TRƯỜNG ĐẠI HỌC VINH

# **VIỆN KỸ THUẬT CÔNG NGHỆ**



**BÁO CÁO THỰC HÀNH**

**KỸ THUẬT LẬP TRÌNH**

**MÃ HỌC PHẦN: ELE20004**

SVTH: Lê Văn An

MSSV: 235752021610044

GVHD: MAI THẾ ANH

**NGHỆ AN - 2024**

**MỤC LỤC**

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

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

Bài 3. Lập trình hàm trong Python

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

Bài 5. Thiết kế module trong Python

Bài 6: Lập trình hướng đối tượng trong Python

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

Bài 8: Lập trình giao diện trong Python

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

### 1.1. 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

### 1.2. Cơ sở lý thuyết

#### 1.2.1. Giới thiệu Flowgorithm

Flowgorithm là ngôn ngữ lập trình dành cho người mới bắt đầu miễn phí dựa trên sơ đồ đồ họa đơn giản.

Thông thường, khi một sinh viên lần đầu tiên học lập trình, họ thường sử dụng một trong những ngôn ngữ lập trình dựa trên văn bản. Tùy thuộc vào ngôn ngữ, điều này có thể dễ dàng hoặc khó khăn. Nhiều ngôn ngữ yêu cầu bạn viết các dòng mã khó hiểu chỉ để hiển thị văn bản "Xin chào, thế giới!".

Bằng cách sử dụng sơ đồ, bạn có thể tập trung vào các khái niệm lập trình hơn là tất cả các sắc thái của ngôn ngữ lập trình thông thường. Bạn cũng có thể chạy các chương trình của mình trực tiếp trong Flowgorithm.

Một khi bạn hiểu logic lập trình, bạn sẽ dễ dàng học một trong những ngôn ngữ chính. Flowgorithm có thể tương tác chuyển đổi sơ đồ của bạn sang hơn 18 ngôn ngữ. Chúng bao gồm: C #, C ++, Java, JavaScript, Lua, Perl, Python, Ruby, Swift, Visual Basic .NET và VBA (được sử dụng trong Office).

#### 1.2.2. Đặc tính của Flowgorithm

* Đầu ra dễ hiểu
* Cửa sổ xem biến đồ họa
* Tương tác tạo mã thật - Phối màu tùy chỉnh
* Hỗ trợ đa ngôn ngữ
* Hỗ trợ đầy đủ các toán tử và hơn 20 hàm cơ bản

#### 1.2.3. Các cửa sổ chính khi làm việc với Flowgorithm

1. *Giao diện chính*

A diagram of a main and end

Description automatically generated

Hình 1. Main window

1. *Cửa sổ điều khiển*

A screenshot of a computer

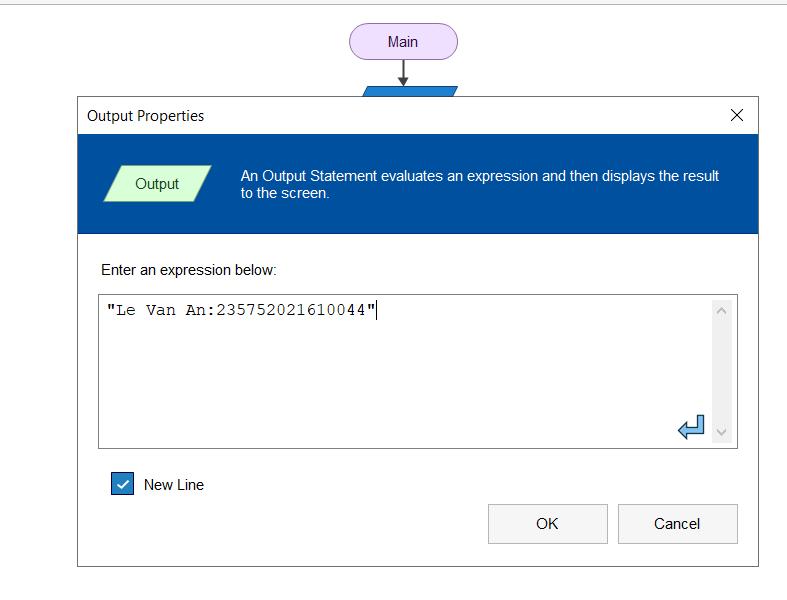
Description automatically generatedA diagram of a process

Description automatically generated

Hình 2. Console Window

1. *Giao diện mã nguồn*

Trình xem mã nguồn có thể chuyển đổi sơ đồ của bạn sang một số ngôn ngữ lập trình chính.

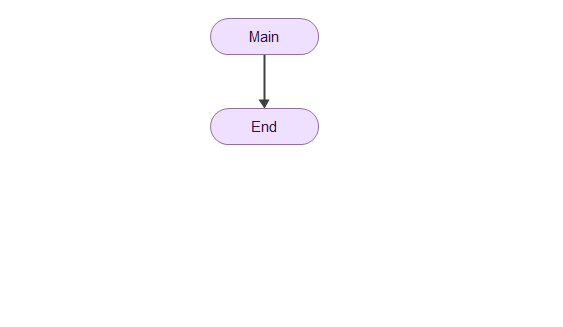


1. *Giao diện xem các biến của chương trình*

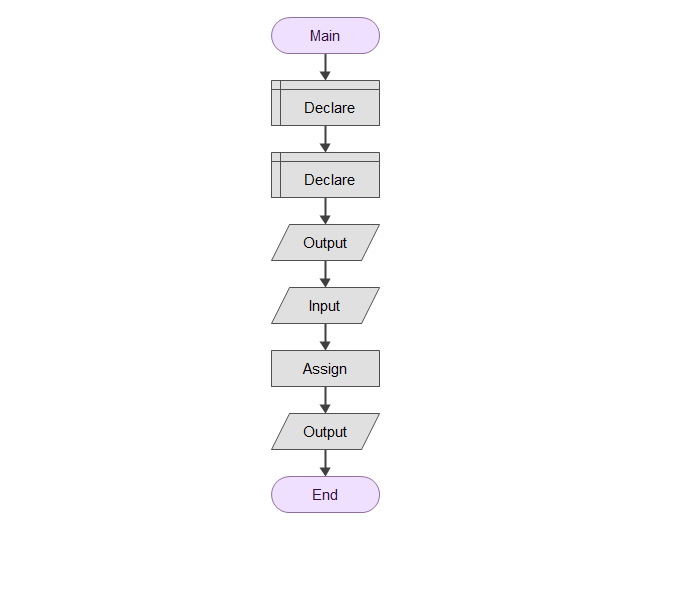
*Câu 2*

a.Tính diện tích hình tròn.

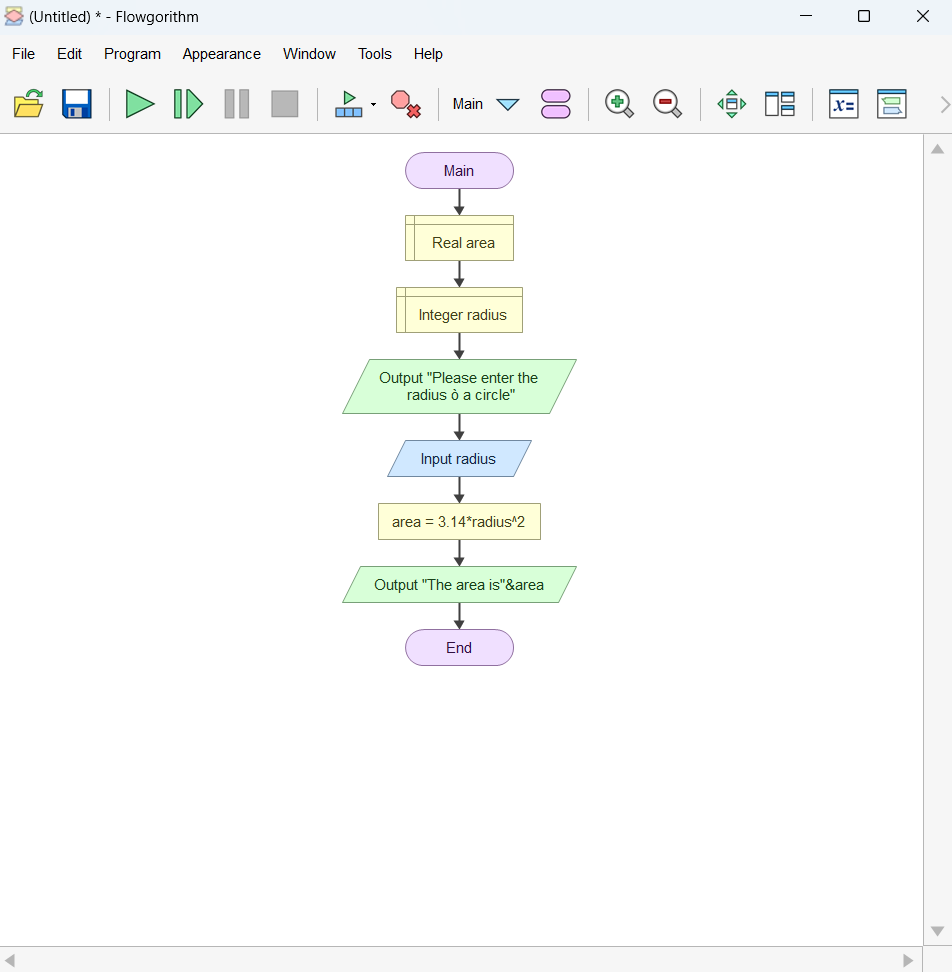
B1:Tạo chương trình mới.



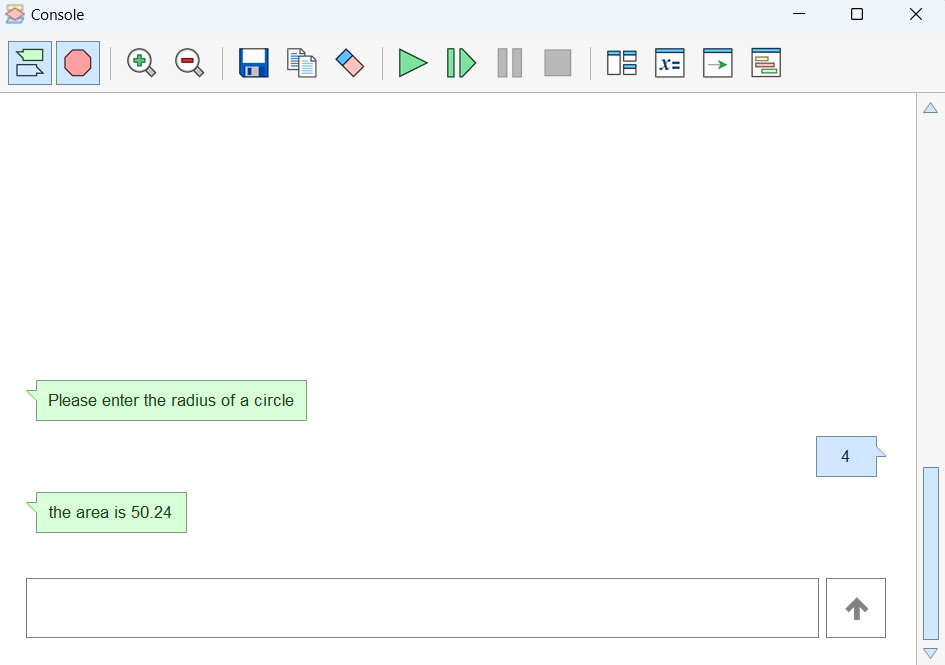
B2:Thêm các khối chức năng.



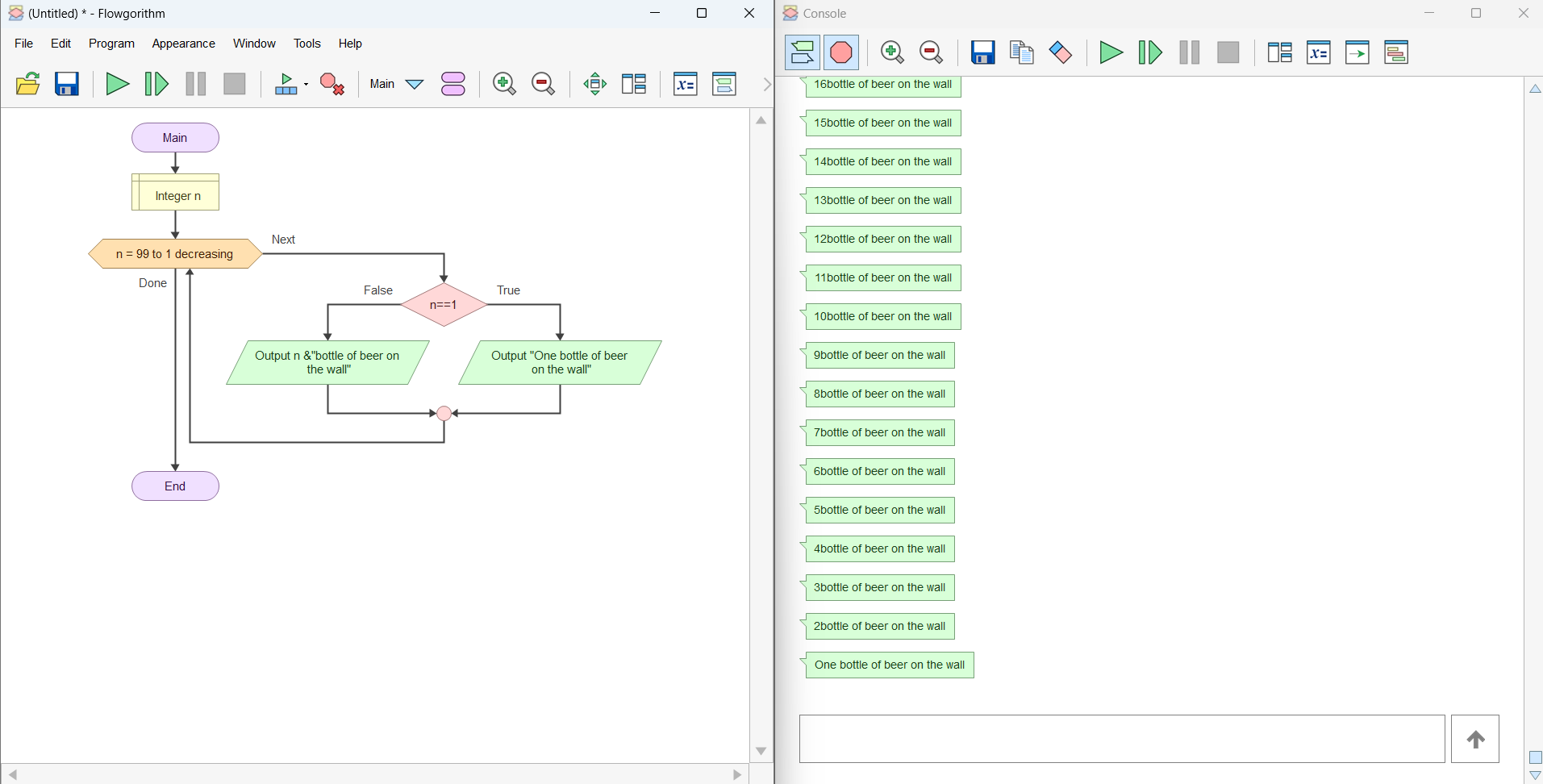
B3: Định nghĩa chức năng cho khối.

**

B4:Khởi chạy chương trình đã thiết kế.

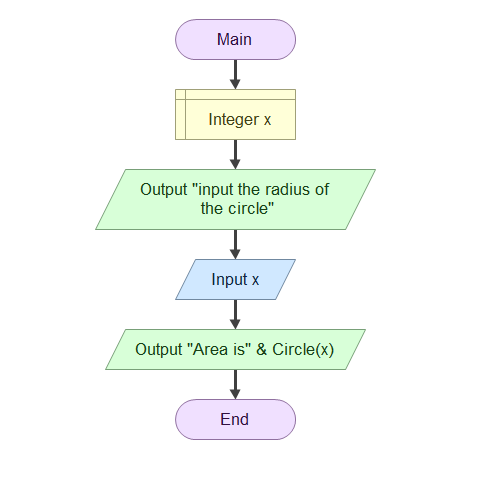


b.Nhập và in số lượng chai sử dụng vòng lặp for.

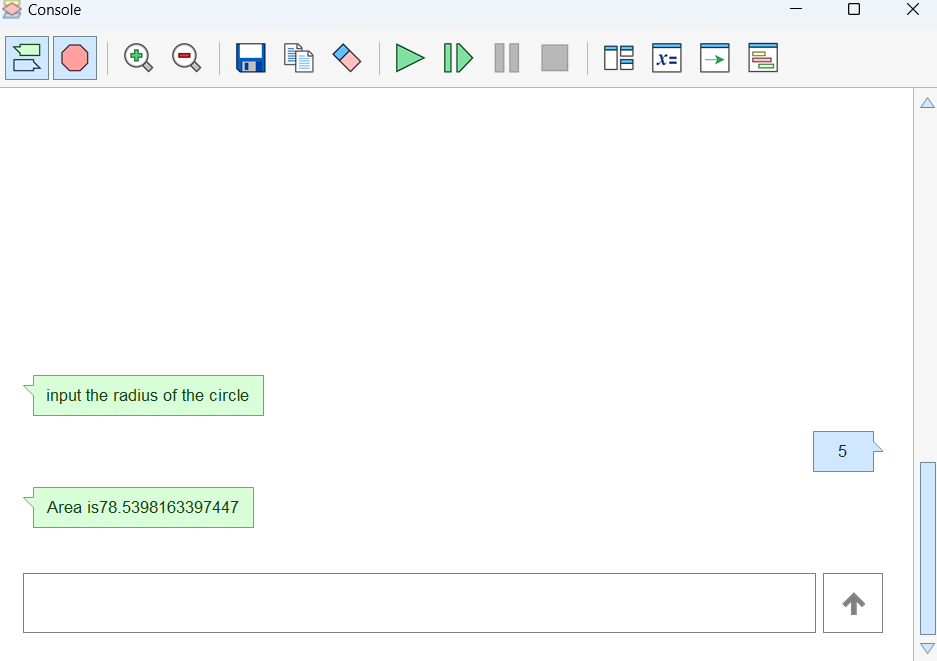


c.Tính diễn tích hình tròn sử dụng hàm (chương trình con).

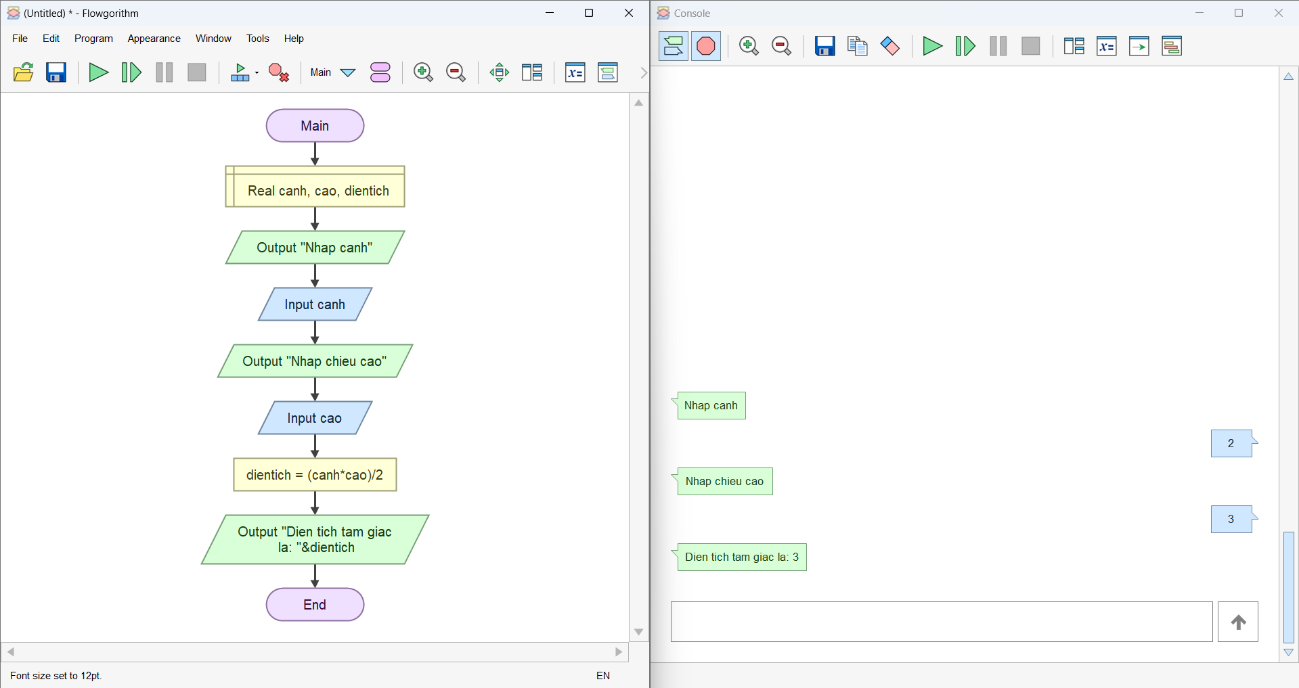
B1:Định nghĩa chức năng cho các khối.

 A diagram of a circle and a circle

Description automatically generated

B2:Chạy chương trình. 

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



e.Xây dựng thuật toán kiểm tra tính nguyên tố (một số nguyên dương n là số nguyên tố khi chỉ có hai ước là 1 và chính nó)

A screenshot of a computer

Description automatically generated

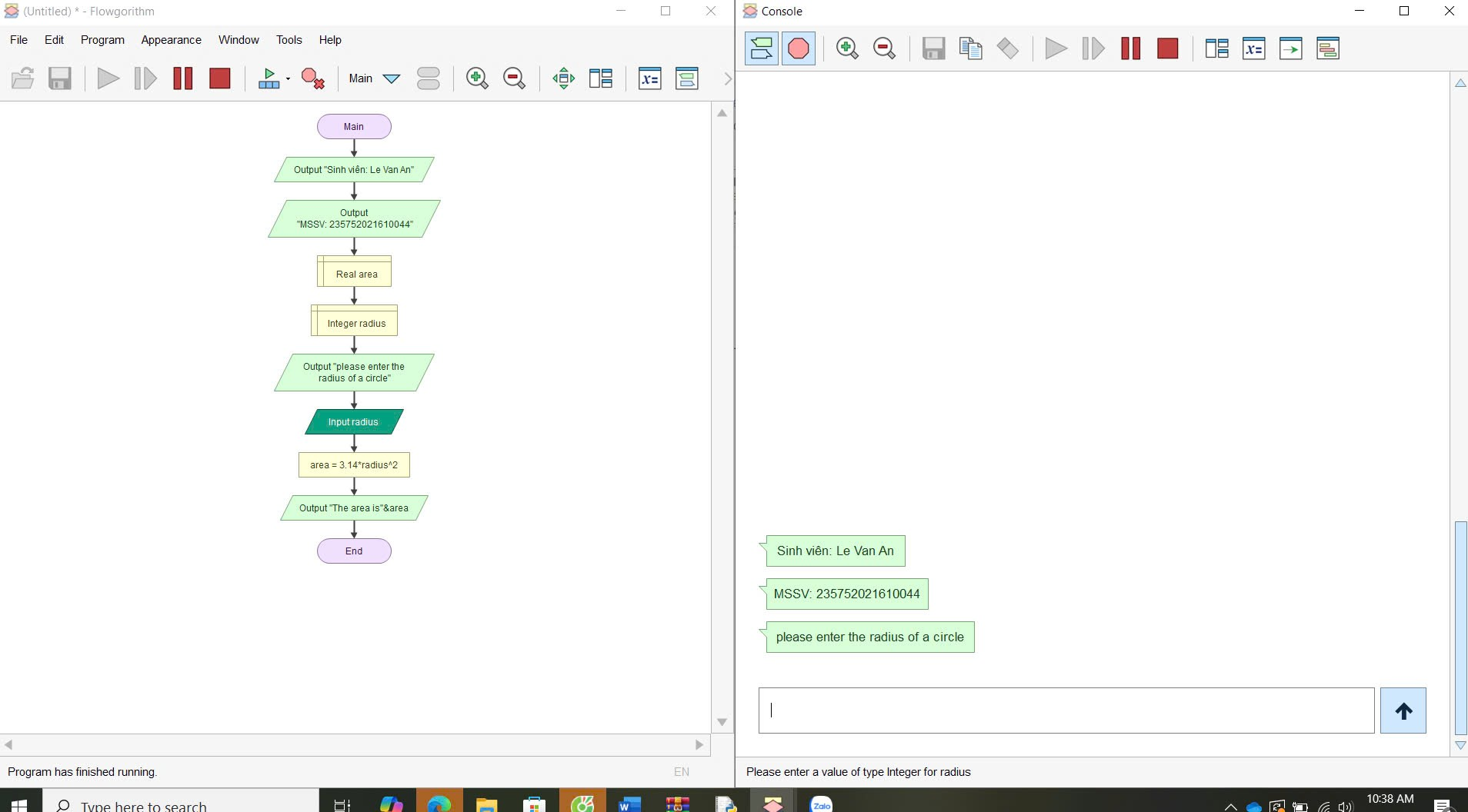
**1.3. Thiết bị thực hành thí nghiệm và vật tư tiêu hao:**

### 1.4. CÁC BƯỚC TIẾN HÀNH THÍ NGHIỆM

#### 1.4.1. Xây dựng thuật toán hiển thị xâu ký tự với Flowgorithm

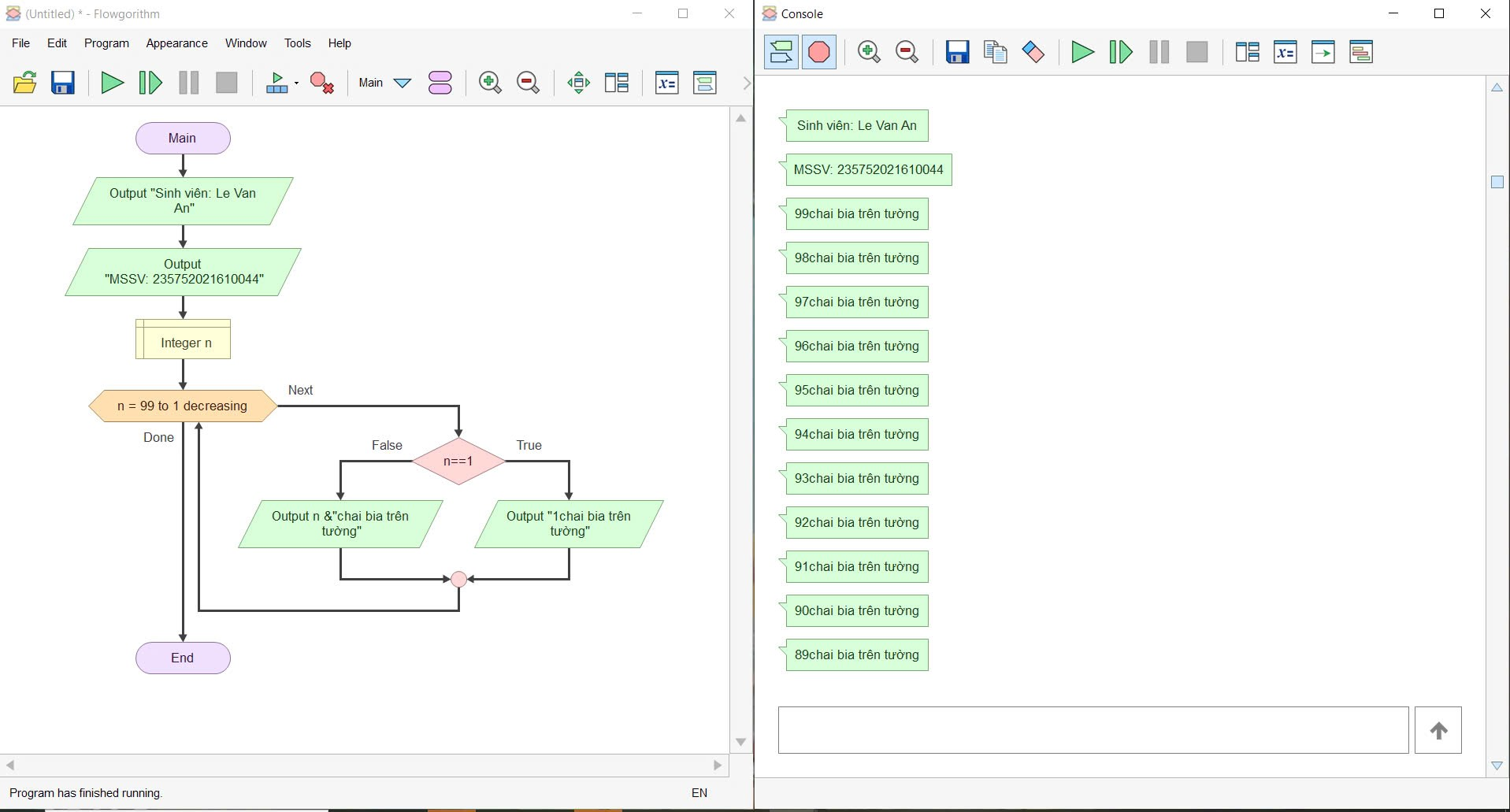
#### 1.4.2. Sử dụng Flowgorithm xây dựng chương trình giải quyết các bài toán

1. *Tính diện tích hình tròn*

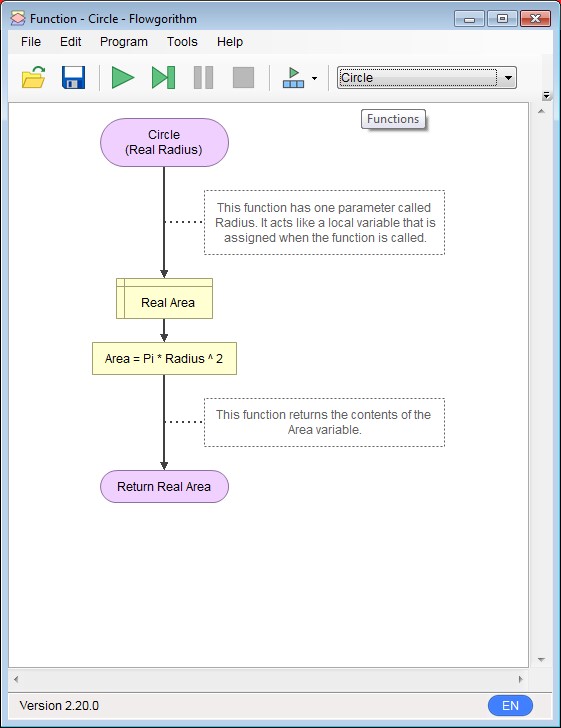
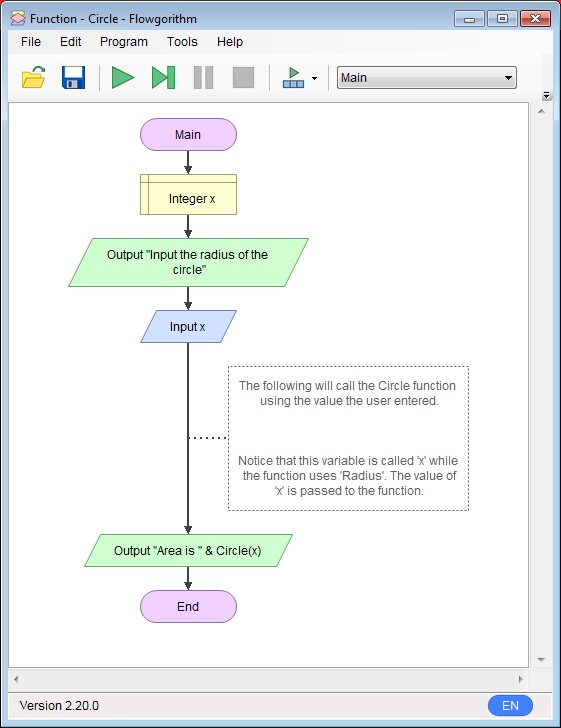


Hình 11. Sơ đồ giải thuật bài toán tính diện tích hình tròn

1. *Nhập và in số lượng chai sử dụng vòng lặp for*

Hình 12. Giải thuật bài toán nhập và in số lượng chai sử dụng vòng lặp while

1. *Tính diện tích hình tròn sử dụng hàm (chương trình con)*



Hình 13. Sử dụng hàm cho giải thuật tính diện tích hình tròn

1. *Viết chương trình nhập và cạnh và chiều cao tương ứng của một tam giác và in ra màn hình diện tích tam giác.*
2. *Xây dựng thuật toán kiểm tra tính nguyên tố (một số nguyên dương n là số nguyên tố khi chỉ có hai ước là 1 và chính nó)* Gợi ý:

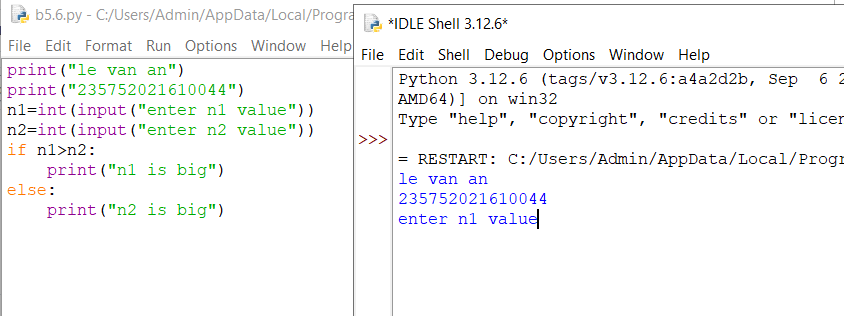
A screenshot of a computer

Description automatically generated

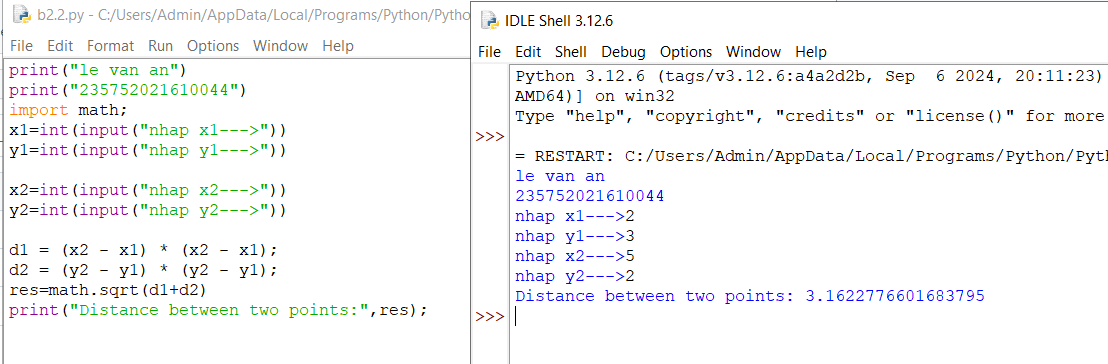
**1.6. Tài liệu tham khảo** <http://www.flowgorithm.org/documentation/index.htm>

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

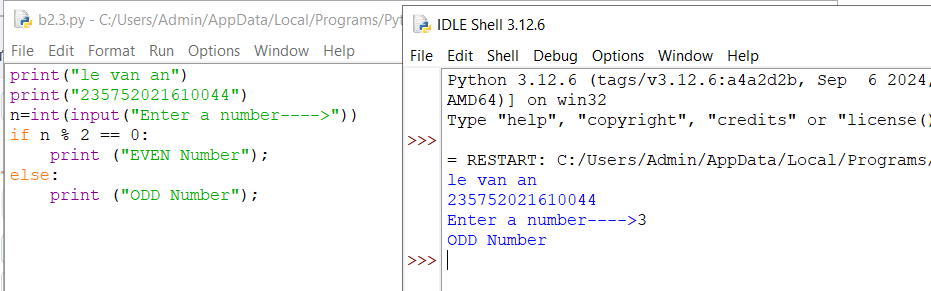
1. Viết đoạn chương trình sau và sửa lỗi



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

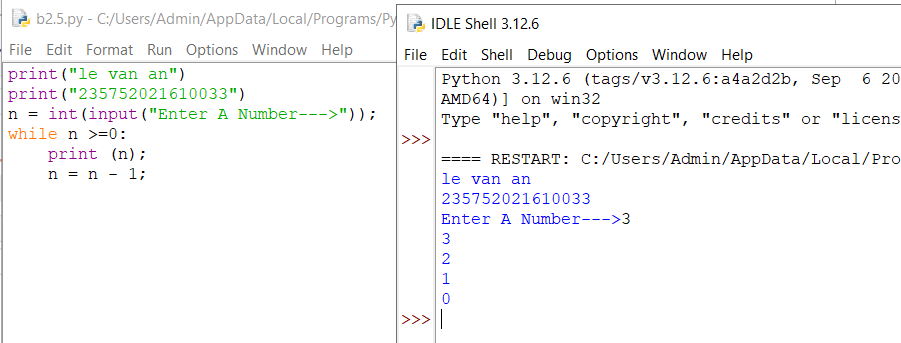


1. Viết chương trình nhập vào một số và kiểm tra số đó là chẵn hay lẻ, in thông báo ra màn hình

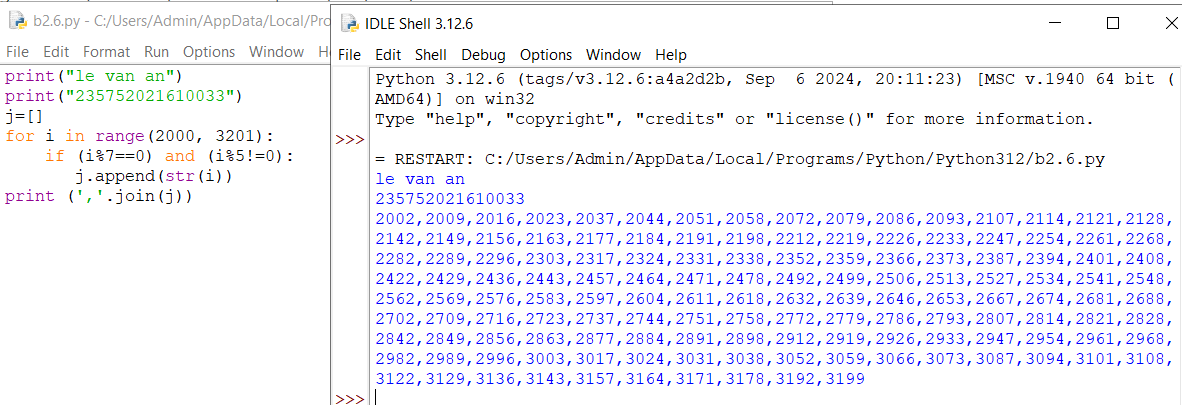


1. Viết chương trình in ra màn hình số nghịch đảo và kết quả dưới dạng thập phân của một dãy số tự nhiên trong khoảng (a,b)



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

1. 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.



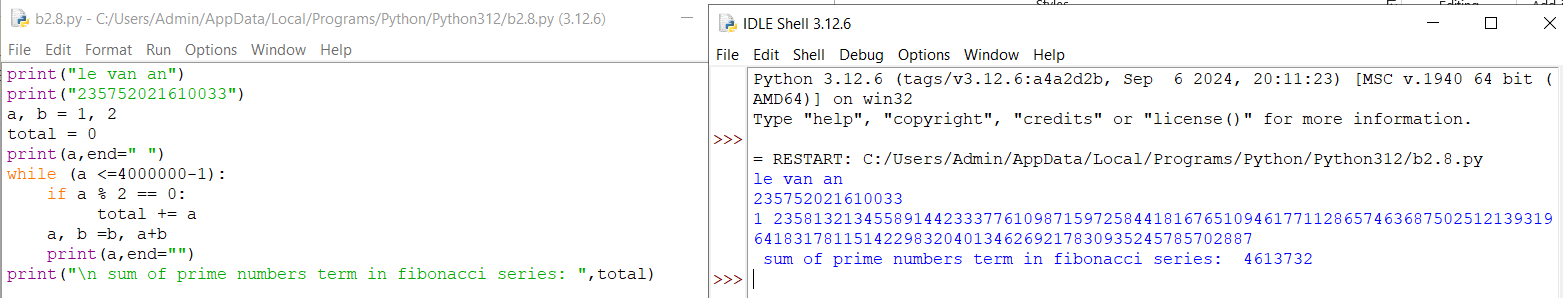
1. Với số nguyên n nhất định, hãy viết chương trình để tạo ra một [dictionary](https://quantrimang.com/gioi-thieu-qua-ve-chuoi-so-list-trong-python-140881) chứa

(i, i\*i) như là số nguyên từ 1 đến n (bao gồm cả 1 và n) sau đó in ra dictionary này. Ví dụ: Giả sử số n là 8 thì đầu ra sẽ là: {1: 1, 2: 4, 3: 9, 4: 16, 5: 25, 6: 36, 7:

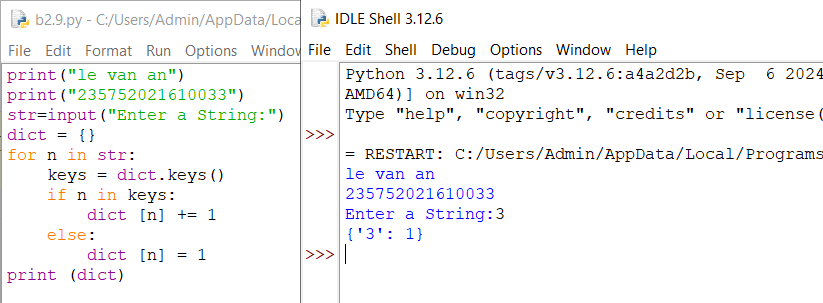
49, 8: 64}.



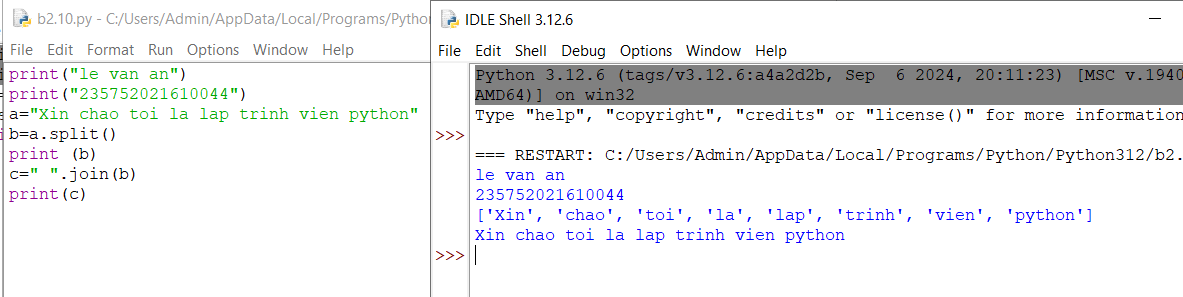
1. Viết chương trình in ra màn hình dãy số Fibonacci nhỏ hơn 4.000.000, tìm tổng các số chẵn trong dãy đã in



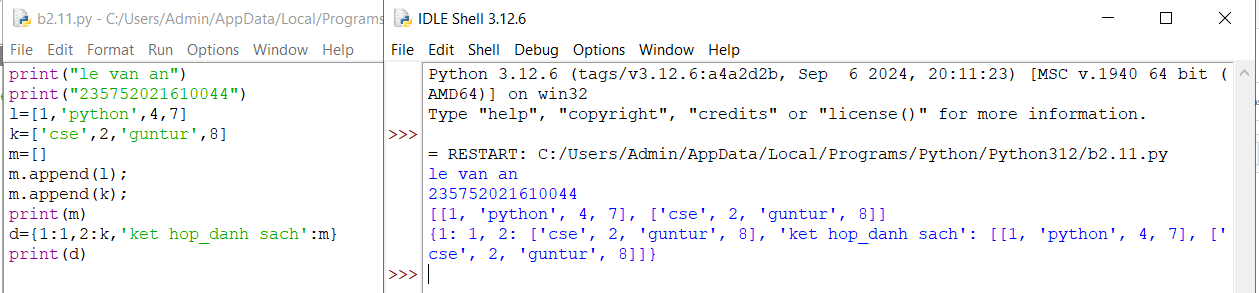
1. Viết chương trình đếm số ký tự trong 1 xâu ký tự nhập vào từ bàn phím, lưu các ký tự vào cấu trúc từ điển



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

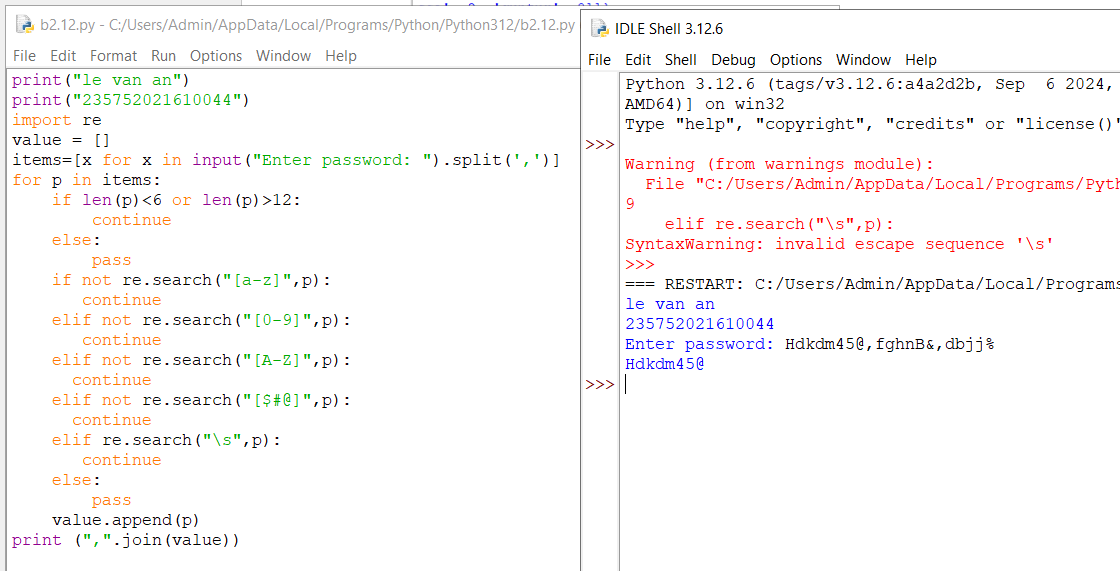


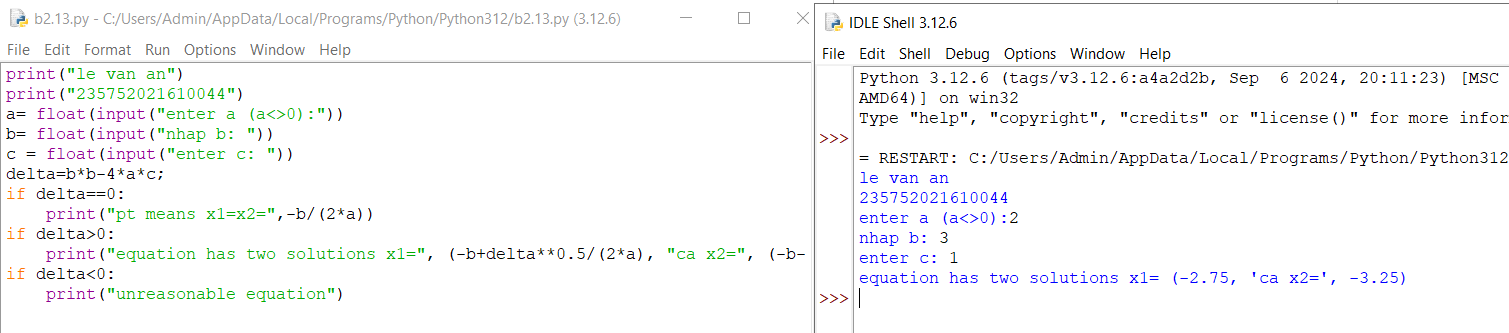
1. Viết chương trình kết nối các danh sách vào từ điển



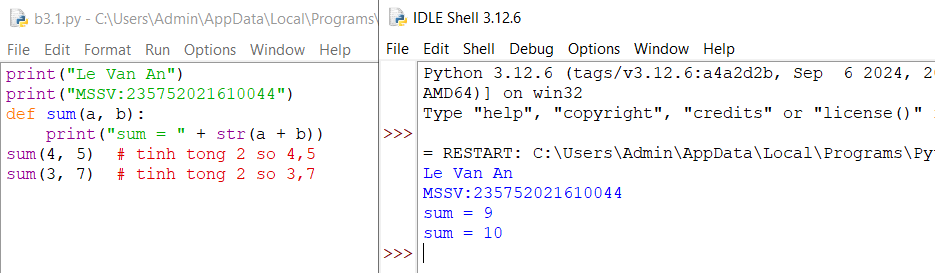
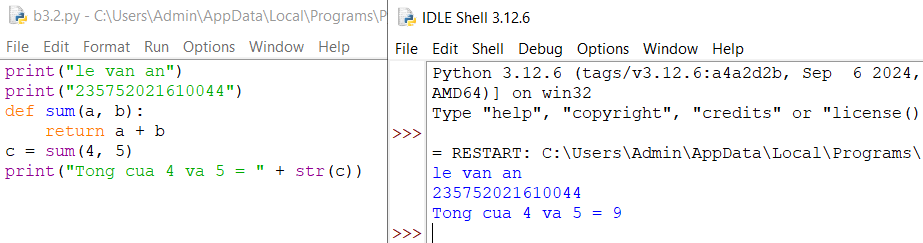
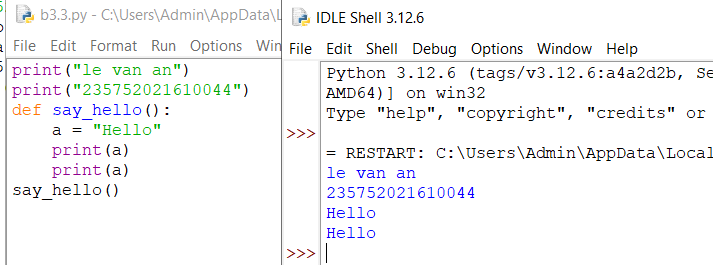
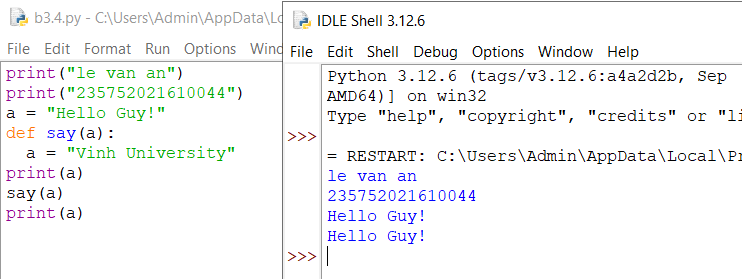
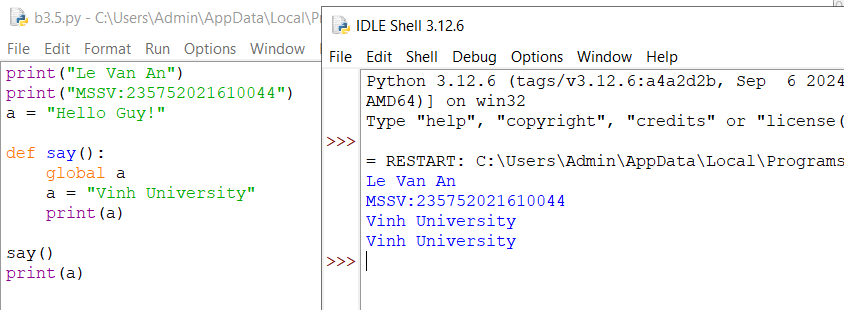
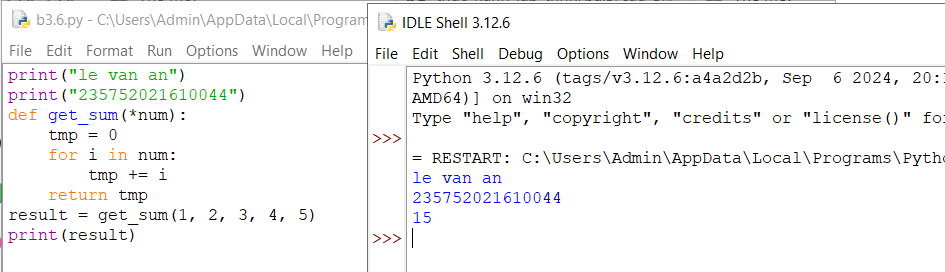
1. Một website yêu cầu người dùng nhập tên người dùng và mật khẩu để đăng ký.

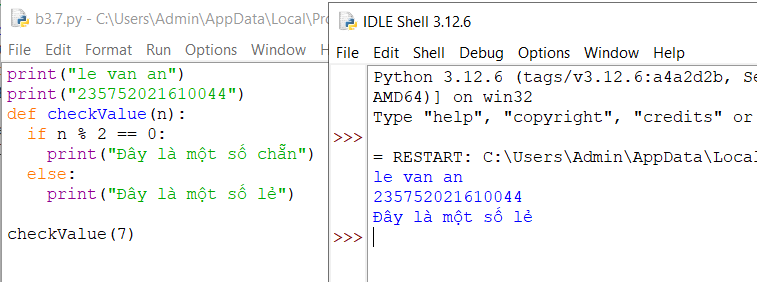
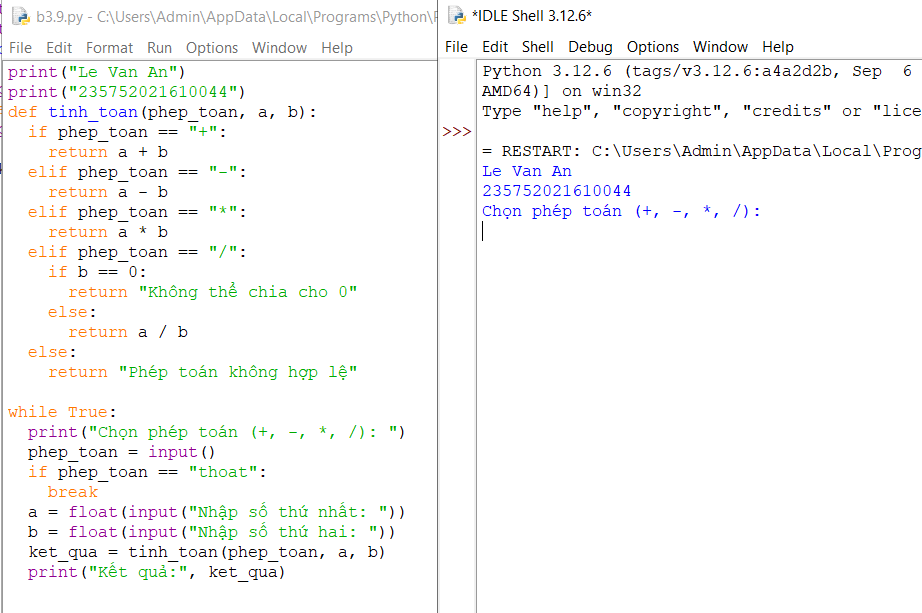
Viết chương trình để kiểm tra tính hợp lệ của mật khẩu mà người dùng nhập vào.

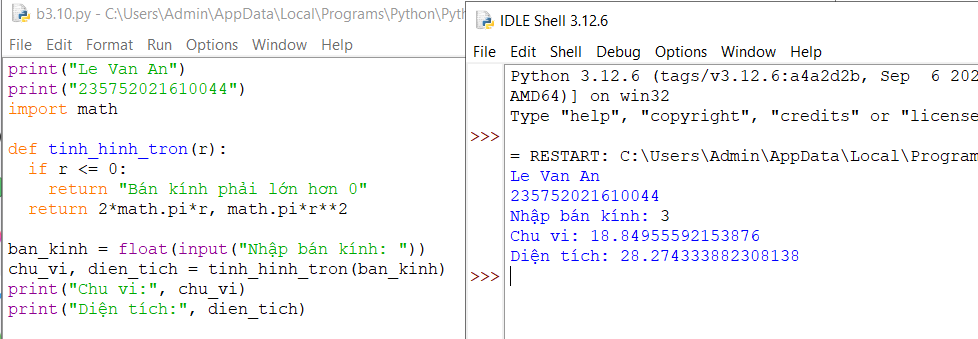
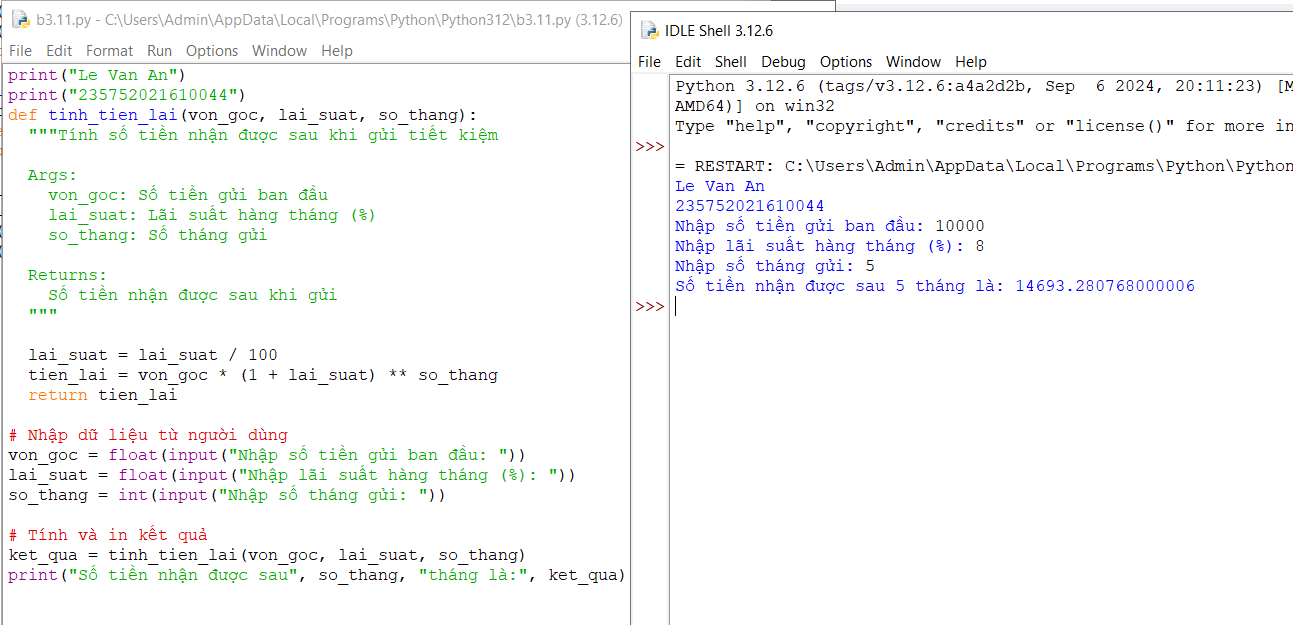


13. Viết 

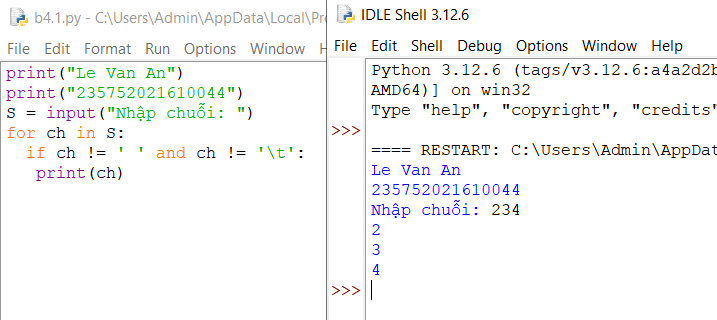
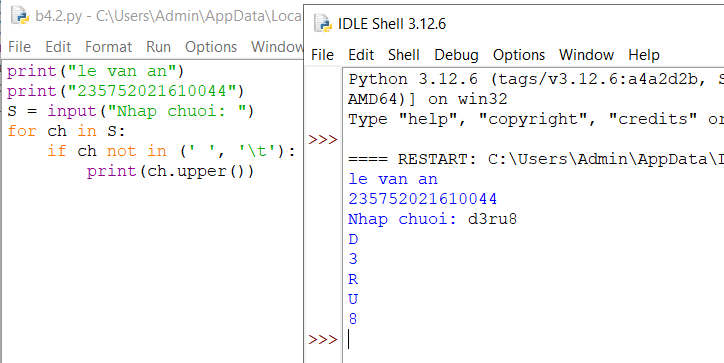
## Bài 3. Lập trình hàm trong Python

1. Viết hàm sum() tính tổng hai số 
2. Viết hàm sum() với kết quả trả về 
3. Tìm và sửa lỗi chương 
4. Viết chương trình có phạm vi biến như 
5. Viết chương trình sau và xem sự thay đổi của biến 
6. Viết chương trình sau và giải thích việc truyền tham số của hàm 
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.

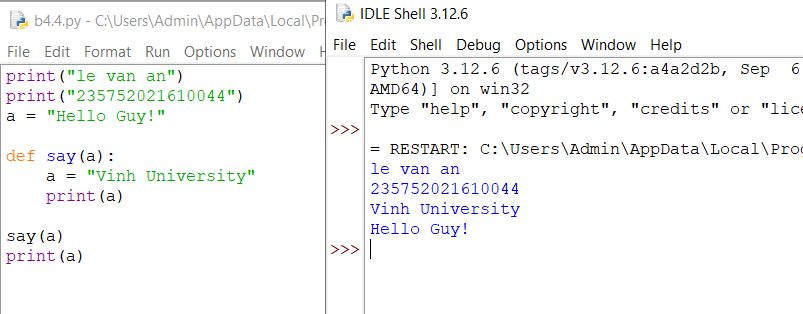
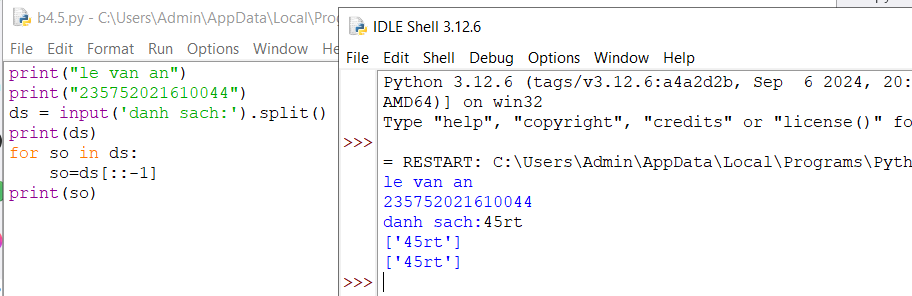
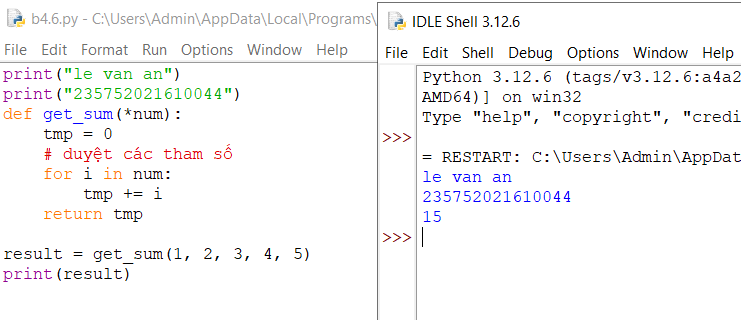
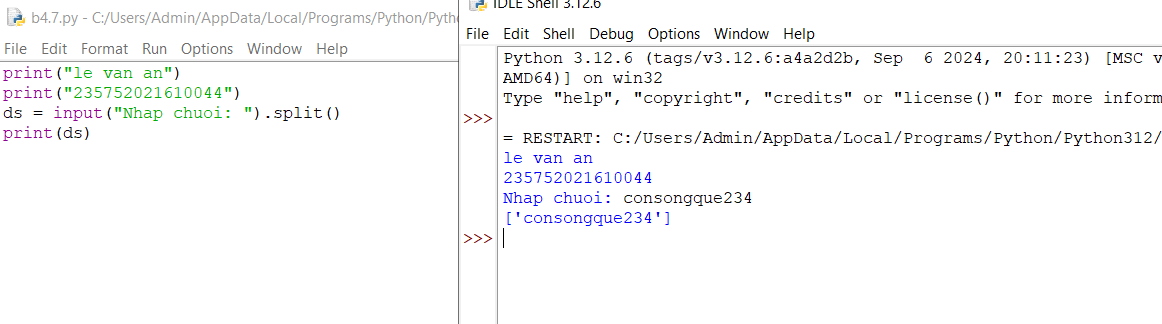
9. Chương trình máy tính thực hiện các phép tính đơn giản 

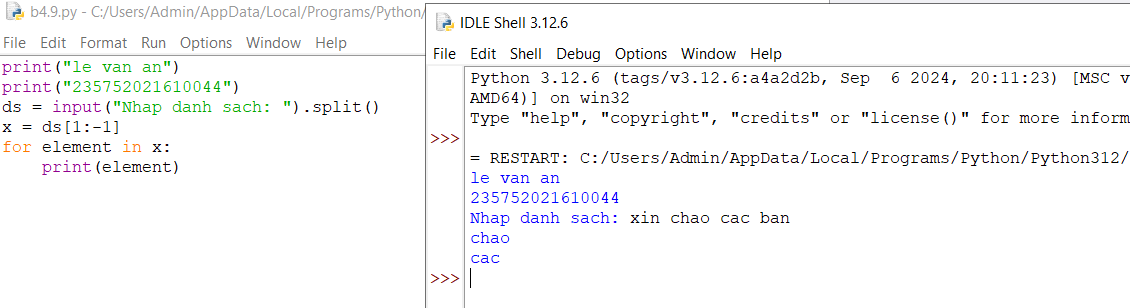
1. Viết hàm “def Tinh(R):” tính chu vi và diện tích hình tròn, với bán kính R được nhập từ bàn phím, và kiểm tra giá trị bán kính đầu vào là hợp lệ. 
2. Biết lãi suất tiết kiệm là t%/tháng (nhập t từ bàn phím). Nhập số vốn ban đầu n và số tháng gửi k. Tính số tiền nhận được sau k tháng sử dụng cấu trúc hàm def benefit(t,n,k): 

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

1. Chỉnh sửa ví dụ trên: hãy bỏ qua không in ra những kí tự “không nhìn thấy” (dấu space và dấu tab). 
2. Chỉnh sửa ví dụ ở bài 1: hãy các kí tự ở dạng IN HOA. 
3. 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



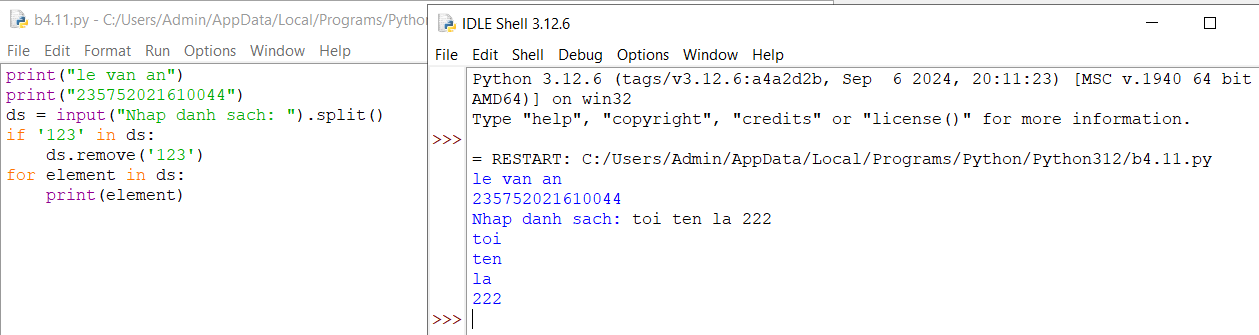
1. Chỉnh sửa ví dụ ở bài 4: nhập 1 danh sách các từ từ bàn phím, in ra các từ đó theo thứ tự ngược lại thứ tự vừa nhập (ví dụ nhập “mot hai ba” thì in ra “ba hai mot”) 
2. Nhập một tên người từ bàn phím, hãy tách phần họ và tên riêng của người đó và in chúng ra màn hình (giả thiết họ và tên riêng chỉ gồm một âm). 
3. 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. 
4. 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.
5. Nhập một list từ bàn phím
6. Cắt list: lấy list nhưng bỏ phần tử đầu và cuối



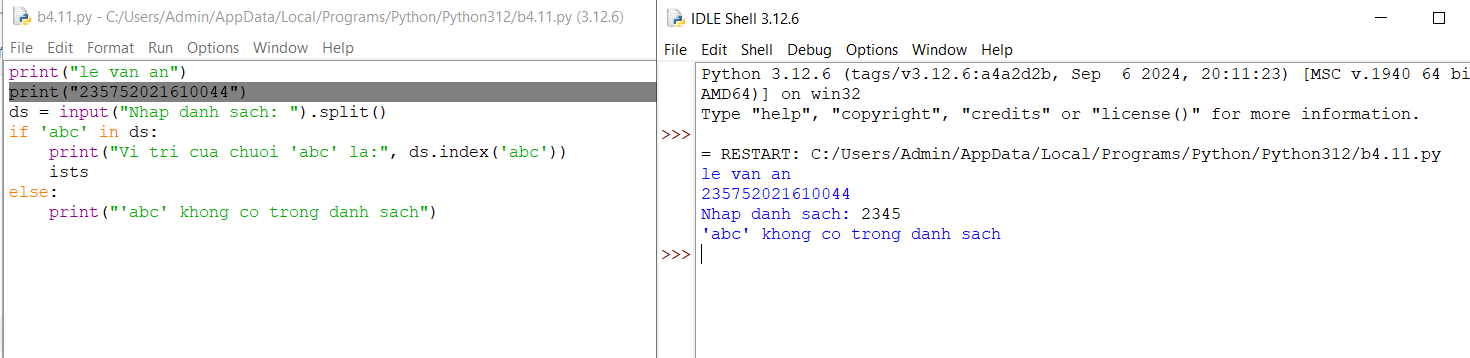
1. Thêm phần tử vào list



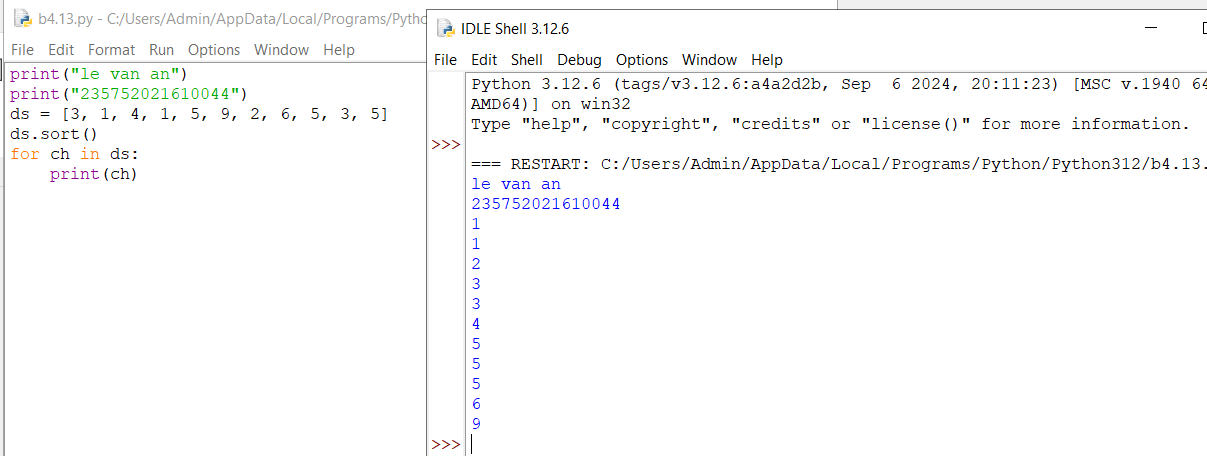
1. Bỏ phần tử khỏi list

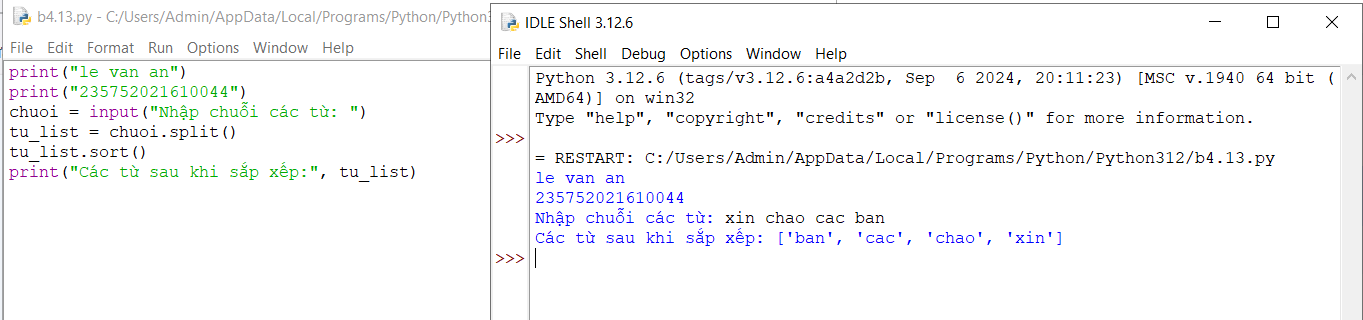
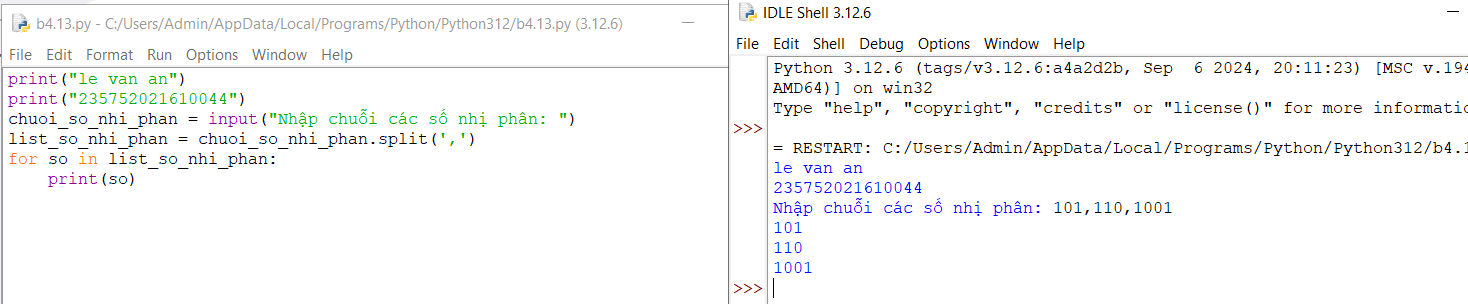
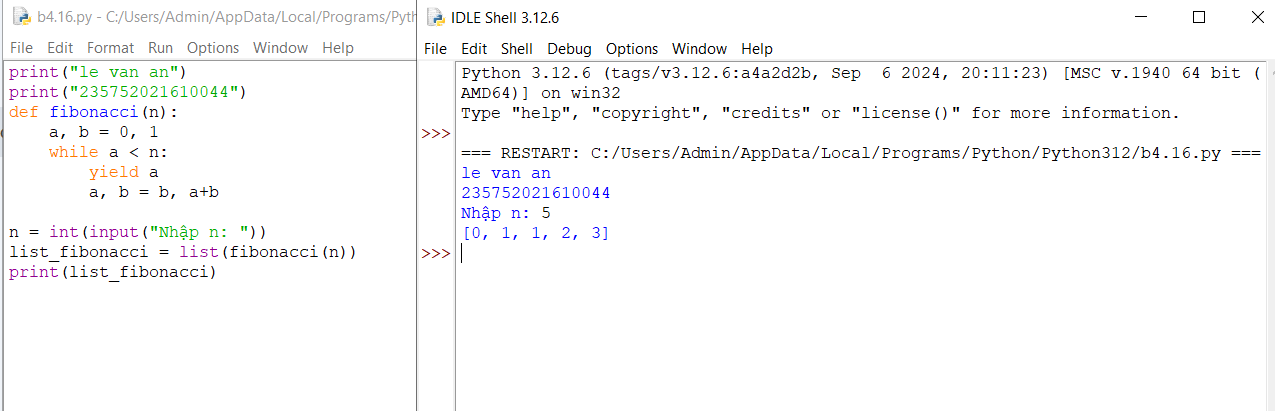
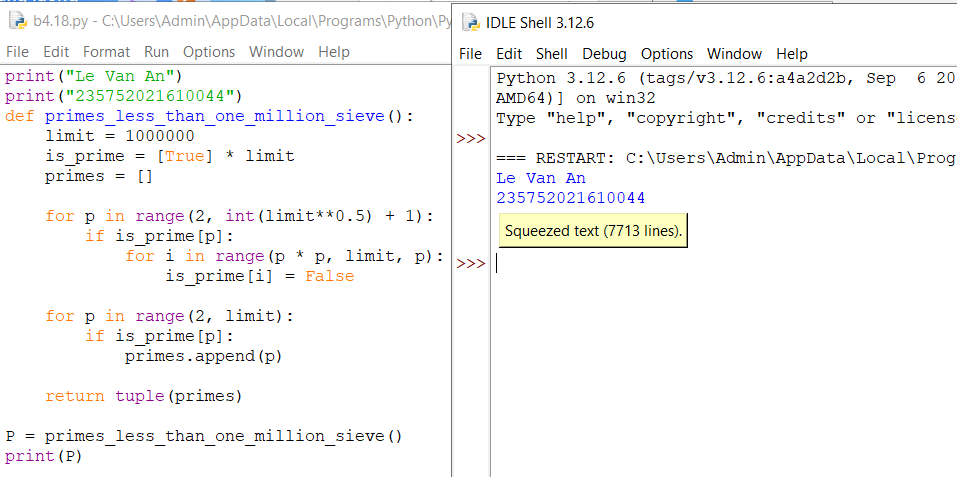
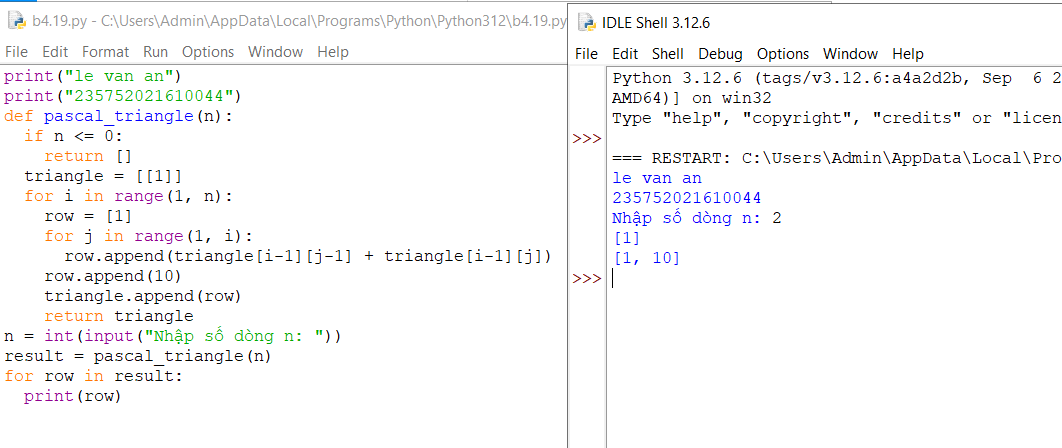
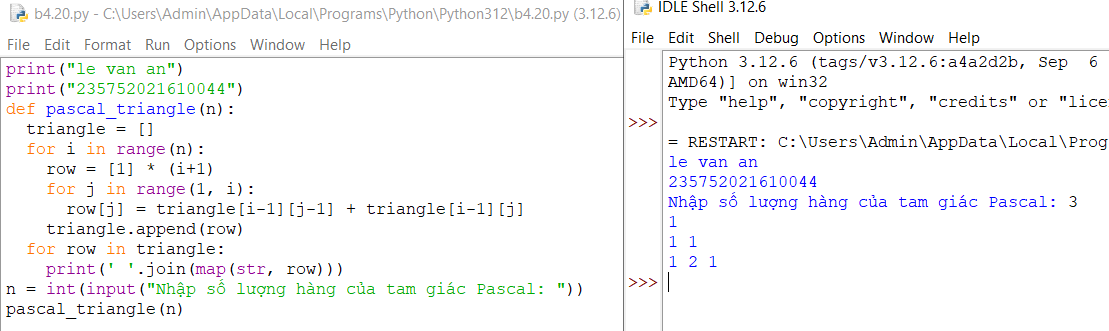
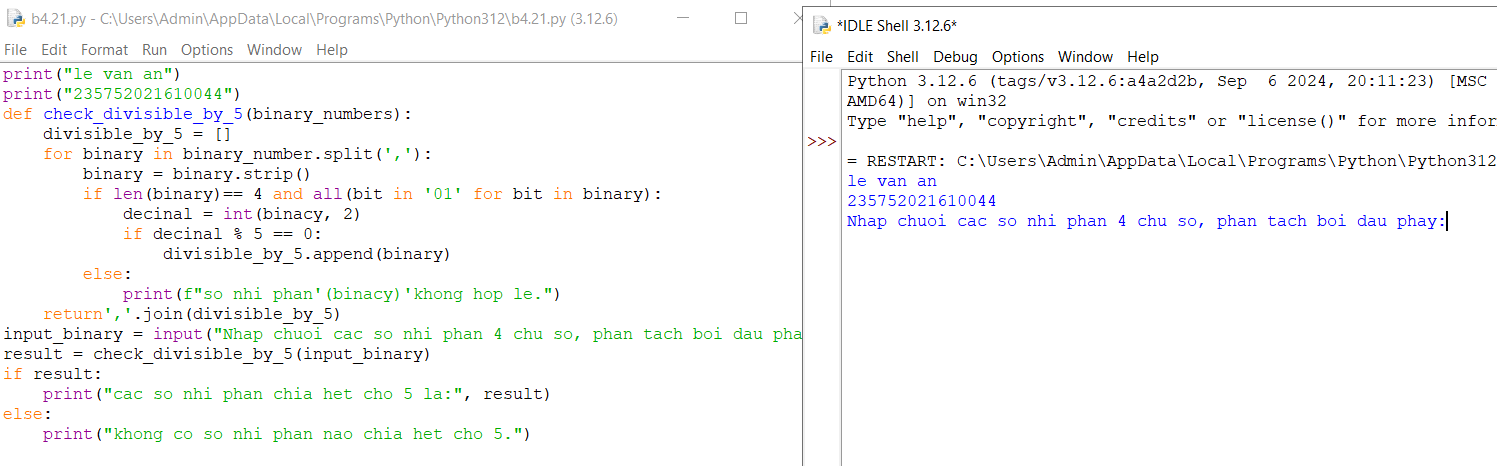
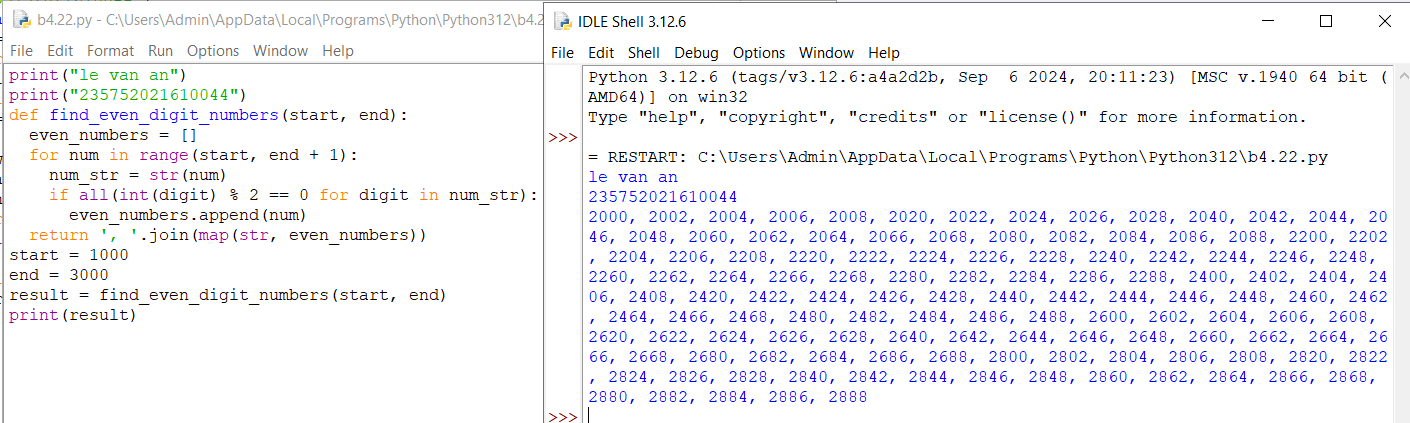


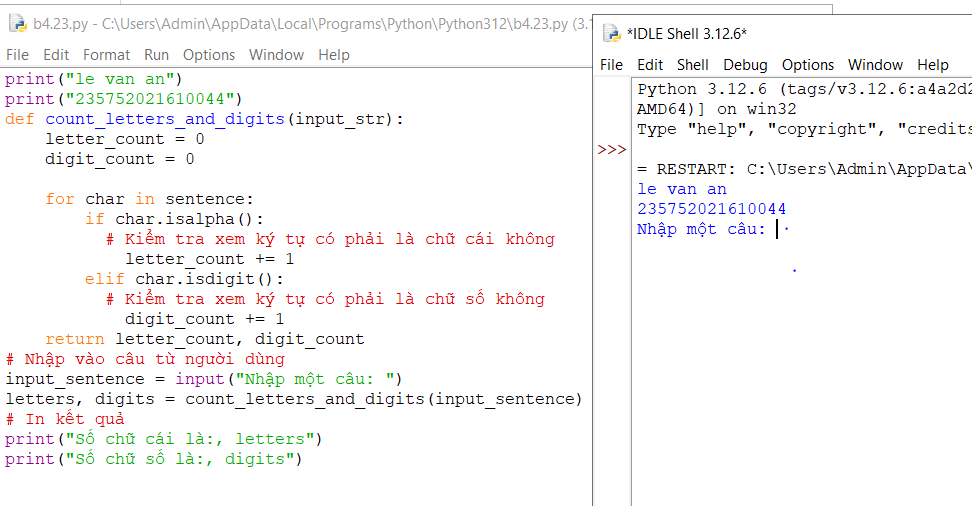
1. Tìm kiếm phần tử trong list

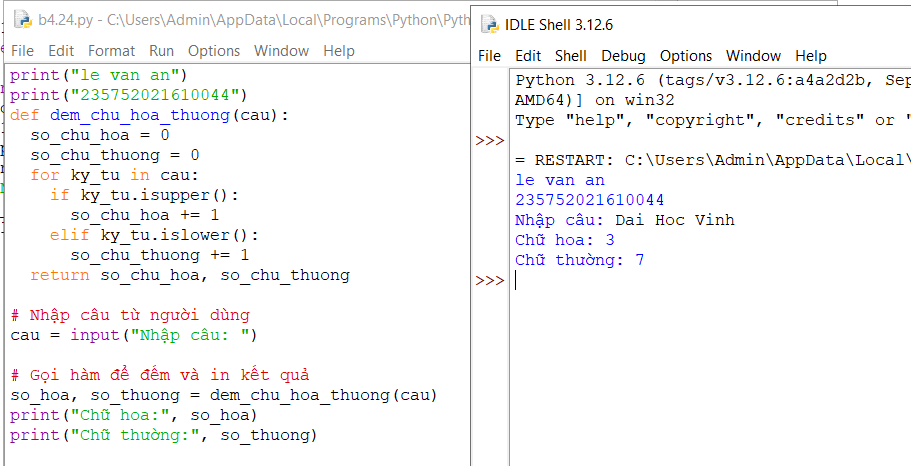


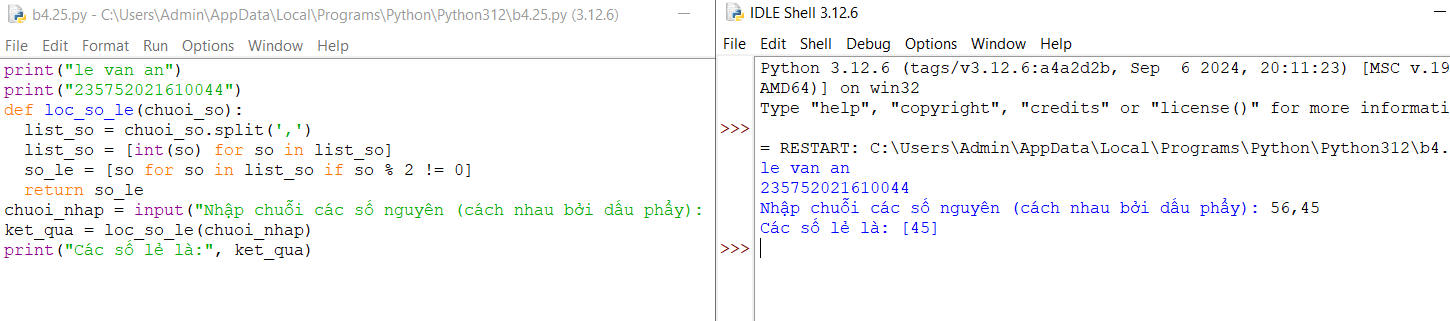
1. Sắp xếp các phần tử trong list



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 theo thứ tự từ điển. 
2. 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. 
3. Nhập số n, in ra màn hình các số nguyên dương nhỏ hơn n có tổng các ước số lớn hơn chính nó. 
4. Hãy nhập số nguyên n, tạo một list gồm các số fibonacci nhỏ hơn n và in ra màn hình. 
5. Hãy tạo ra tuple P gồm các số nguyên tố nhỏ hơn 1 triệu. 
6. Nhập n, in n dòng đầu tiên của tam giác pascal. 
7. 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. 
8. 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. 
9. 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: 
10. Viết một chương trình chấp nhận đầu vào là một câu, đếm chữ hoa, chữ thường.



1. Sử dụng một danh sách để lọc các số lẻ từ danh sách được người dùng nhập vào.
2. 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.



26.

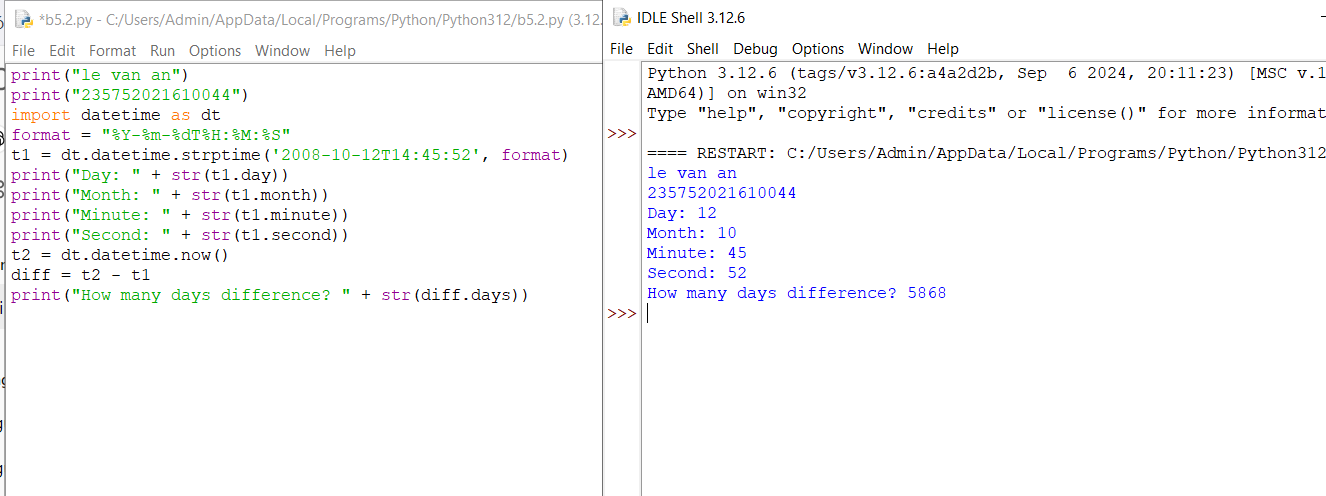


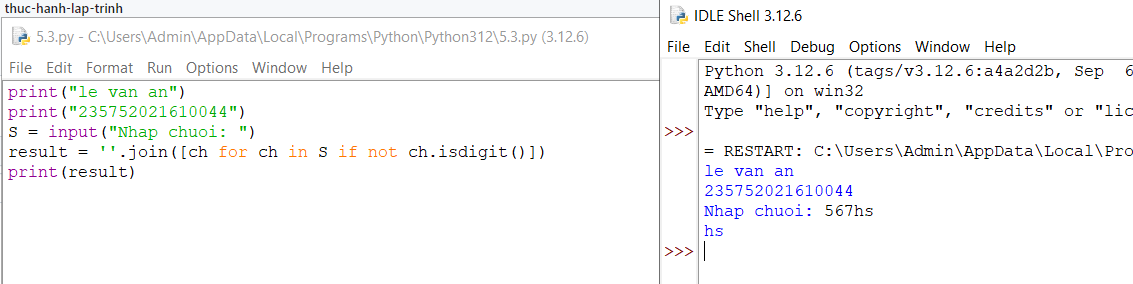
## Bài 5. Thiết kế module trong Python

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

Sử dụng các hàm bằng cách sử dụng *“import”* gọi hàm theo cấu trúc

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

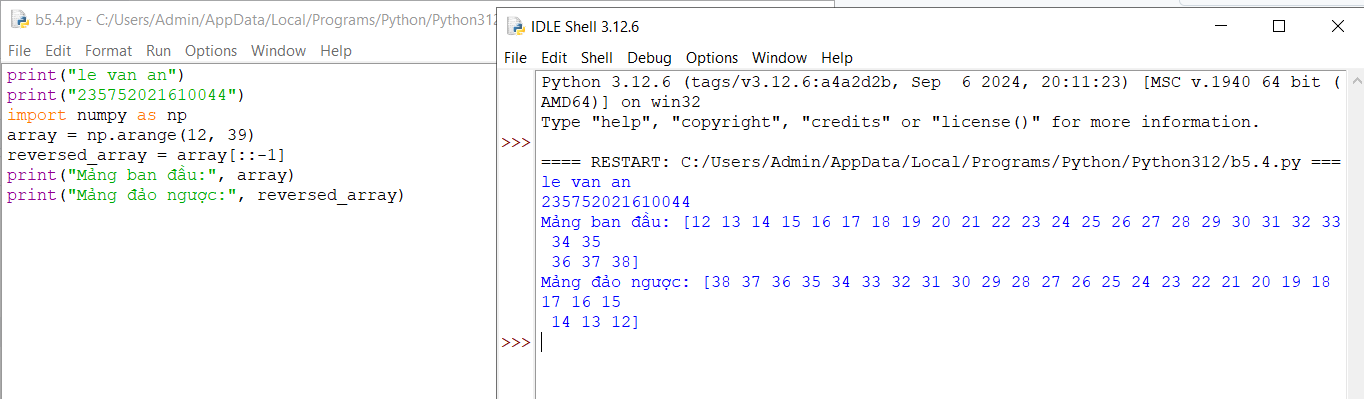
1. Viết chương trình sử dụng thư viện NumPy để tạo một mảng với các giá trị nằm trong khoảng từ 12 đến 38 Đầu ra:
   1. 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37]



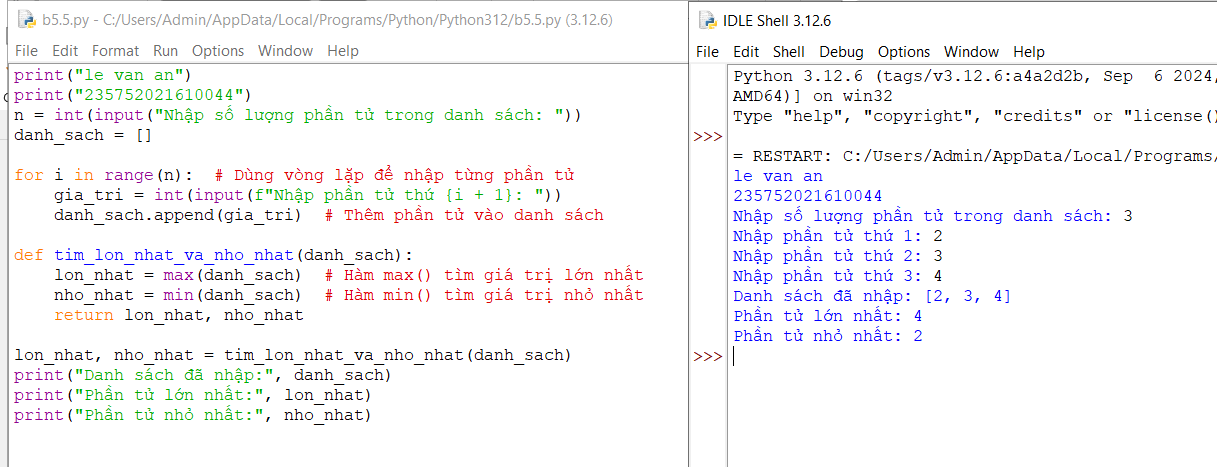
1. Viết chương trình để tạo một mảng với các giá trị nằm trong khoảng từ 12 đến 38 và đảo ngược mảng đã tạo (phần tử đầu tiên trở thành cuối cùng) Mảng được tạo:
   1. 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37]

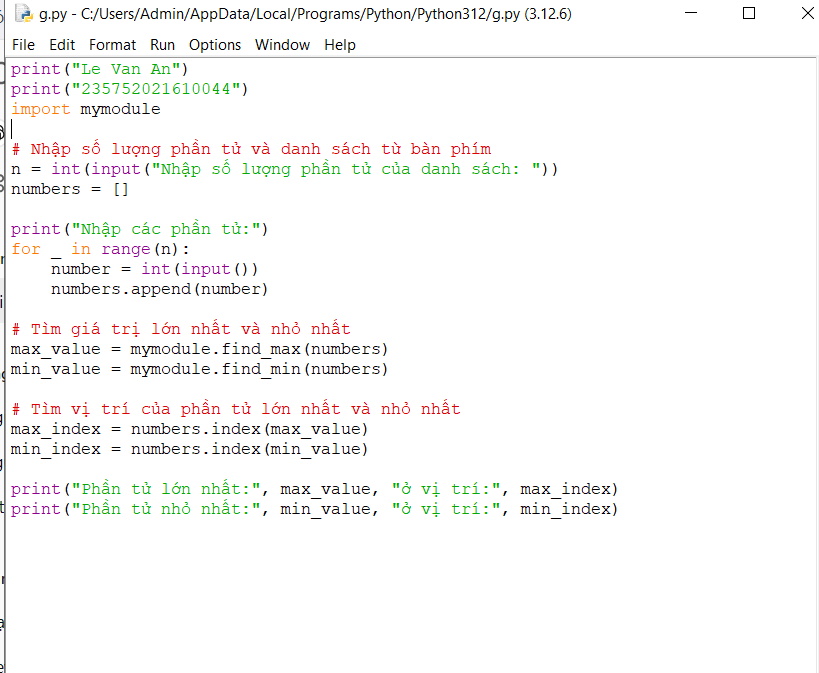
Mảng đảo ngược:

[37 36 35 34 33 32 31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12] Gợi ý:

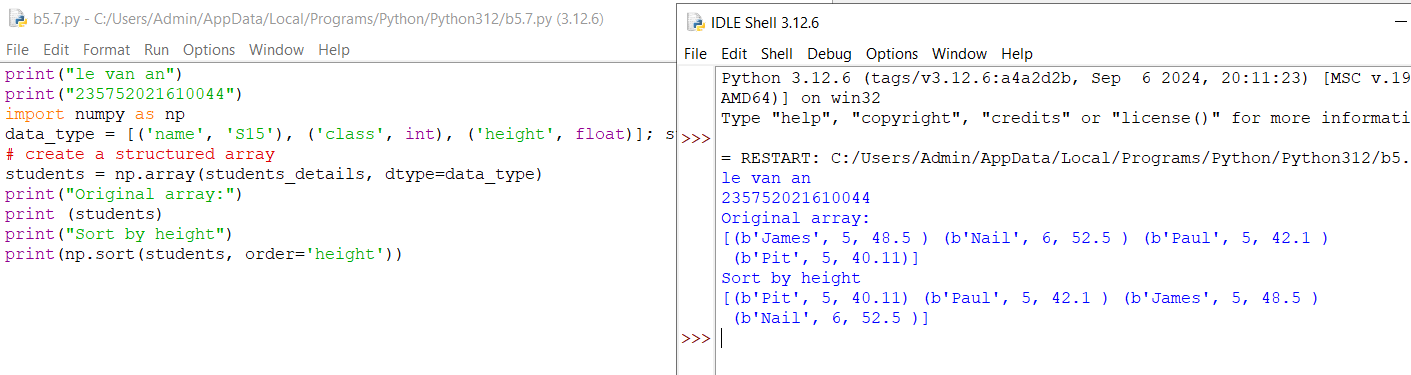


1. Viết chương trình tìm phần tử lớn nhất và nhỏ nhất của một danh sách
   * + Số lượng và giá trị của list được nhập từ bàn phím
     + Phương thức sắp xếp và tìm phần tử lớn nhất được viết thành module

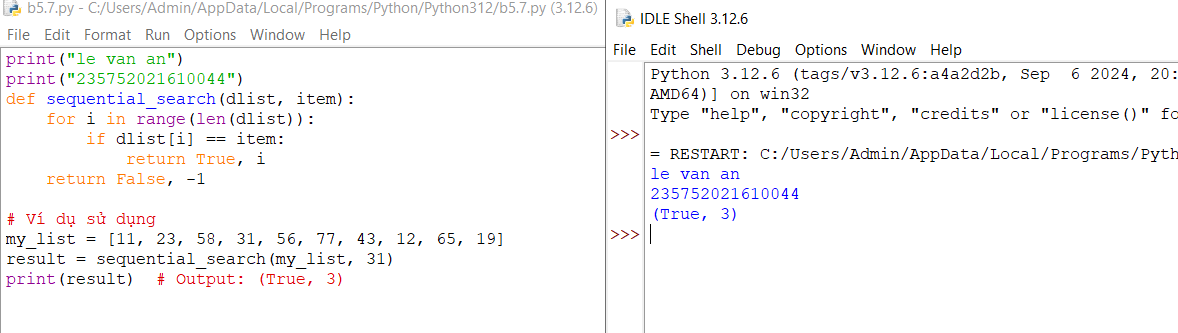


1. In ra vị trí phần tử lớn nhất và nhỏ nhất tìm được ở bài tập trên 

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.

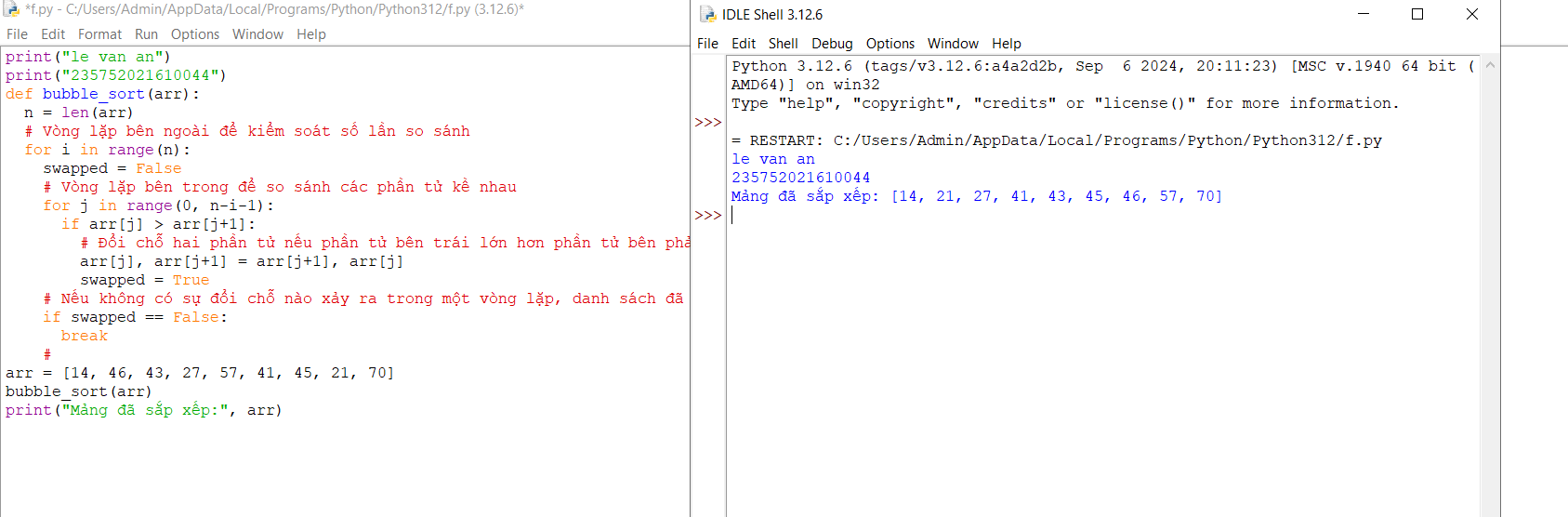
Sequential\_Search([11,23,58,31,56,77,43,12,65,19],31) -> (True, 3) Gợi 9. Xây dựng hàm *“binary\_search(list, value)”* (giải thuật tìm kiếm nhị phân) dưới dạng module. Viết chương trình nhập một list n phần tử từ bàn phím và tìm kiếm phần tử *value* bất kỳ.



10. 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]

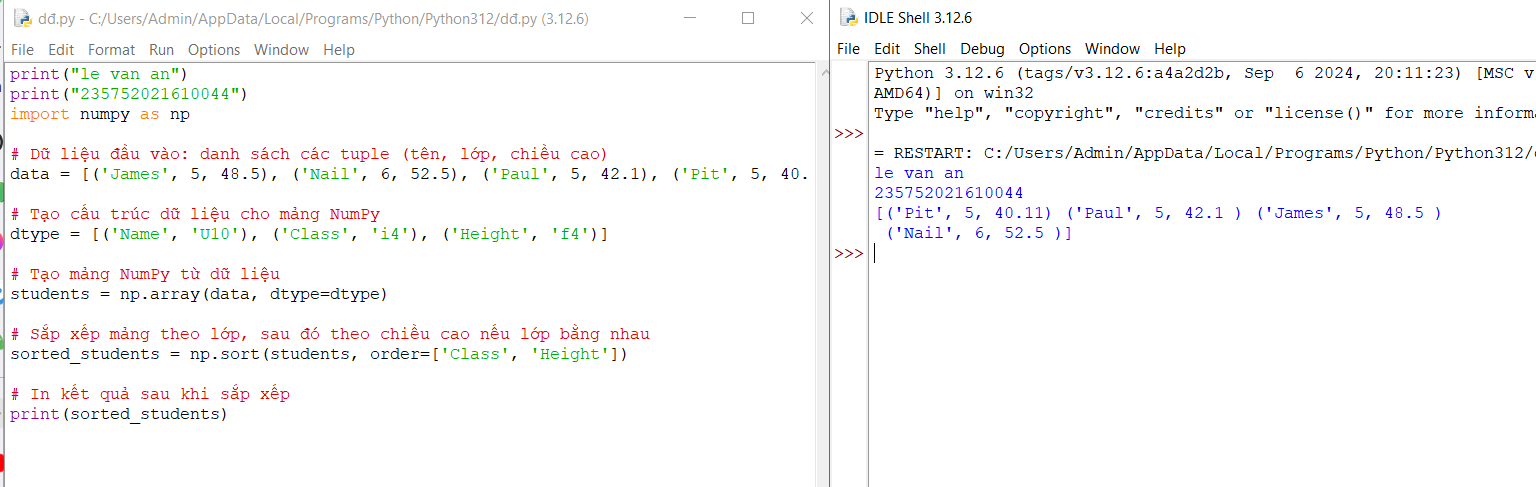
Expected Result: [14, 21, 27, 41, 43, 45, 46, 57, 70] Gợi 

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.

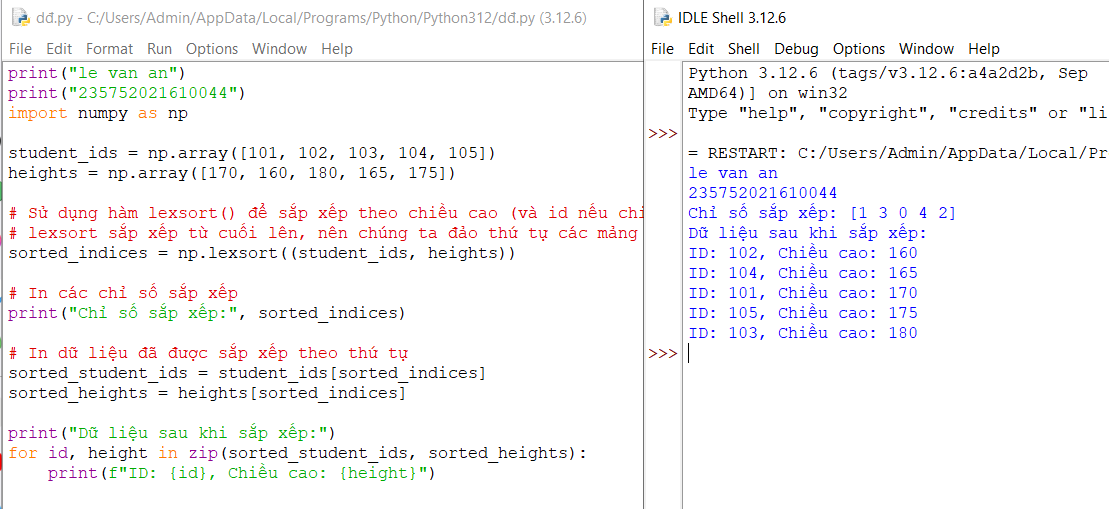
Dữ liệu đầu vào:

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

Kết quả sắp xếp:

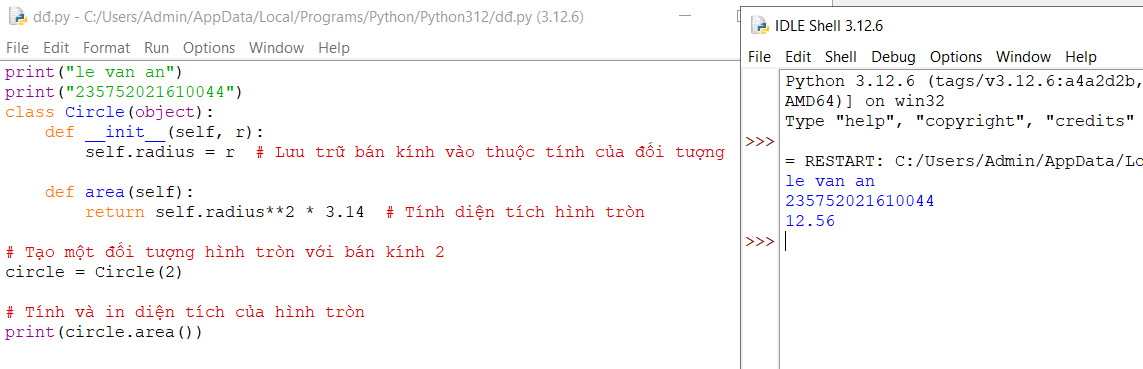
[('Pit', 5, 40.11) ('Paul', 5, 42.1 ) ('James', 5, 48.5 ) ('Nail', 6, 52.5 )] 

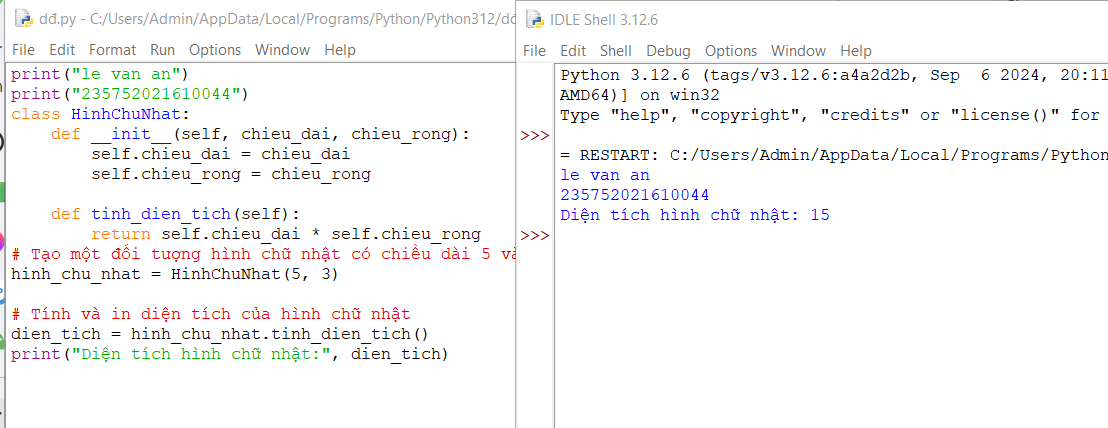
1. Viết chương trình sử dụng thư viện NumPy để sắp xếp id sinh viên với chiều cao tăng dần của sinh viên từ id sinh viên và chiều cao đã cho. In các chỉ số nguyên mô tả thứ tự sắp xếp theo nhiều cột và dữ liệu được sắp xếp (sử dụng hàm *lexsort()*)



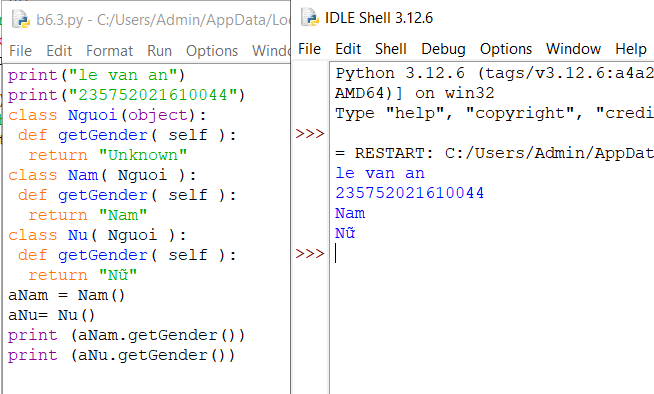
## Bài 6: Lập trình hướng đối tượng trong Python

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.

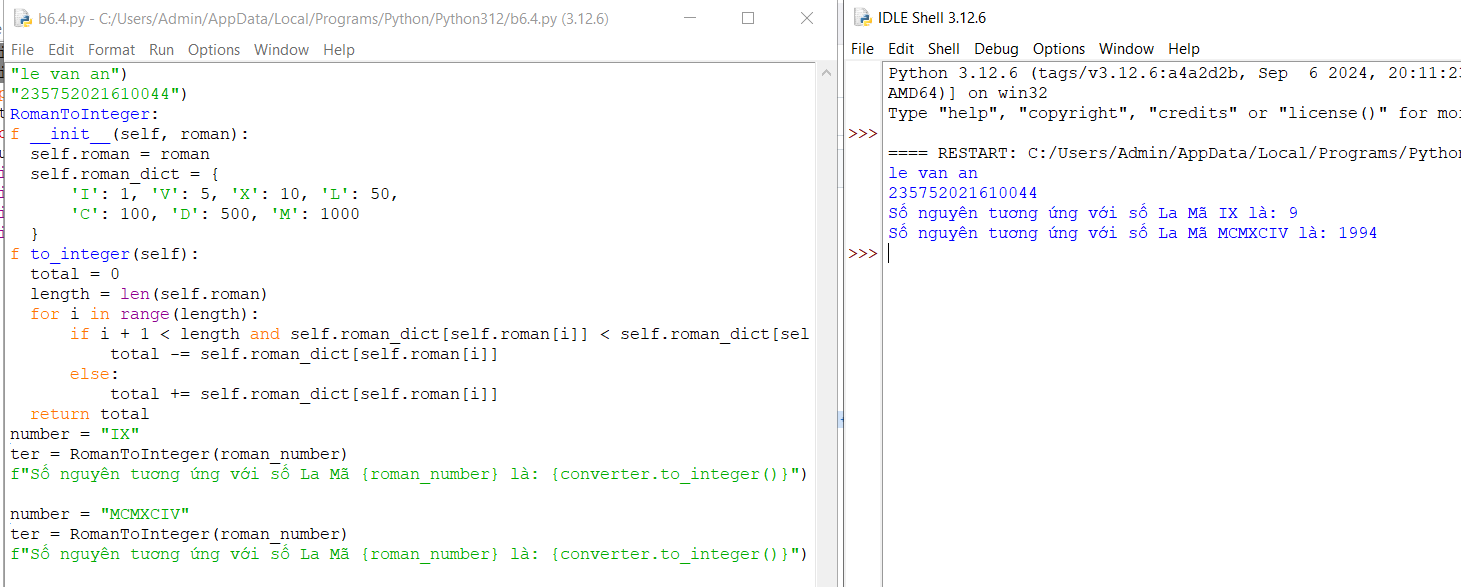


1. Định nghĩa class có tên là *Hinhchunhat* được xây dựng bằng chiều dài và chiều rộng. Class Hinhchunhat có method để tính diện tích. 

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*.

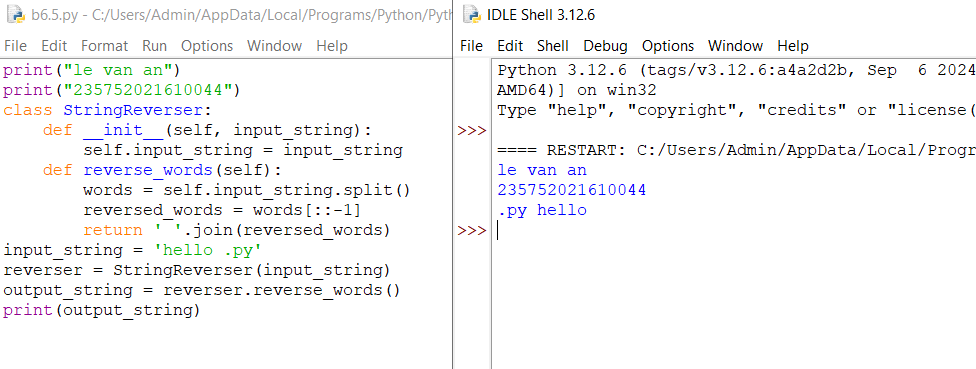


1. Viết chương trình Python dưới dạng class để chuyển đổi một số La Mã thành một số nguyên. Flowchart:

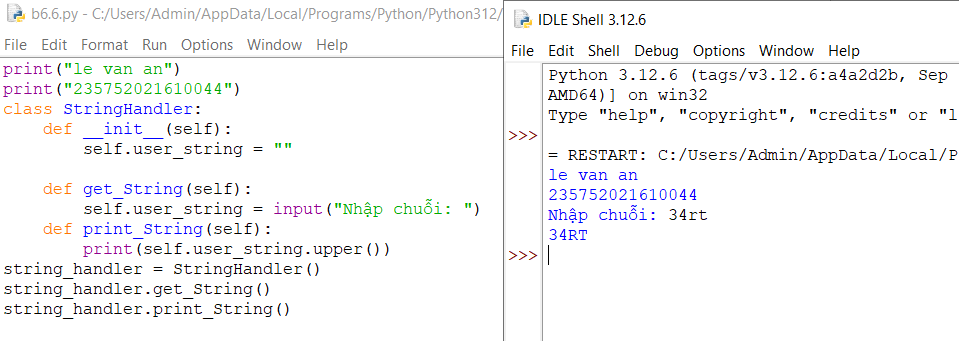


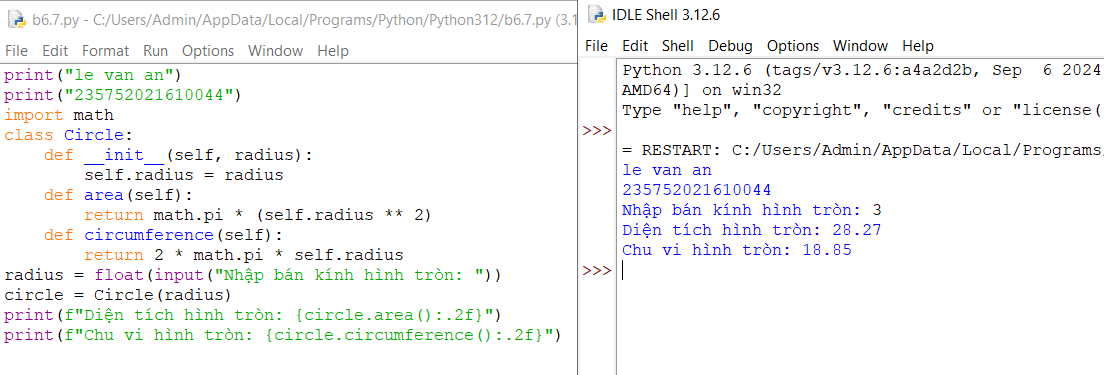
1. Viết chương trình Python dưới dạng class để đảo ngược chuỗi từ từng chữ. Dữ liệu vào : 'hello .py'

Đầu ra : '.py hello' Flowchart:

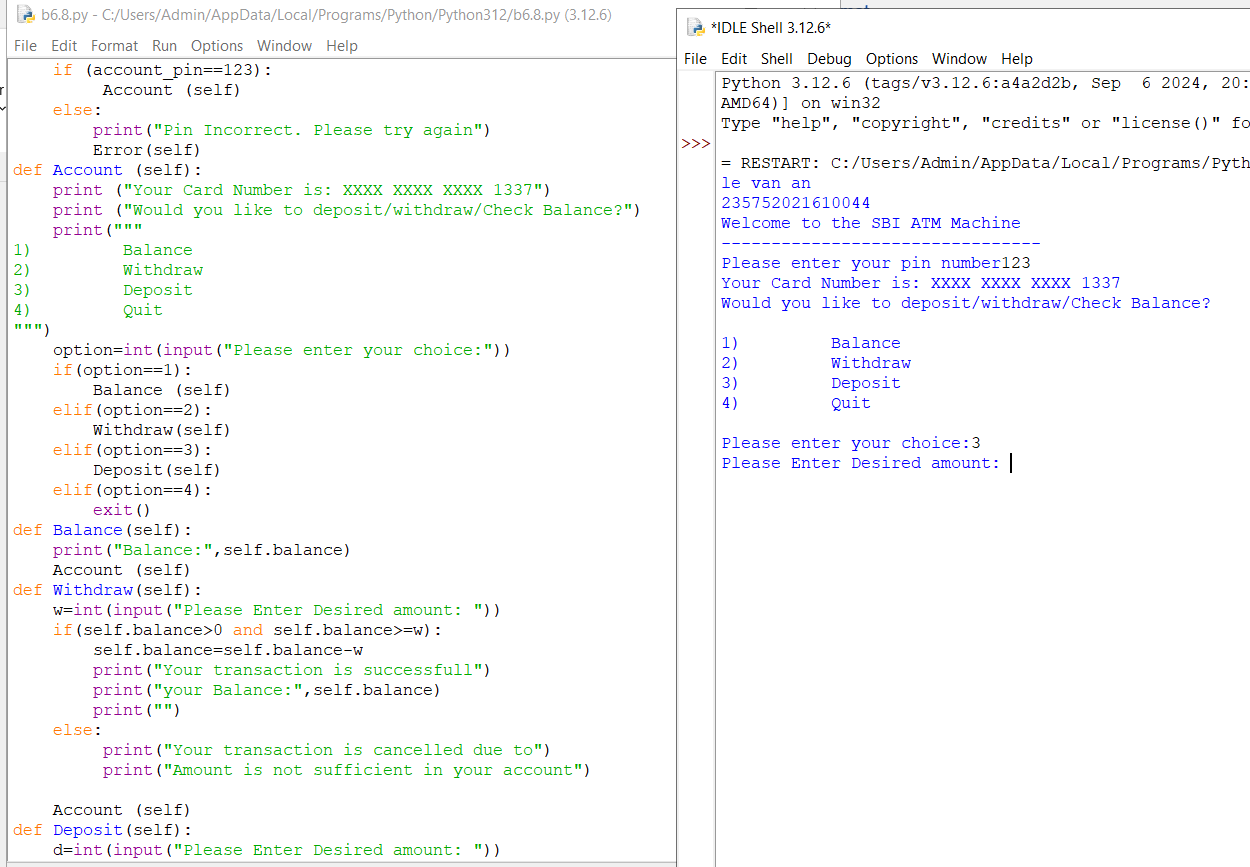


1. Viết một class Python có hai phương thức get\_String và print\_String. get\_String chấp nhận một chuỗi từ người dùng và print\_String in chuỗi đó bằng chữ in hoa.



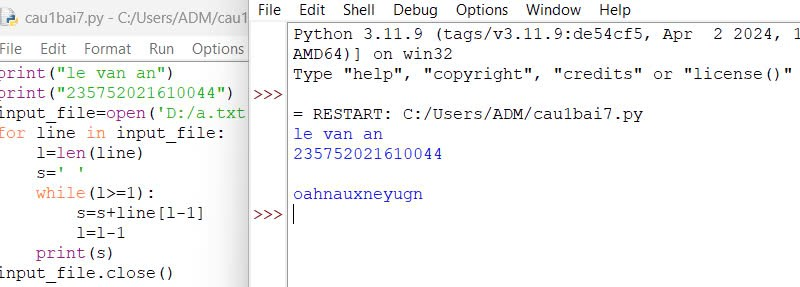
1. Viết một class Python có tên Circle được xây dựng theo bán kính và hai phương thức sẽ tính diện tích và chu vi của hình tròn. 

1. Chương trình ATM đơn giản

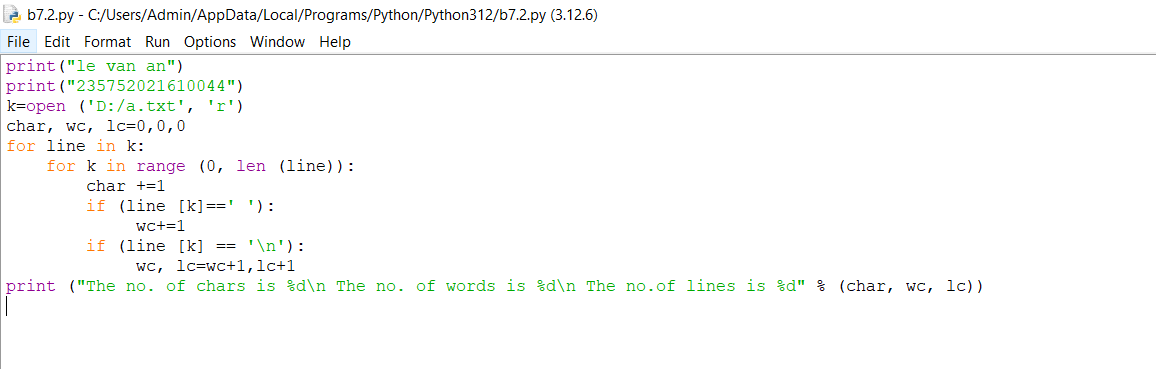


## Bài 7: Thao tác trên tập tin và thư mục trong Python

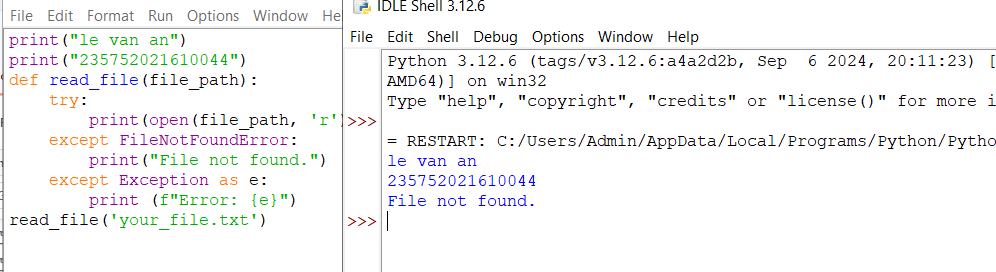
1. Chương trình đọc file và in đảo ngược kết quả



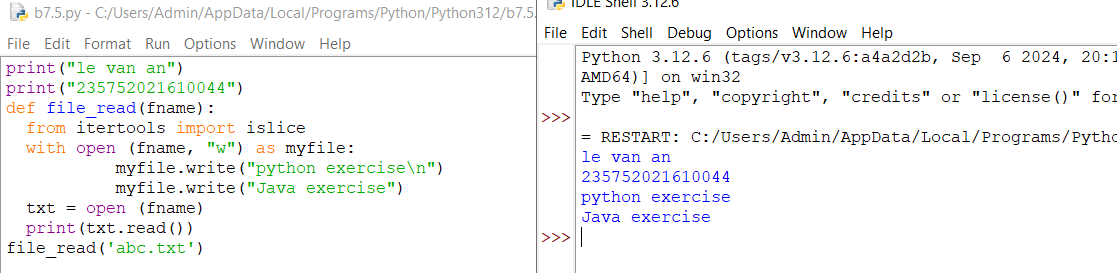
1. Chương trình đọc một file, tính số ký tự, số từ và số dòng của file

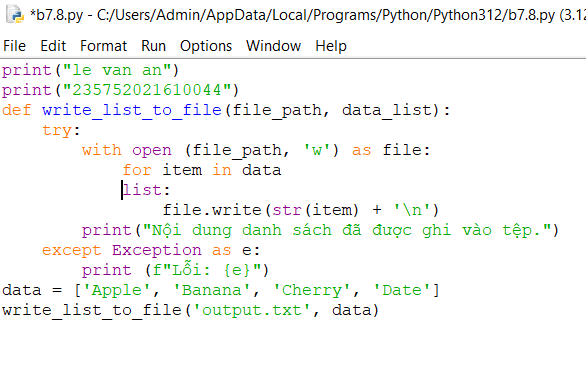
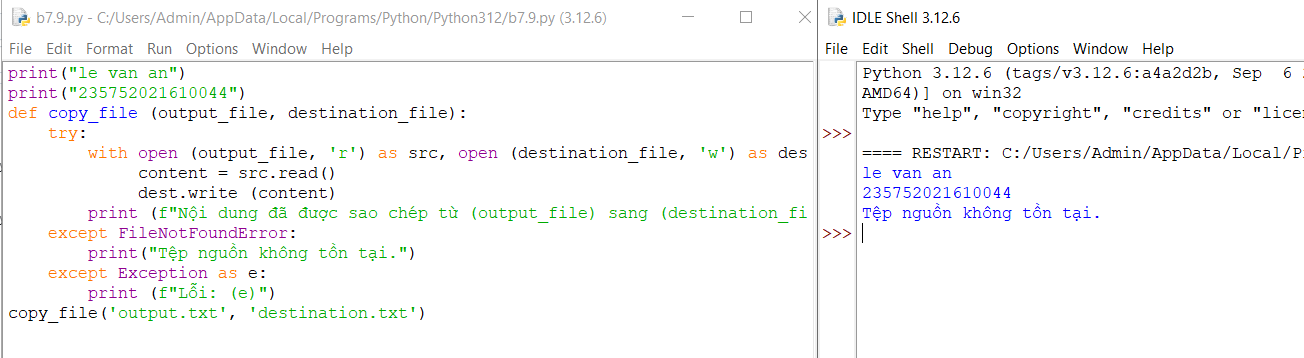


1. Viết chương trình Python để đọc toàn bộ tệp văn bản



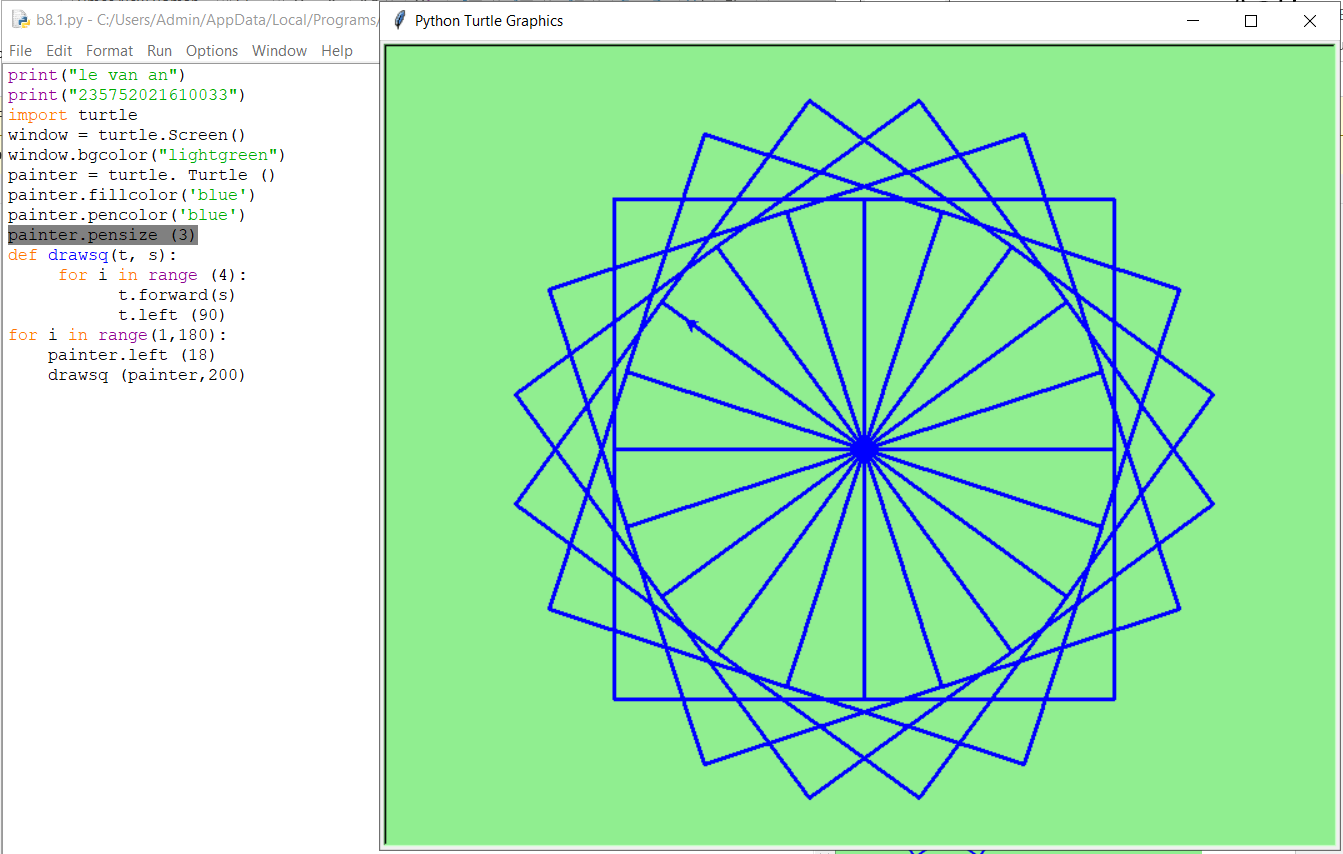
1. Chương trình Python để đọc n dòng đầu tiên của tệp 
2. Chương trình Python để nối văn bản vào tệp và hiển thị văn bản.



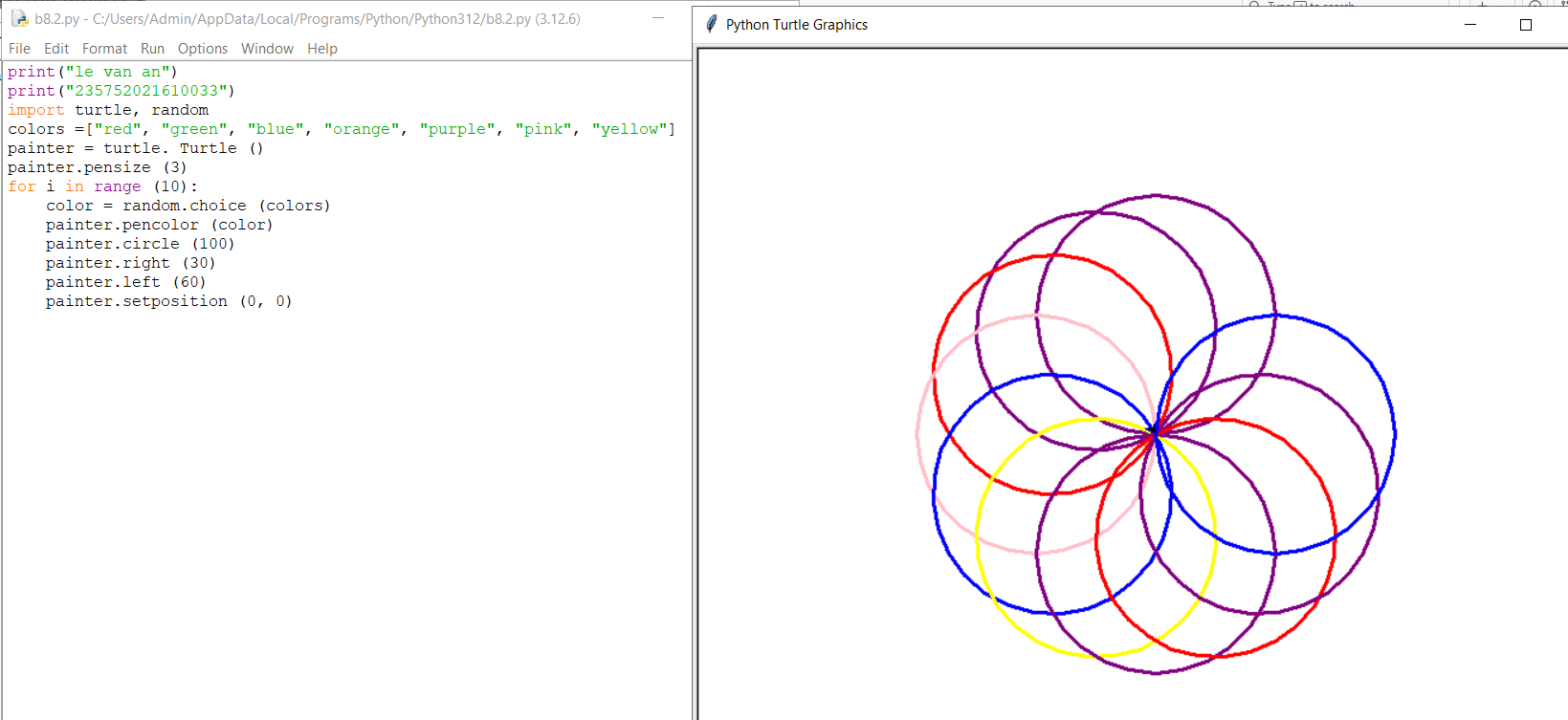
1. Viết chương trình Python để đếm số dòng trong tệp văn bản 
2. Viết chương trình Python để viết nội dung danh sách vào tệp. 
3. Viết chương trình Python để sao chép nội dung của tệp này sang tệp khác. 
4. Viết chương trình python để tìm những từ dài nhất trong văn bản

## Bài 8: Lập trình giao diện trong Python

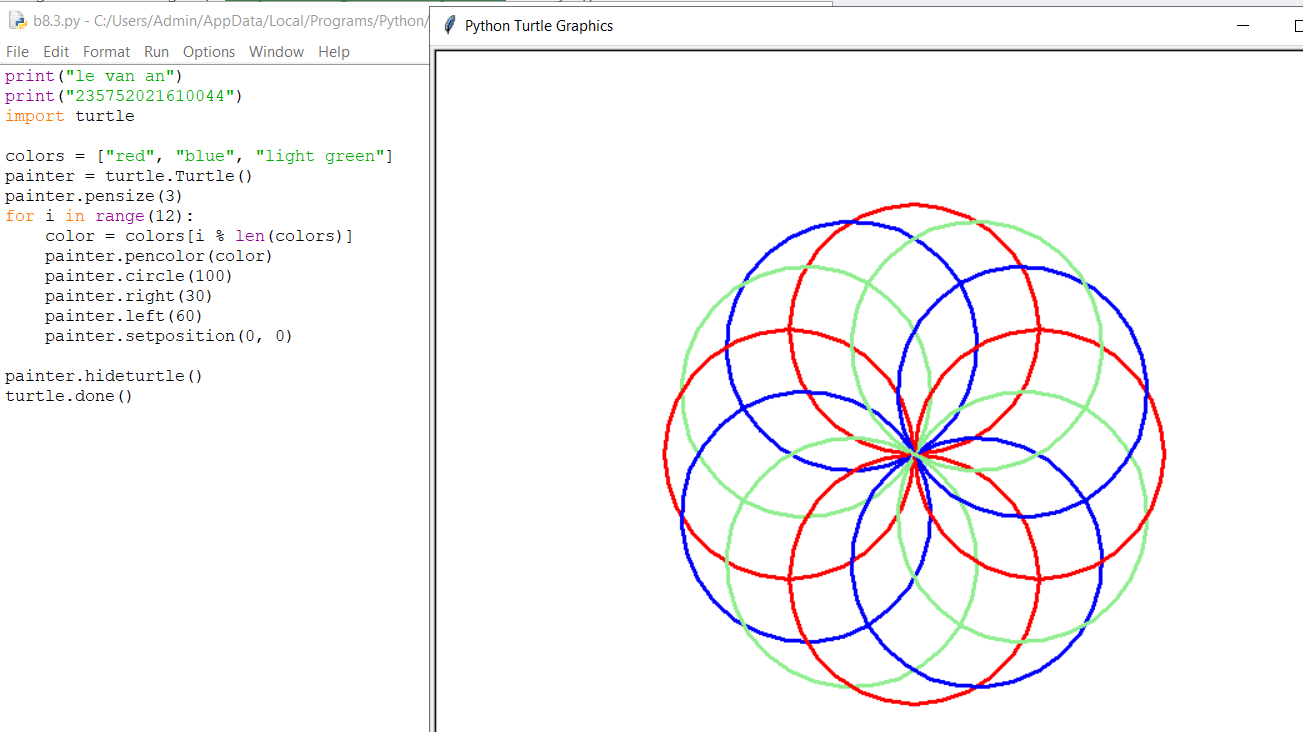
1. Viết chương trình đồ họa sử dụng thư viện turtle, kiểm tra kết quả và giải thích chương trình

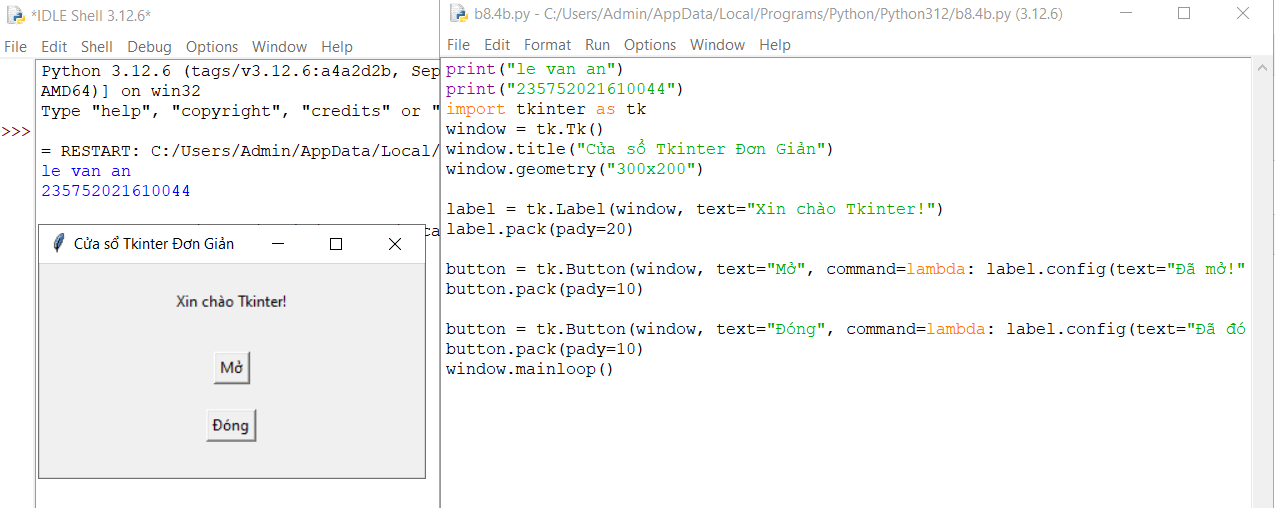
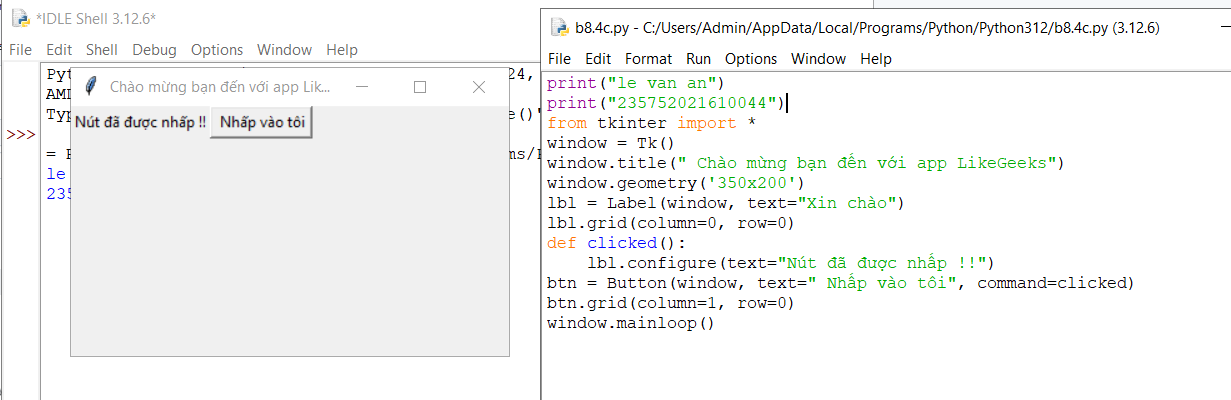


1. Viết chương trình đồ họa sử dụng thư viện turtle, kiểm tra kết quả và giải thích chương trình



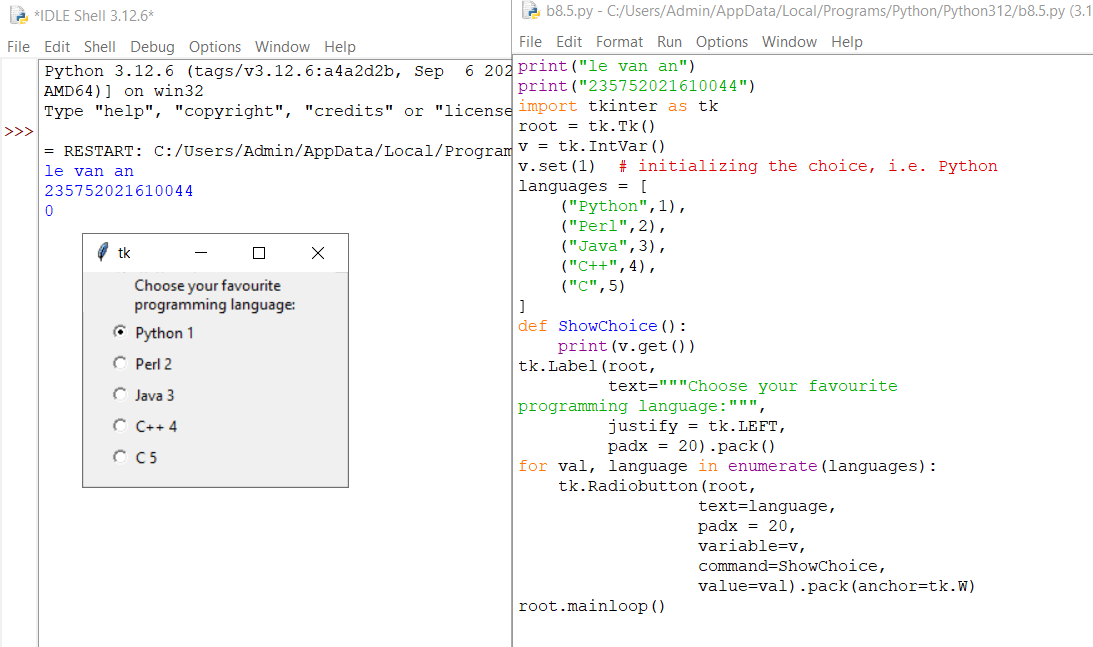
1. Dựa trên các kết quả đạt được từ các chương trình trên hãy viết chương trình hiển thị hình ảnh đồ họa sau

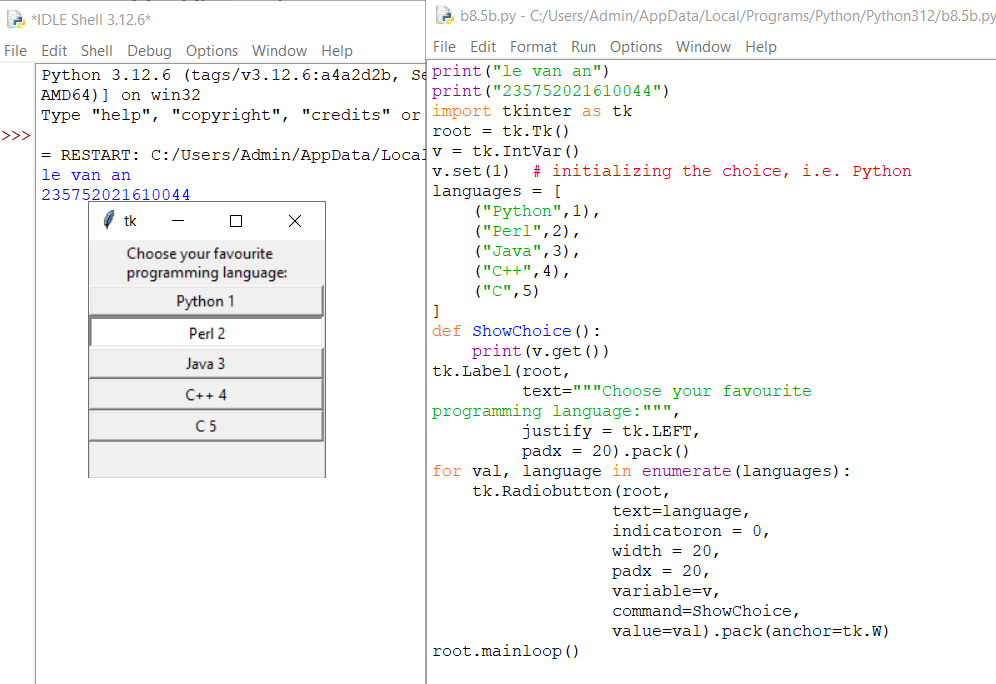


1. Viết chương sử dụng thư viện đồ họa tkinter thực hiện:
2. Xây dựng cửa sổ đồ họa window form 
3. Thêm một widget (button) vào window form 
4. Xây dựng phương thức xử lý sự kiện phím bấm Gợi ý: 

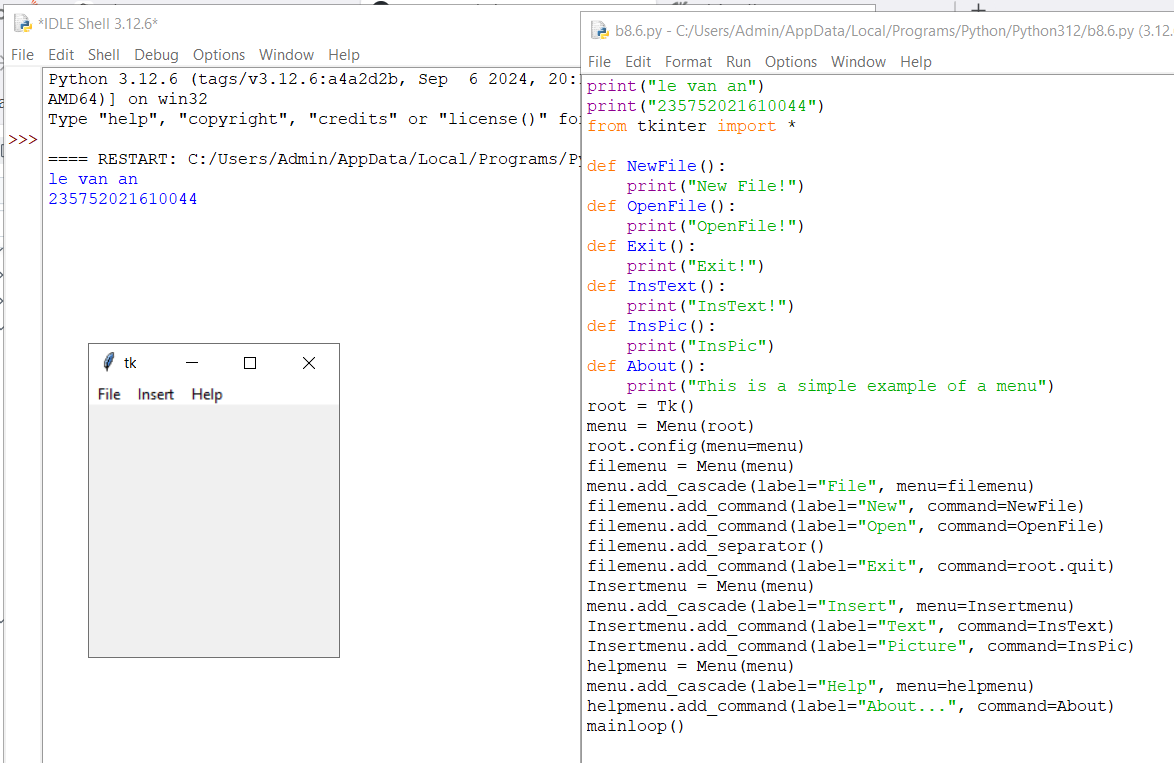
### d) Thay đổi màu nền và màu chữ của button sử dụng thuộc tính “bg” và “fg”

5. Sử dụng thư viện tkinter thực hiện:

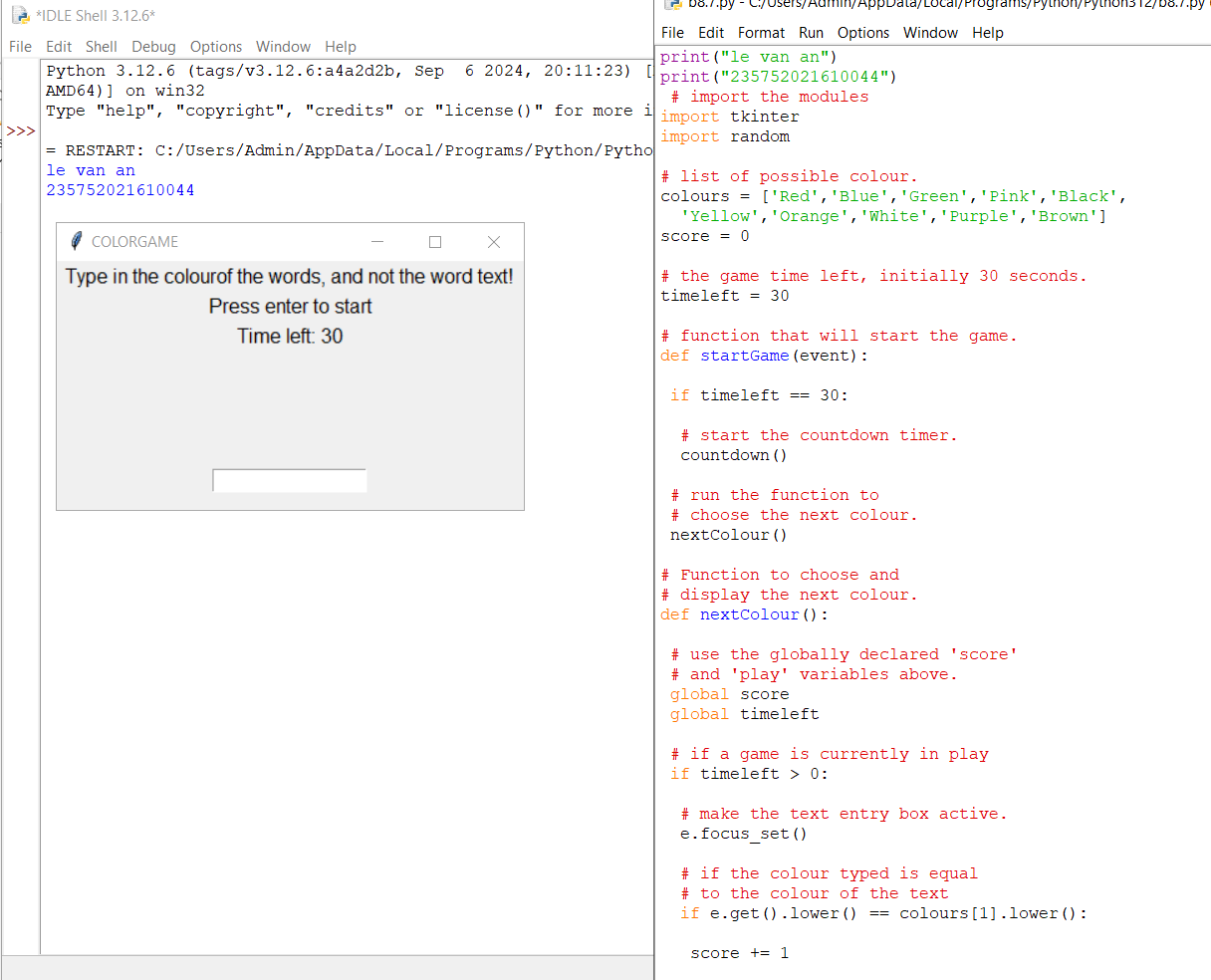
a) Xây dựng các radio button cho phép thực hiện các lựa chọn khác nhau 

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

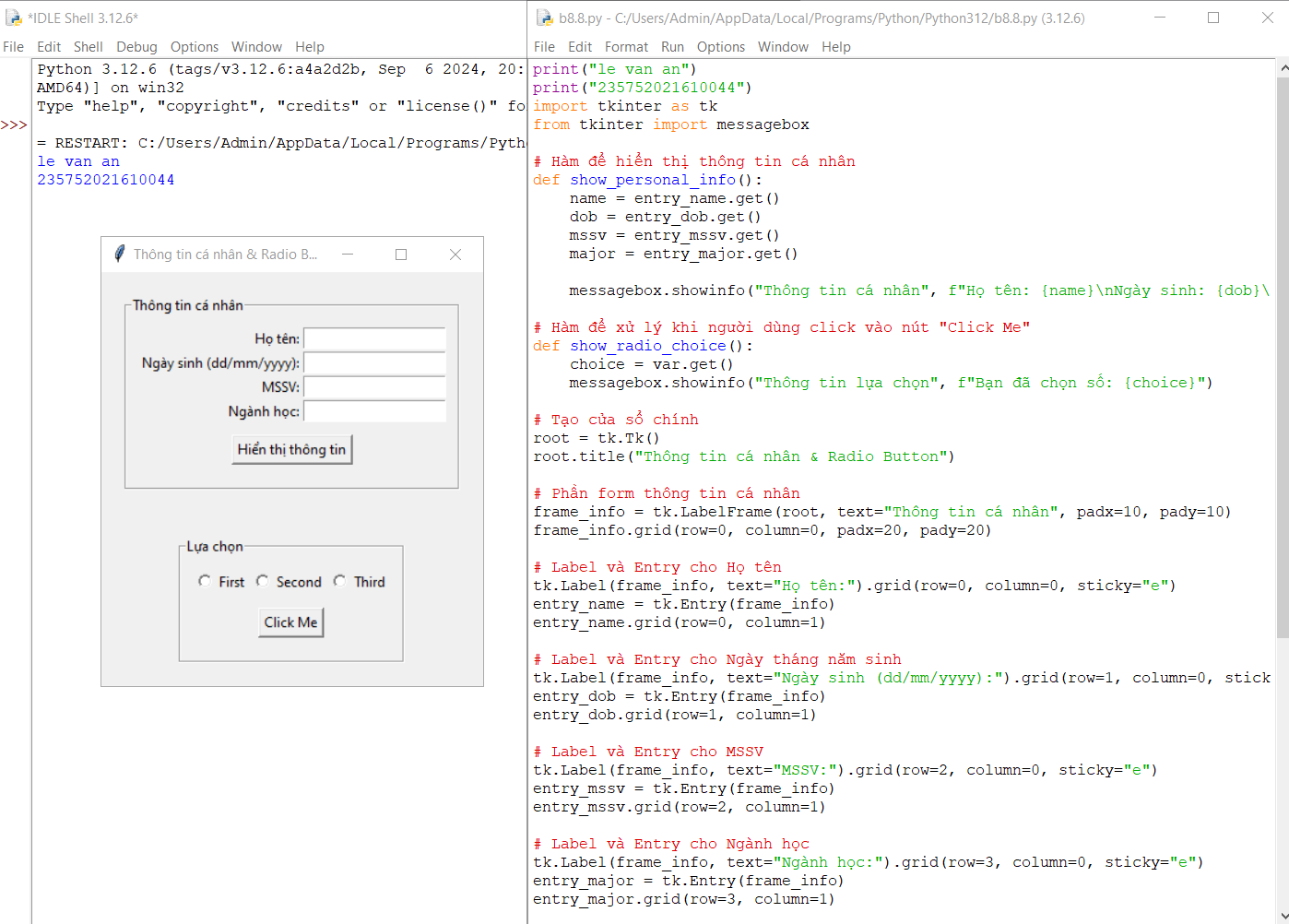
6. Viết chương trình thực hiện tạo menu theo các bước sau:



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



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



#### 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/