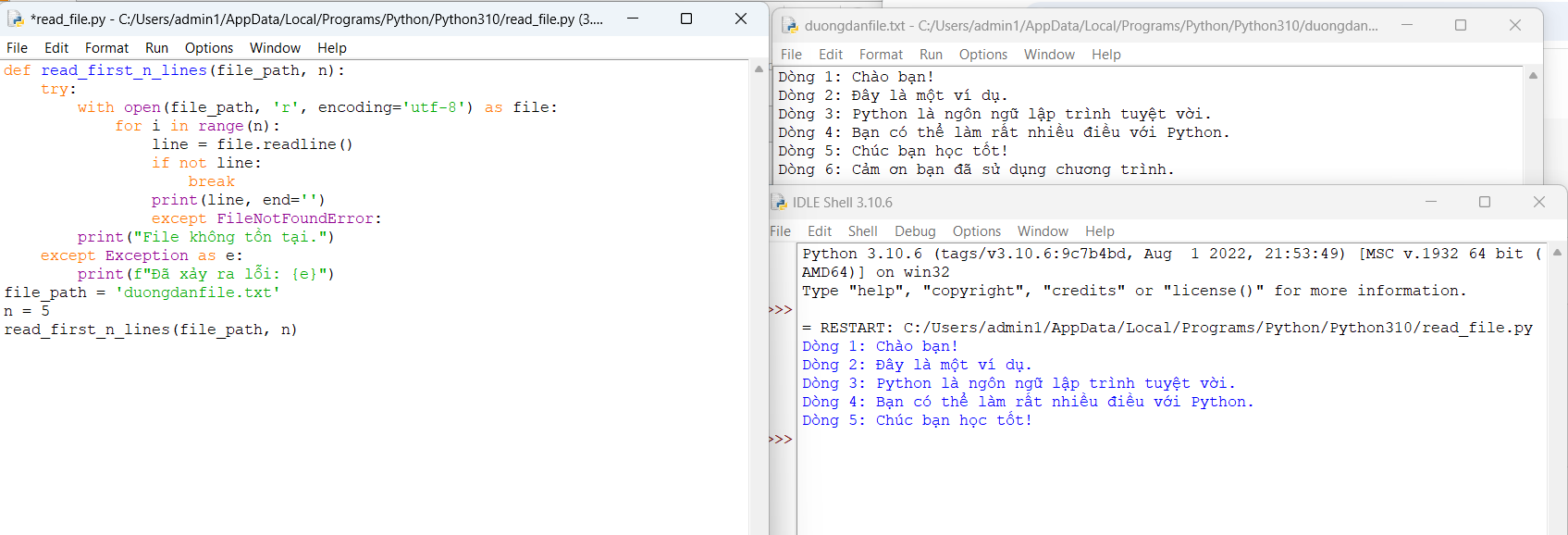
1. Chương trình đọc một file, tính số ký tự, số từ và số dòng của file.



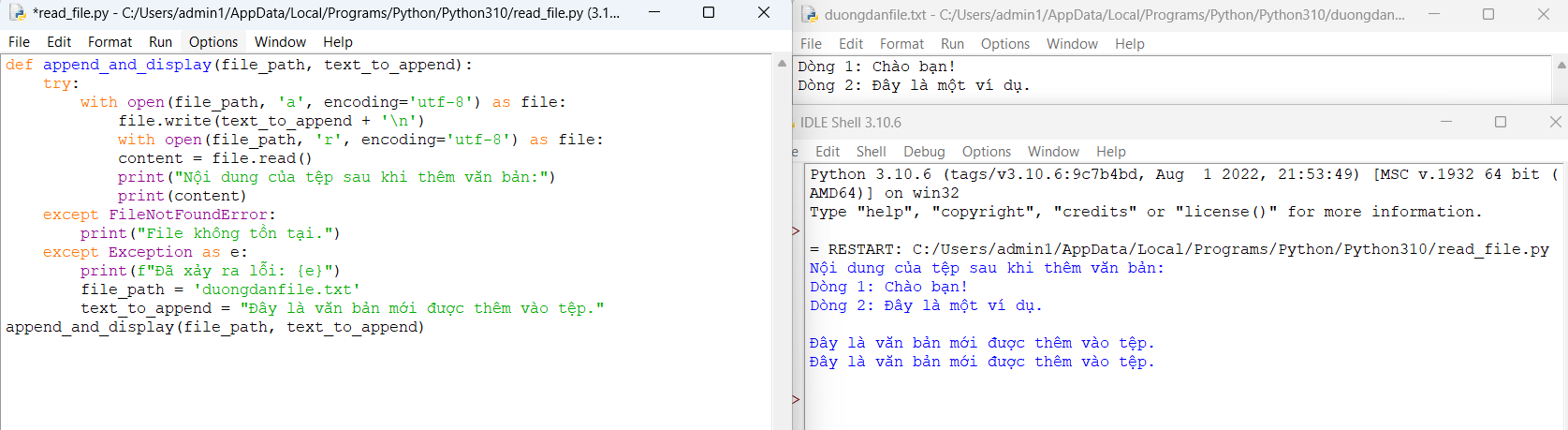
1. Viết chương trình Python để đọc toàn bộ tệp văn bản



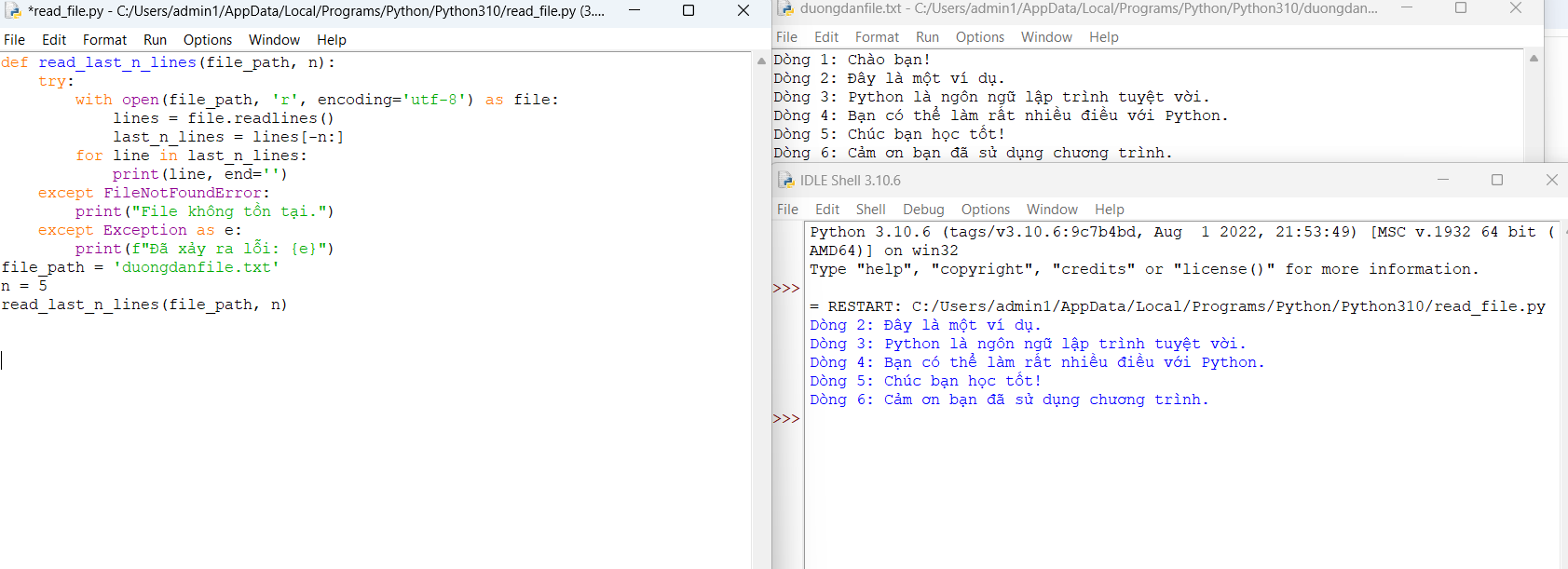
1. Chương trình Python để đọc n dòng đầu tiên của tệp



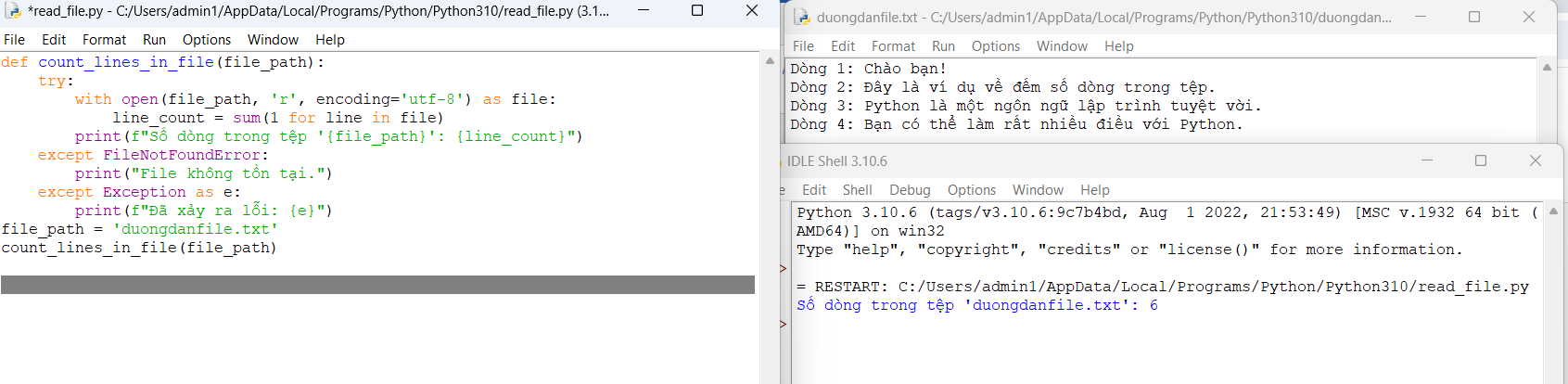
1. Chương trình Python để nối văn bản vào tệp và hiển thị văn bản



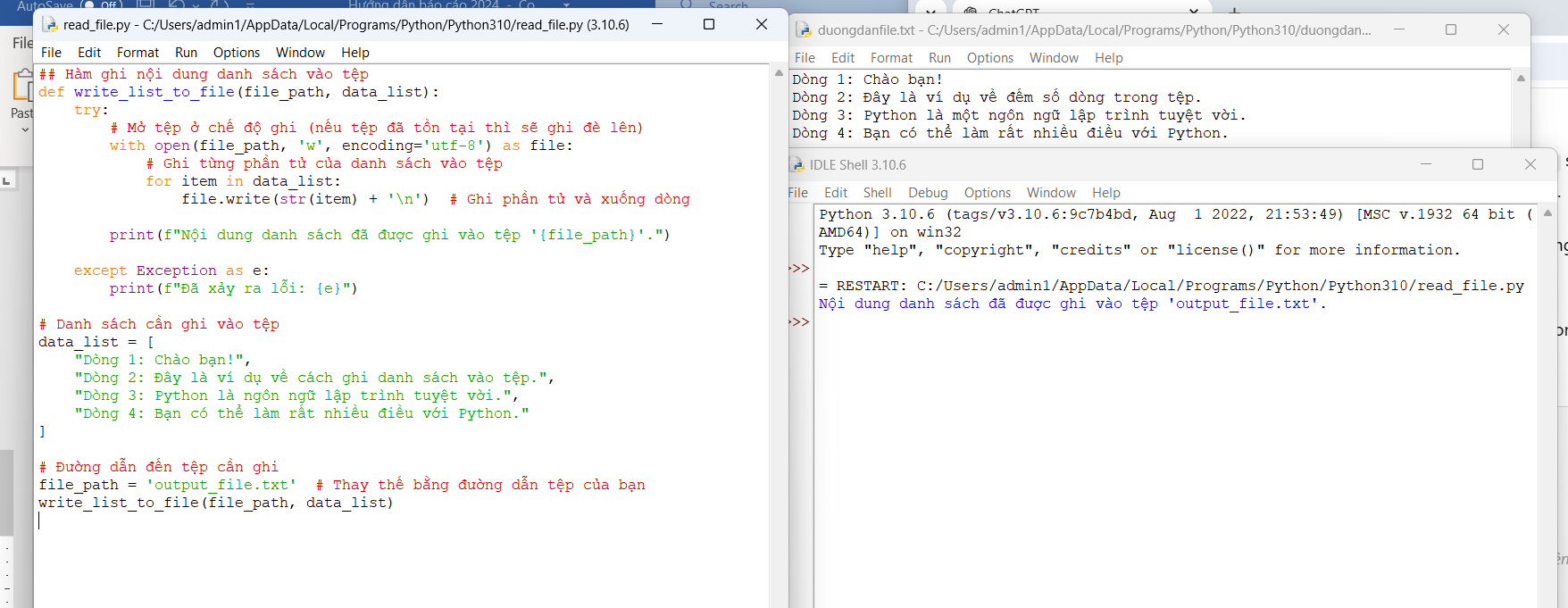
1. Chương trình Python để đọc n dòng cuối cùng của tệp



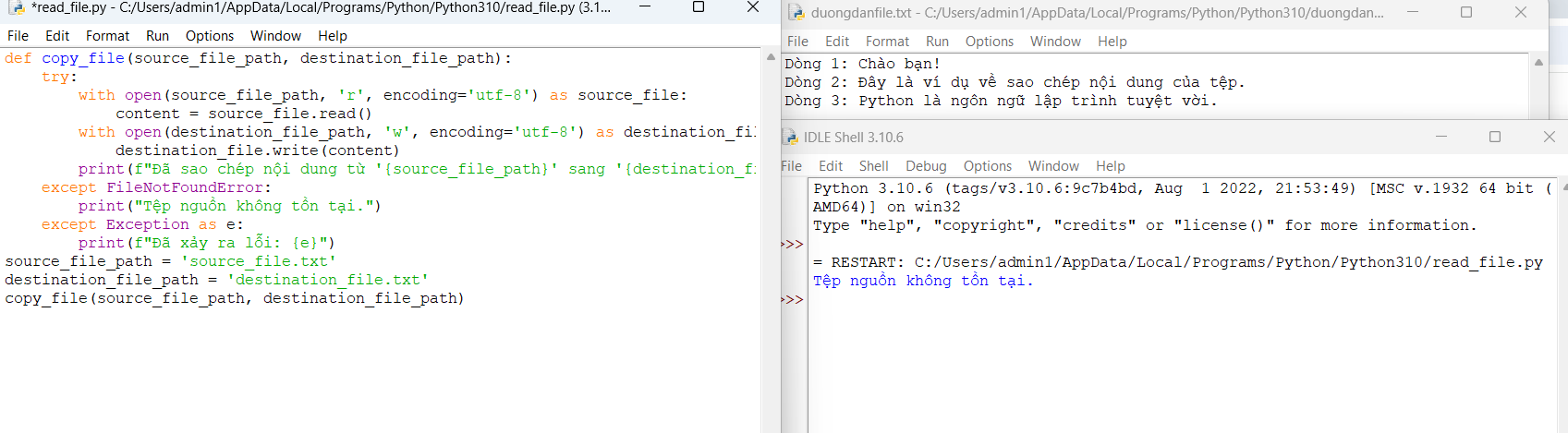
1. Viết chương trình Python để đếm số dòng trong tệp văn bản



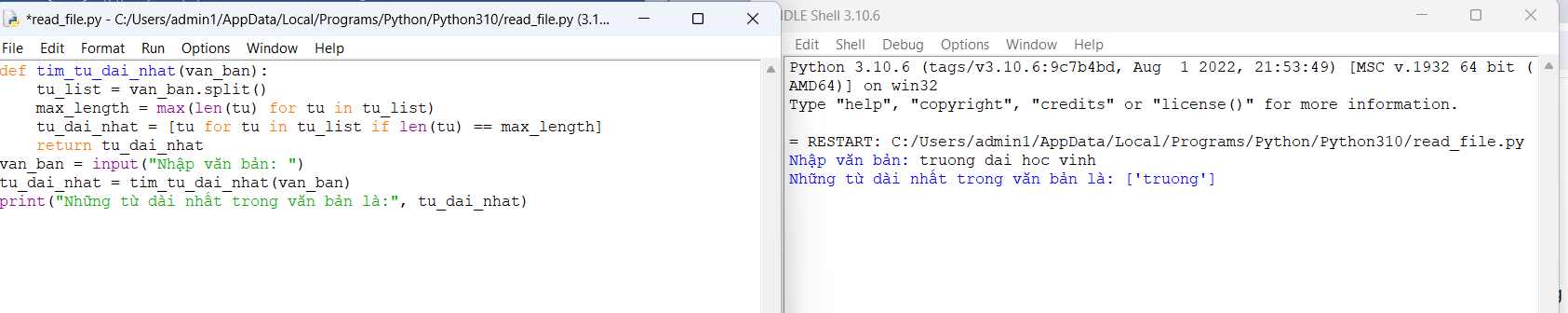
1. Viết chương trình Python để viết nội dung danh sách vào tệp



1. Viết chương trình Python để sao chép nội dung của tệp này sang tệp khác



1. Viết chương trình python để tìm những từ dài nhất trong văn bản



**1.5. Câu hỏi kiểm tra**

- Giải thích chương trình đã viết.

- Các thao tác với file văn bản.

**1.6. Tài liệu tham khảo**

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

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

**1.1. Mục đích**

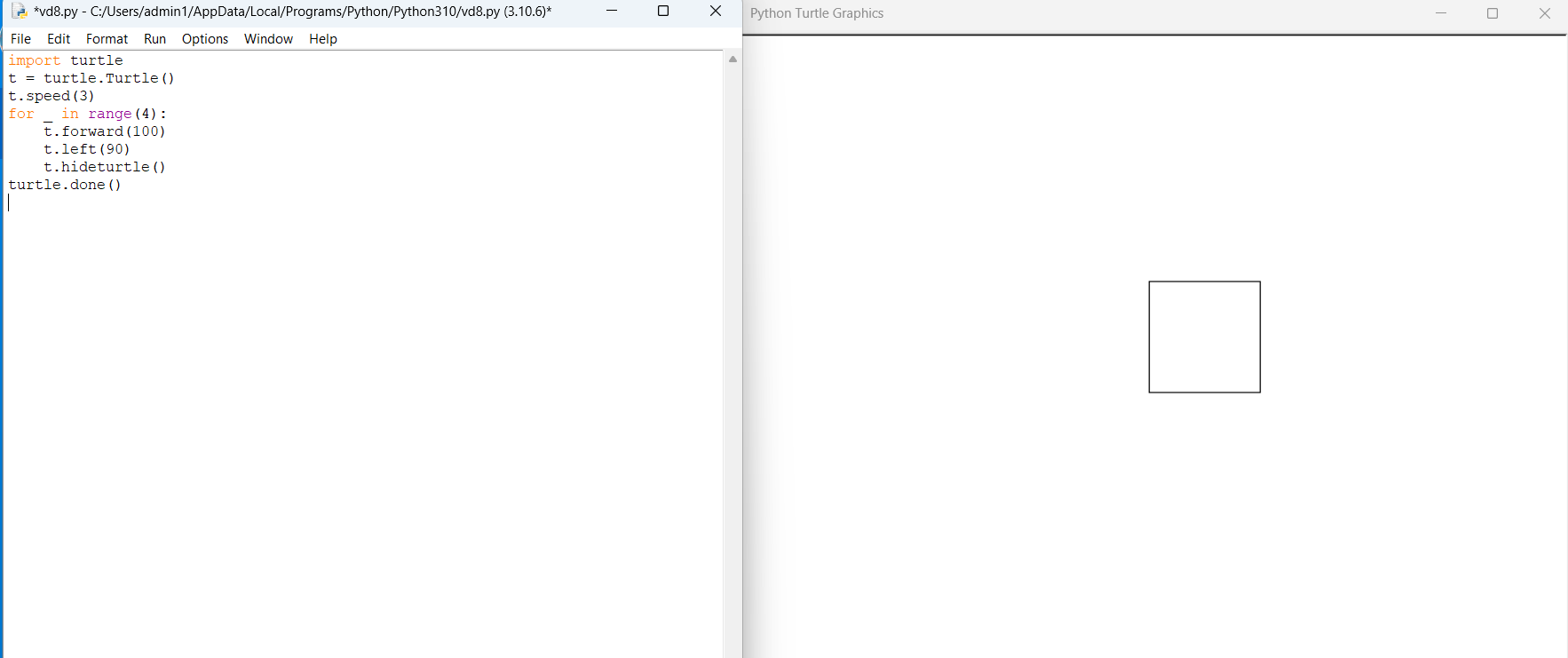
Giúp sinh viên làm quen với lập trình đồ họa và sử dụng thư viện turtle, Tkinter trong python.

**1.2.Các bước tiến hành và kết quả**

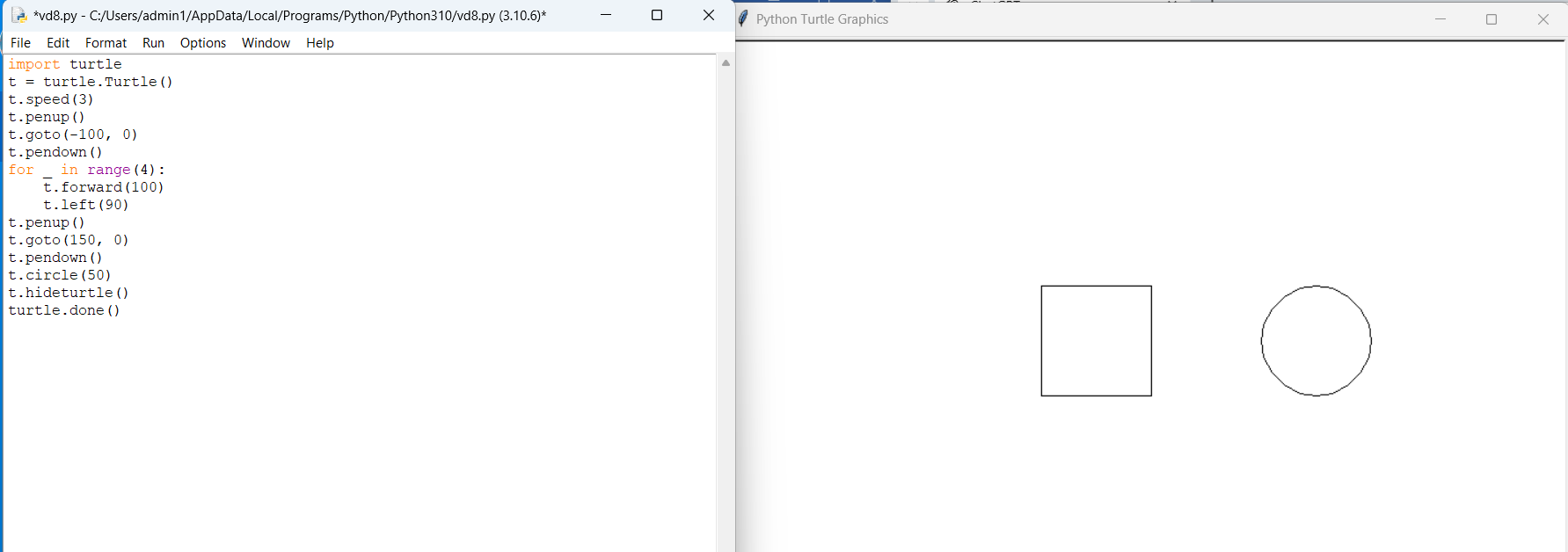
Xem các quy tắc xây dựng và quản lý layout, widget, hộp thoại, đồ họa trong

Python

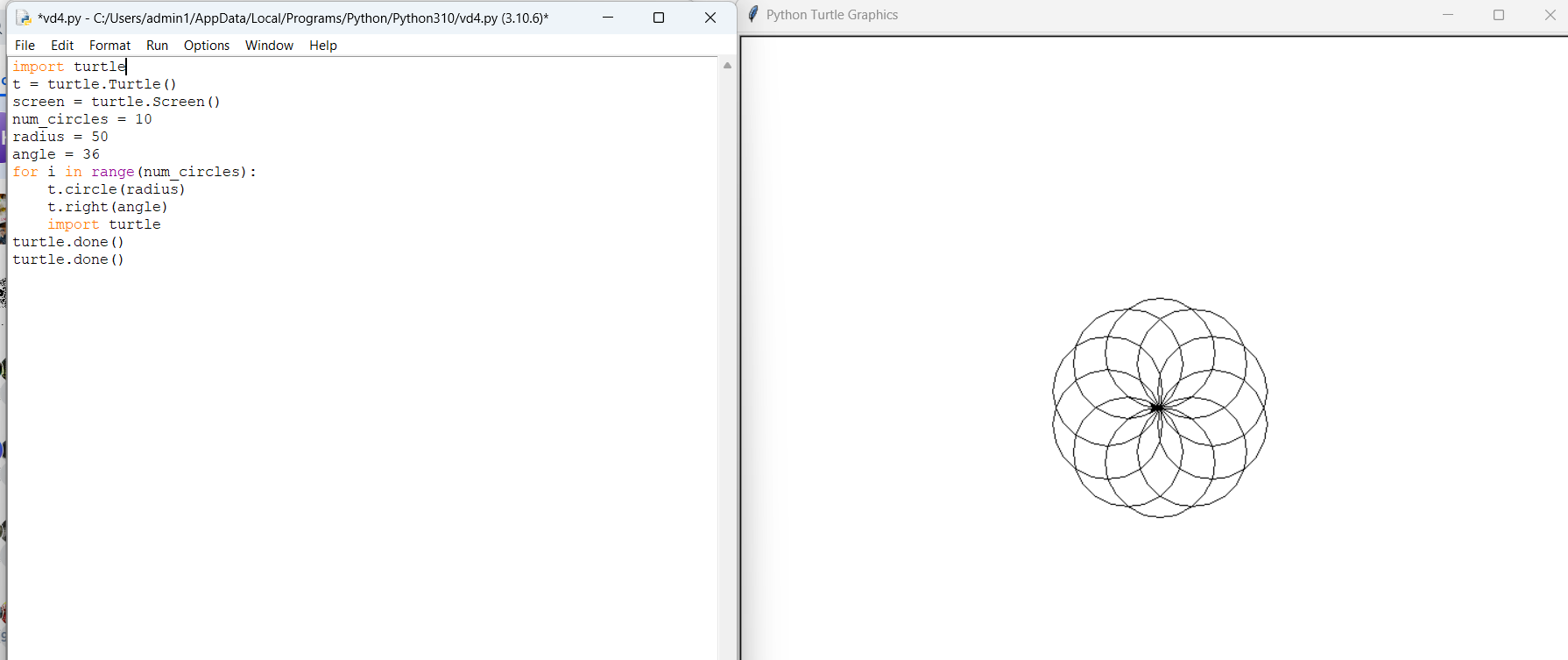
1. Viết chương trình đồ họa sử dụng thư viện turtle, kiểm tra kết quả và giải thích chương trình



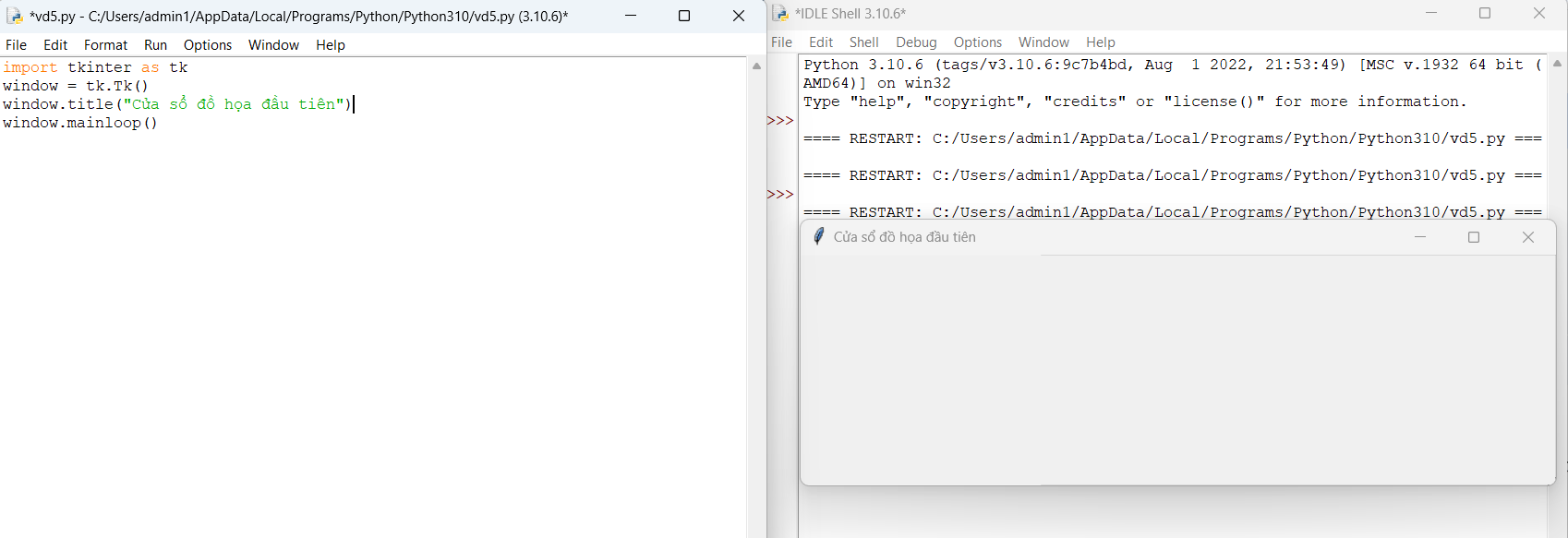
1. Viết chương trình đồ họa sử dụng thư viện turtle, kiểm tra kết quả và giải thích chương trình



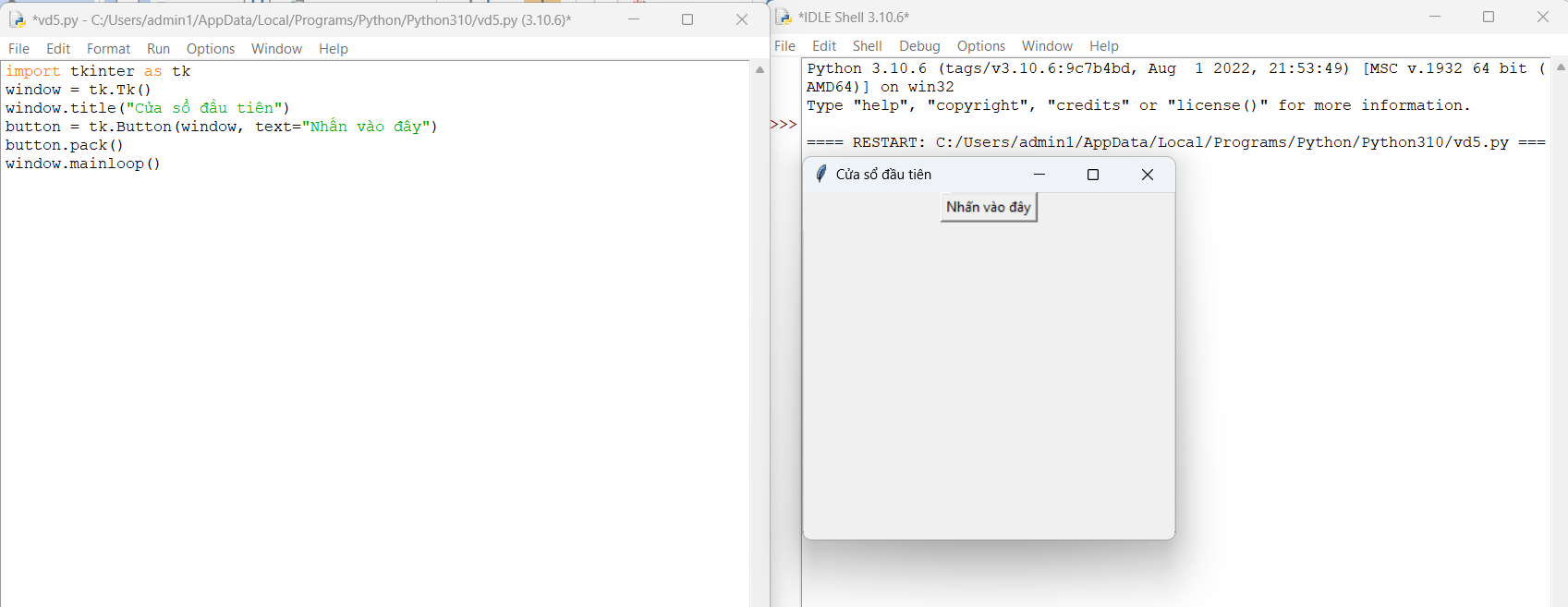
1. Dựa trên các kết quả đạt được từ các chương trình trên hãy viết chương trình hiển thị hình ảnh đồ họa sau



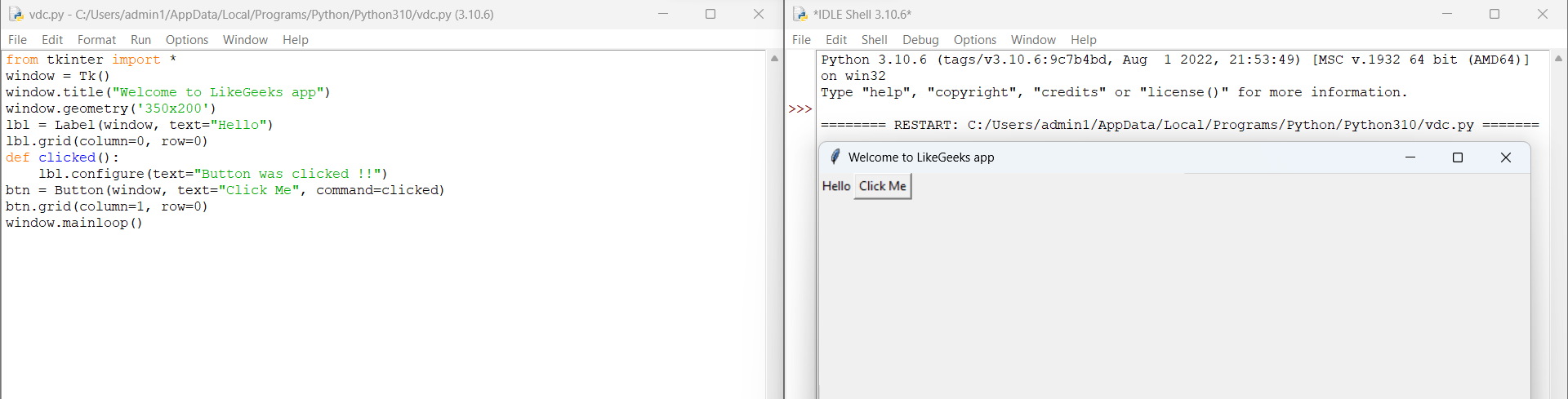
1. Viết chương sử dụng thư viện đồ họa tkinter thực hiện:
2. Xây dựng cửa sổ đồ họa window form



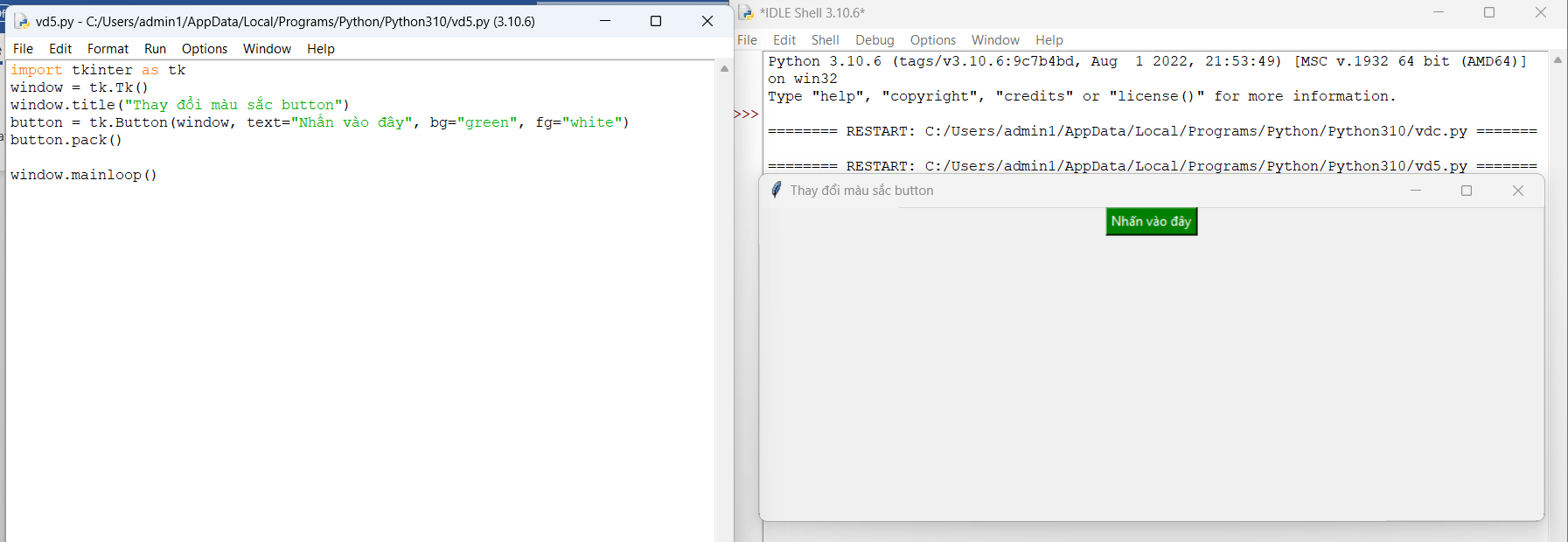
1. Thêm một widget (button) vào window form



1. Xây dựng phương thức xử lý sự kiện phím bấm

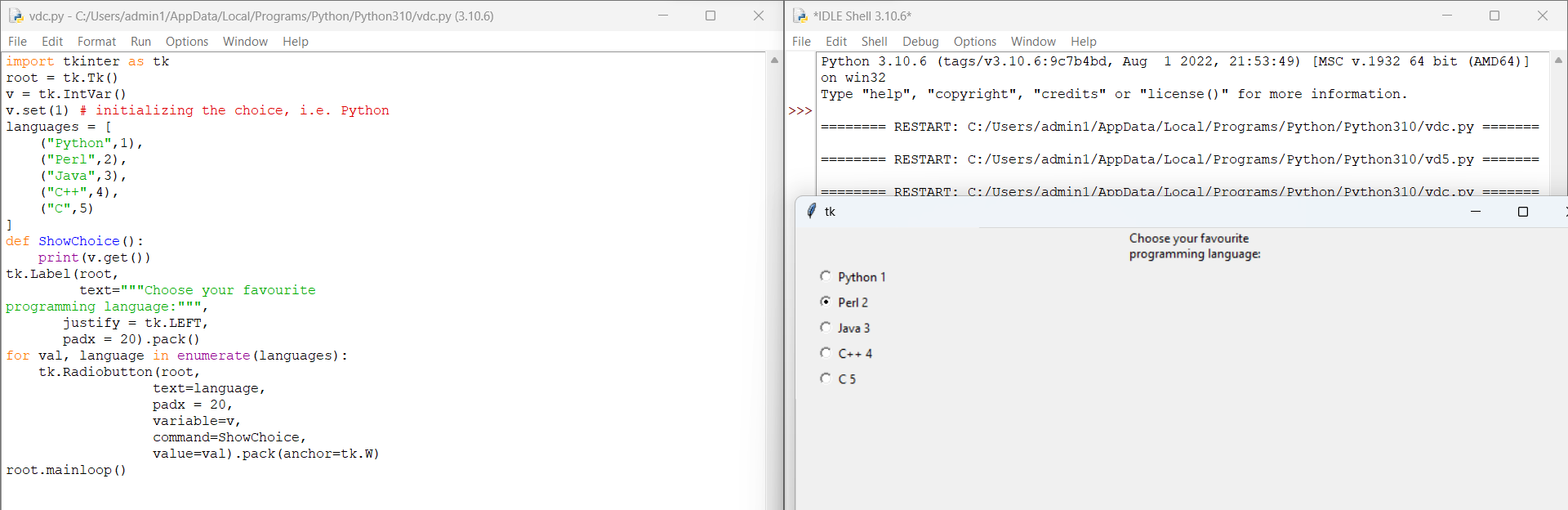


1. Thay đổi màu nền và màu chữ của button sử dụng thuộc tính “bg” và “fg”

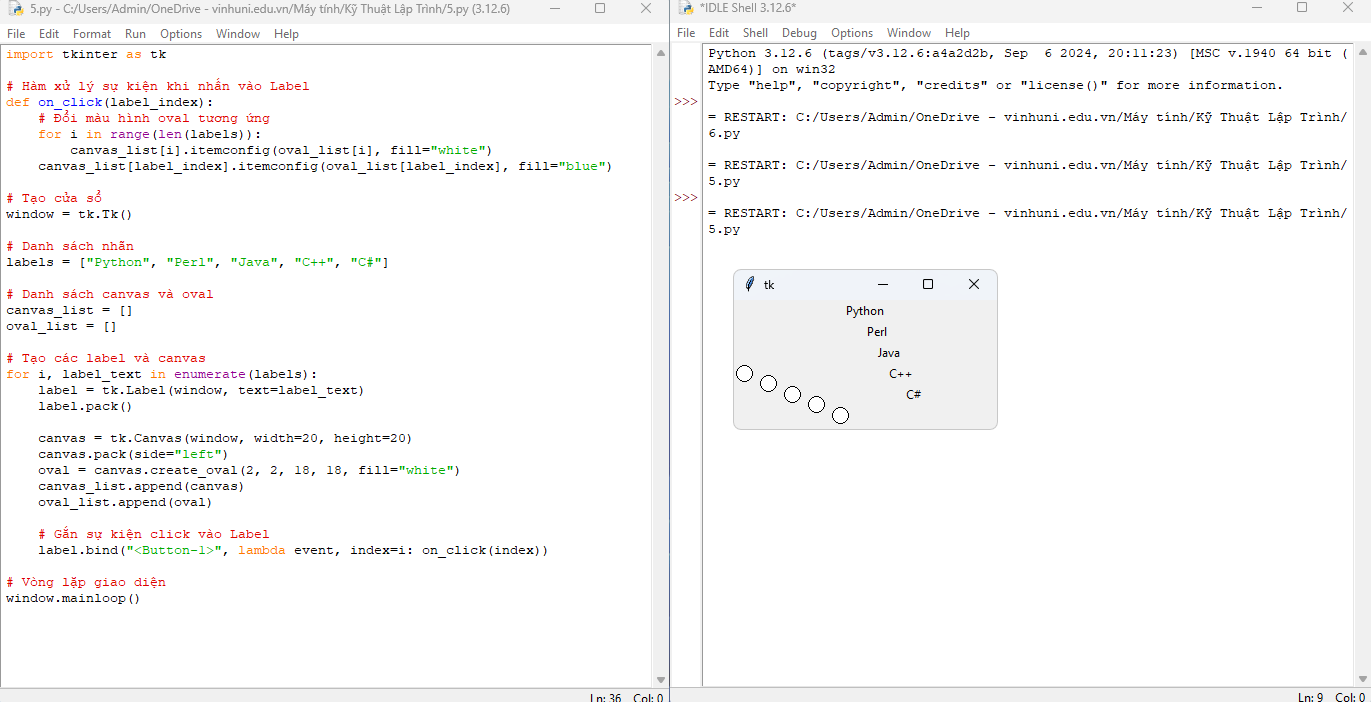


1. Sử dụng thư viện tkinter thực hiện:

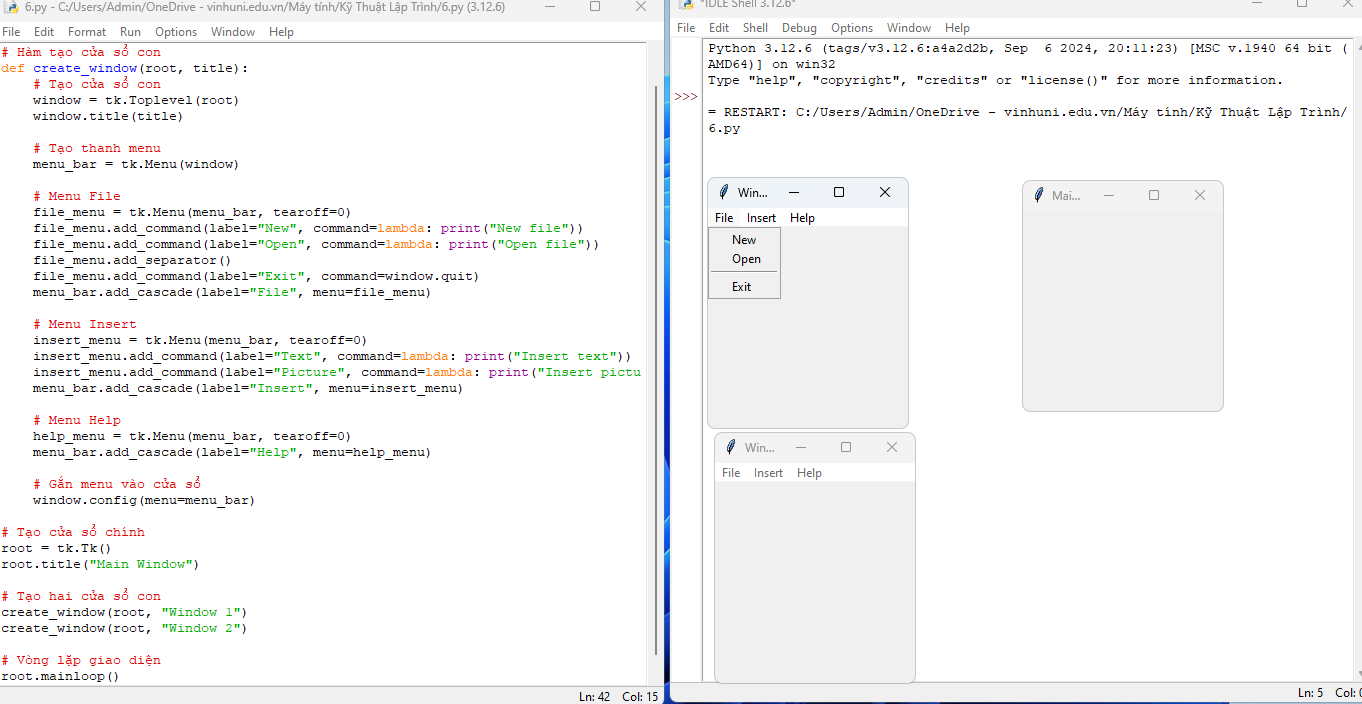
a) Xây dựng các radio button cho phép thực hiện các lựa chọn khác nhau

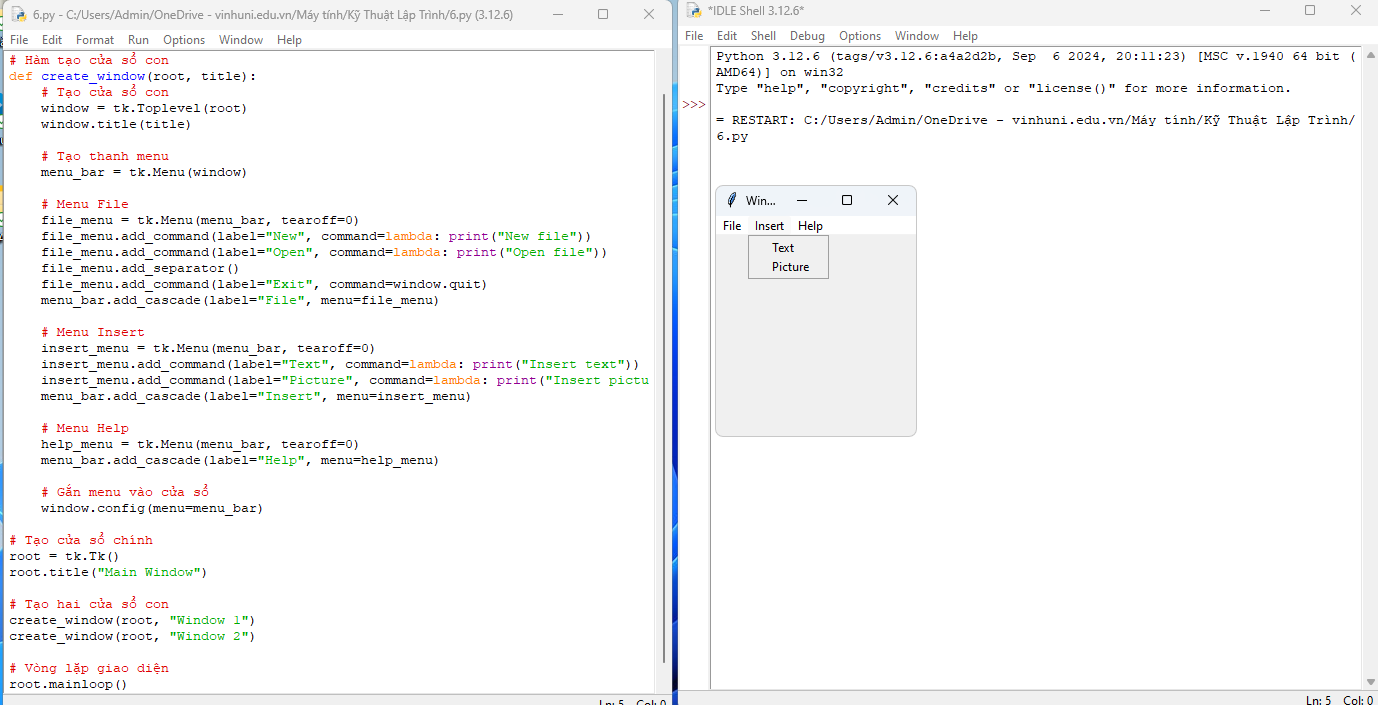


b) Thay thế các radio button thành các indicator như hình

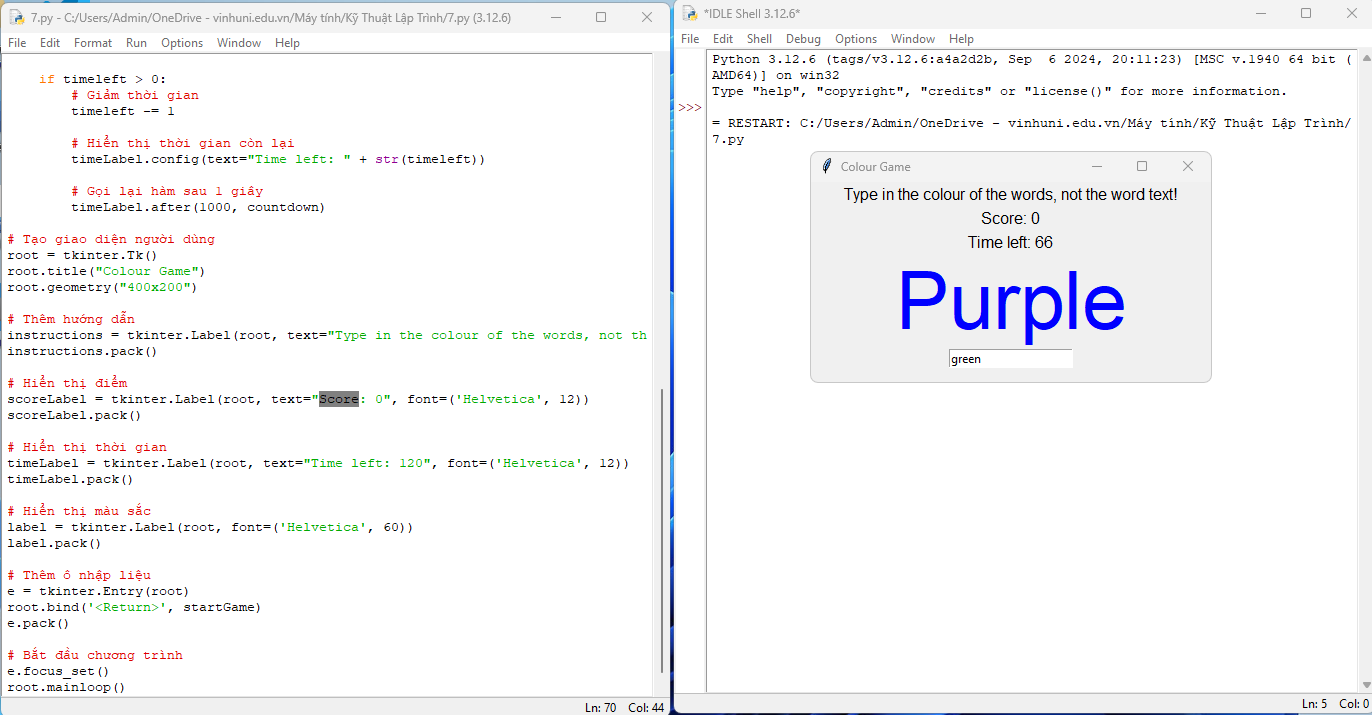


1. Viết chương trình thực hiện tạo menu theo các bước sau:



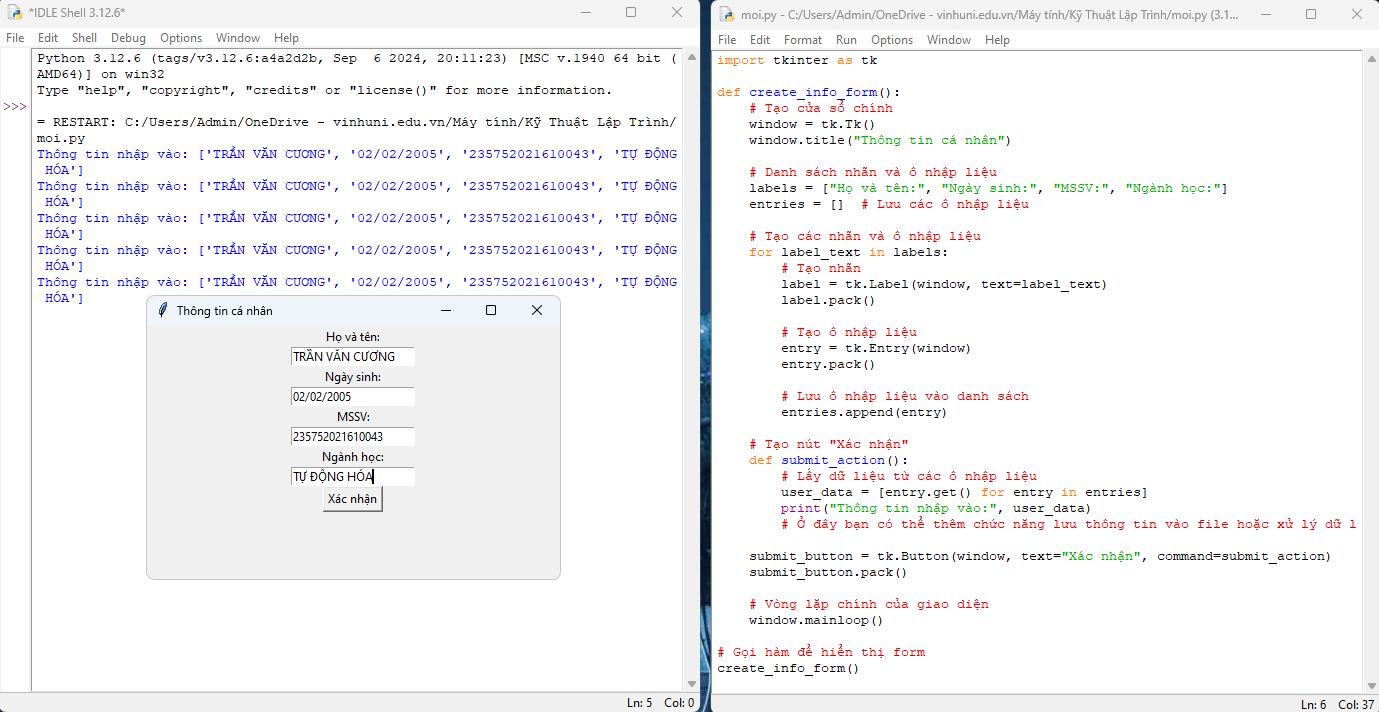


1. Sử dụng thư viện đồ họa tkinter thực hiện các bước sau

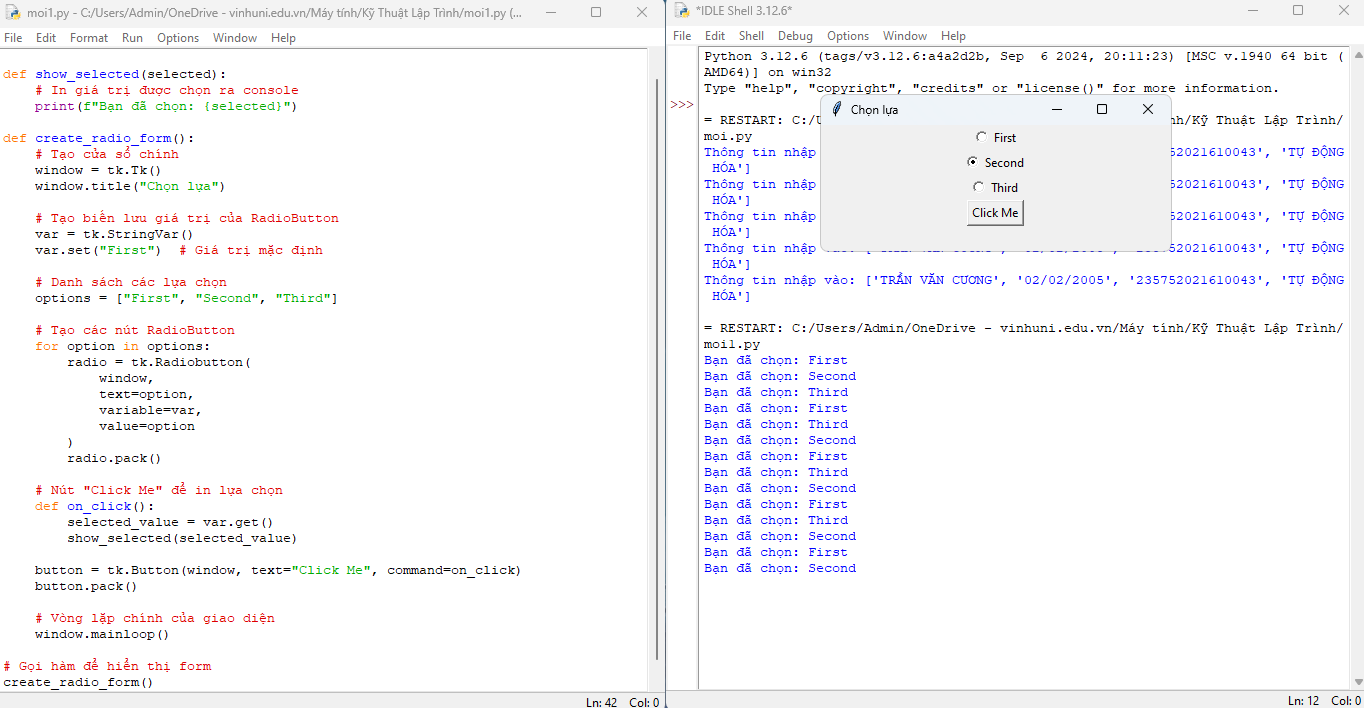


1. Viết chương trình graphic sử dụng thư viện Tkinter thực hiện:

a) Xây dựng form hiển thị thôn tin cá nhân (họ tên, ngày tháng năm sinh, MSSV,ngành học)



b) Xây dựng 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 (tương ứng với các số 1,2, 3)



1.5. Câu hỏi kiểm tra

- Giải thích chương trình đã thực hiện.

- Sử dụng các hàm trong thư viện turtle, tkinter để xây dựng các ứng dụng graphic

với python.

1.6. Tài liệu tham khảo

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

[2]. https://likegeeks.com/python-gui-examples-tkinter-tutorial/