TRƯỜNG ĐẠI HỌC VINH

**VIỆN KỸ THUẬT VÀ CÔNG NGHỆ**



BÁO CÁO THỰC HÀNH

**KỸ THUẬT LẬP TRÌNH**

MÃ HỌC PHẦN: ELE20004

SVTH: ĐINH QUYỀN AN

MSSV: 235752021610059

GVHD: MAI THẾ ANH

**NGHỆ AN - 2024**

**MỤC LỤC**

[Bài 1. Thực hiện các thuật toán bằng phần mềm Flowgorithm 2](#_Toc184193537)

[Bài 2. Các cú pháp, kiểu dữ liệu, lệnh điều khiển trong lập trình Python 7](#_Toc184193538)

[Bài 3. Lập trình hàm trong Python 12](#_Toc184193539)

[Bài 4. Các kiểu dữ liệu có cấu trúc trong Python 15](#_Toc184193540)

[Bài 5. Thiết kế module trong Python 25](#_Toc184193541)

[Bài 6. Lập trình hướng đối tượng trong Python 29](#_Toc184193542)

[Bài 7. Thao tác trên tập tin và thư mục trong Python 34](#_Toc184193543)

Bài [6.Chương trình Python để đọc n dòng cuối cùng của tệp 37](#_Toc184193544)

[Bài 8. Lập trình giao diện trong Python 39](#_Toc184193545)

# Thực hiện các thuật toán bằng phần mềm Flowgorithm

## Mục đích

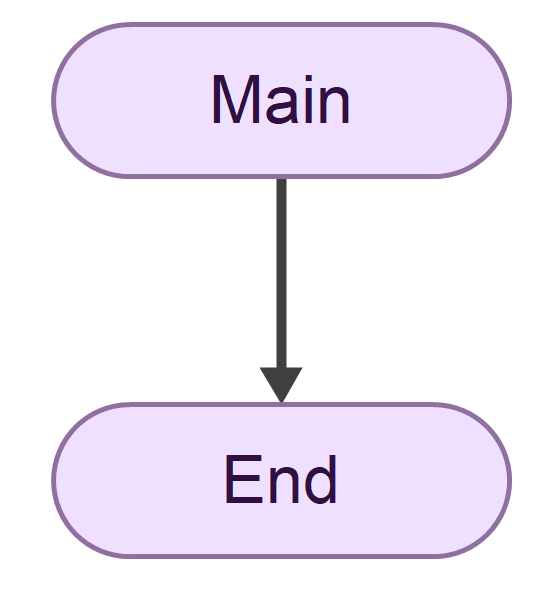
* Sử dụng phần mềm Flowgorithm trong thiết kế và biểu diễn thuật toán;
* Xây dựng thuật toán cho các bài toán cụ thể trên Flowgorithm.

## Các bước thực hiện và kết quả

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

B1: Tạo chương trình mới

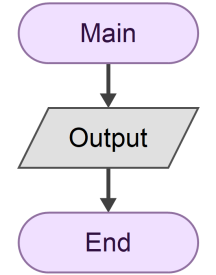
* Khi bắt đầu một sơ đồ mới, chúng ta sẽ thấy hai hình chữ nhật tròn được gọi là "terminals". Những biểu tượng này đại diện cho sự bắt đầu và kết thúc chương trình của bạn.
* Nhiều sơ đồ thuật toán hiển thị văn bản "Begin" trong terminal. Flowgorithm sử dụng văn bản "Main". Hầu hết các ngôn ngữ lập trình bắt đầu với các nỗ lực "Main" và Flowgorithm cũng vậy.



#### Bắt đầu và kết thúc chương trình

B2: Thêm các khối chức năng

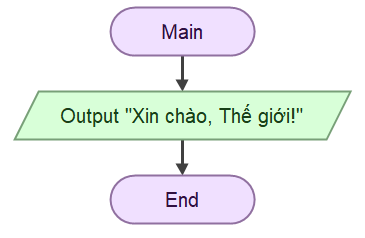
* Tất cả mọi thứ trong một sơ đồ được thể hiện bằng các khối hình. Các khối được thêm vào giữa các terminals Main và End.
* Để thêm hình dạng, di chuyển con trỏ chuột của bạn trên một dòng. Nếu có thể thêm một khối, dòng sẽ chuyển sang màu cam.
* Bấm đúp hoặc bấm chuột phải để thêm hình.
* Hiển thị giao diện với các khối cần thêm.
* Lựa chọn và click vào khối cần thêm sẽ được chương trình như hình 2.



#### Chương trình đã thêm các khối chức năng

B3: Định nghĩa chức năng cho khối

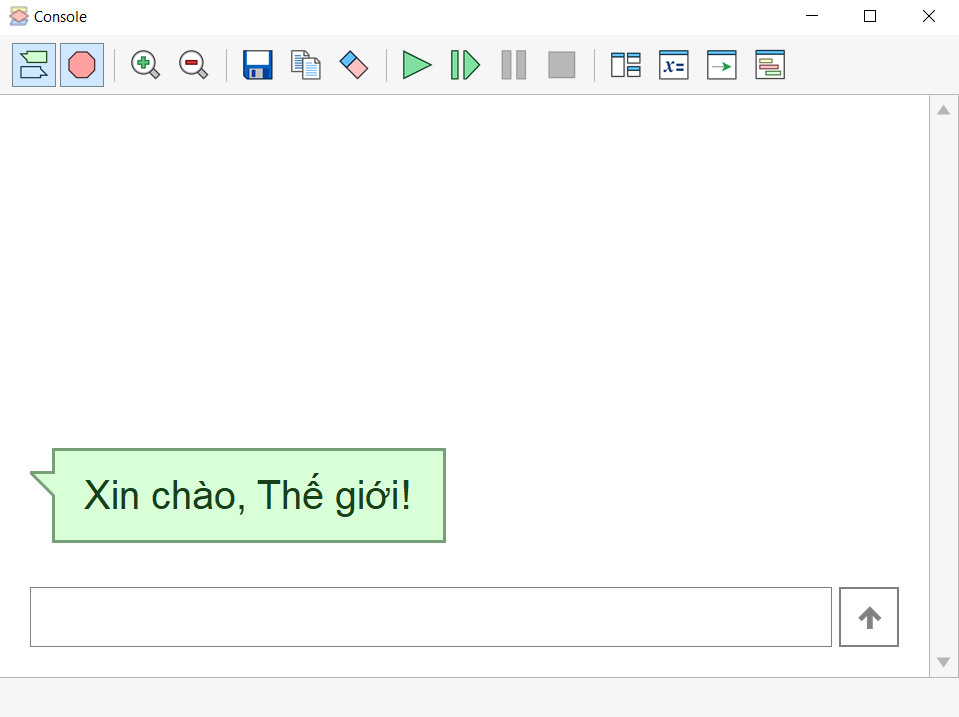
* Cửa sổ “Output Properties” xuất hiện, chúng ta có thể nhập vào các chứng năng theo cú pháp quy định, chương trình sau khi định nghĩa chức năng được chỉ ra trong hình 3.



#### Chương trình sau khi định nghĩa cho các khối chức năng thêm vào

B4: Khởi chạy chương trình đã thiết kế

* Bấm F5 hoặc nút “Run” trên menu của chương trình.



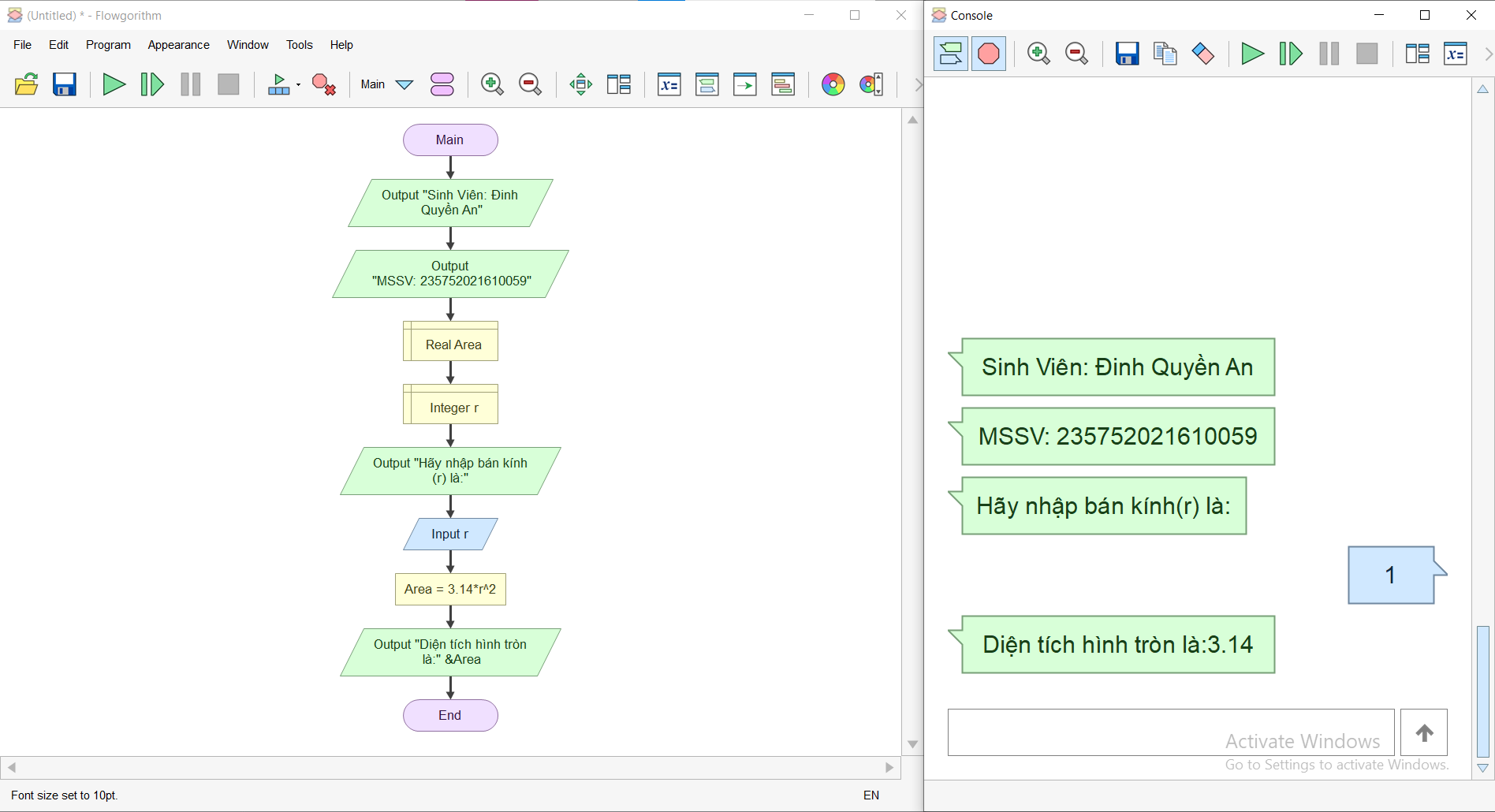
#### Cửa sổ điều khiển hiện thị kết quả

B5: Xem mã nguồn hoặc các biến sử dụng trong chương trình sử dụng menu “Tools”

ở thanh công cụ.

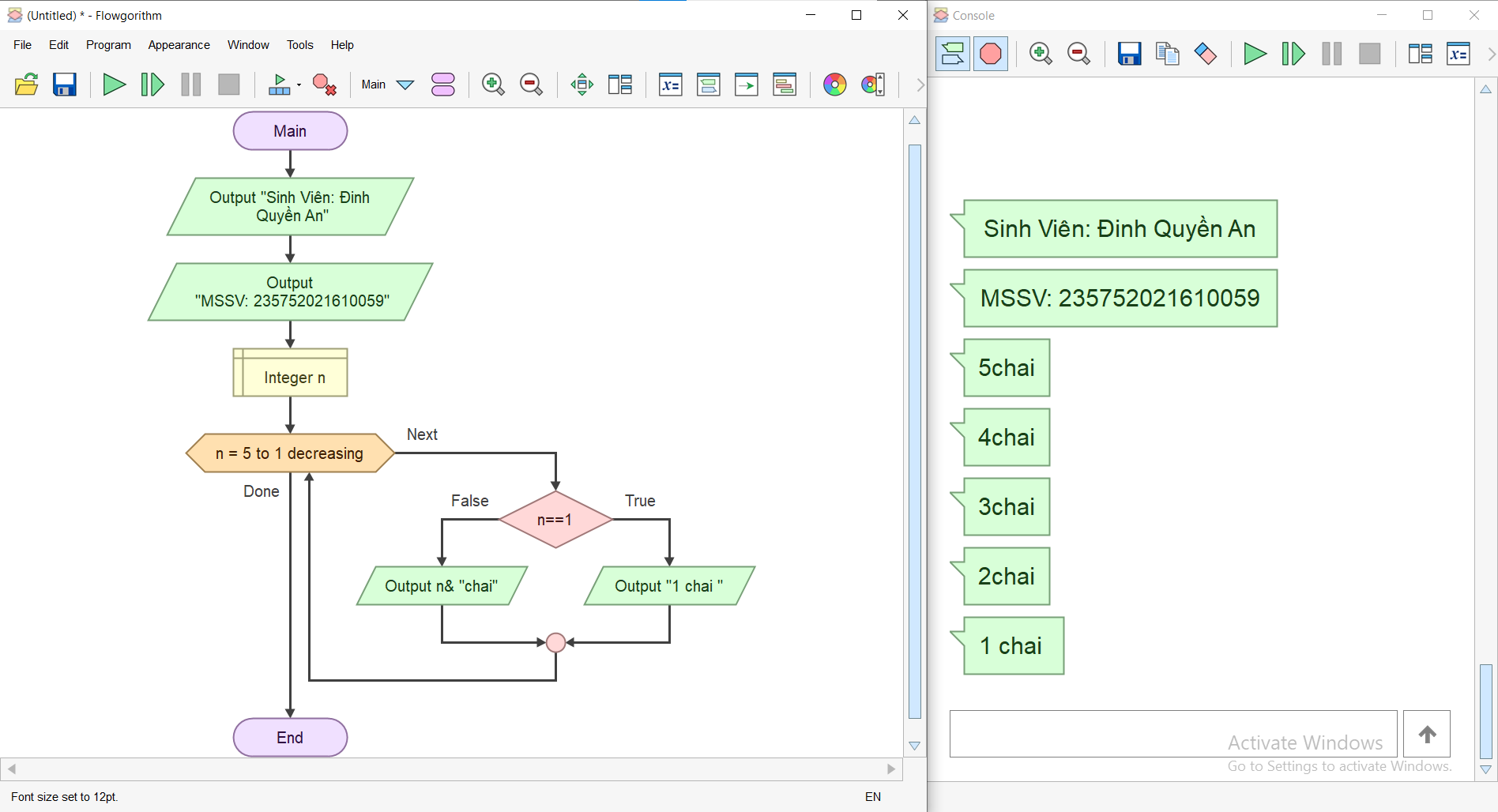
### Sử dụng Flowgorithm xây dựng chương trình giải quyết các bài toán

##### Tính diện tích hình tròn



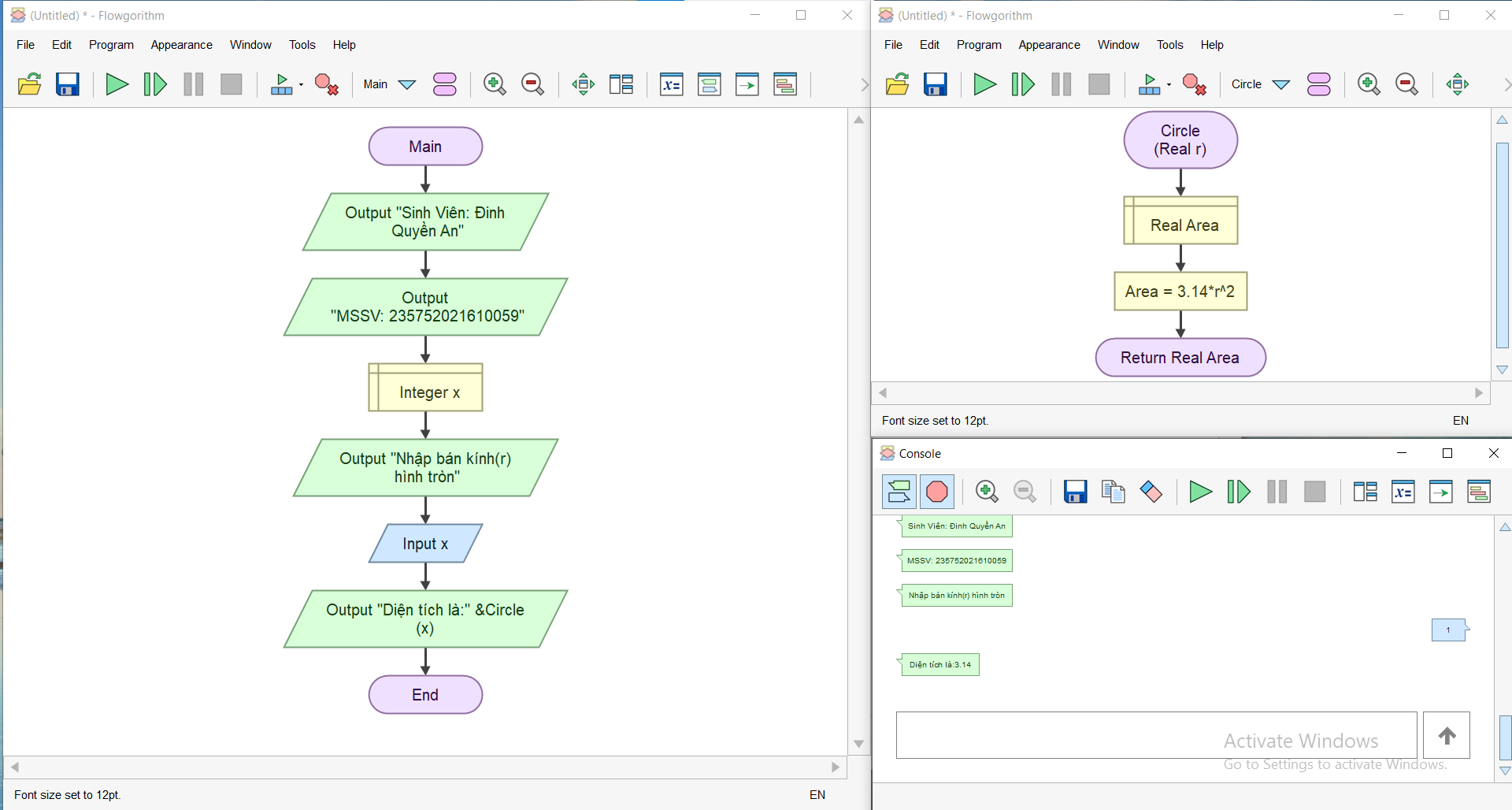
#### Sơ đồ giải thuật bài toán tính diện tích hình tròn

##### Nhập và in số lượng chai sử dụng vòng lặp for



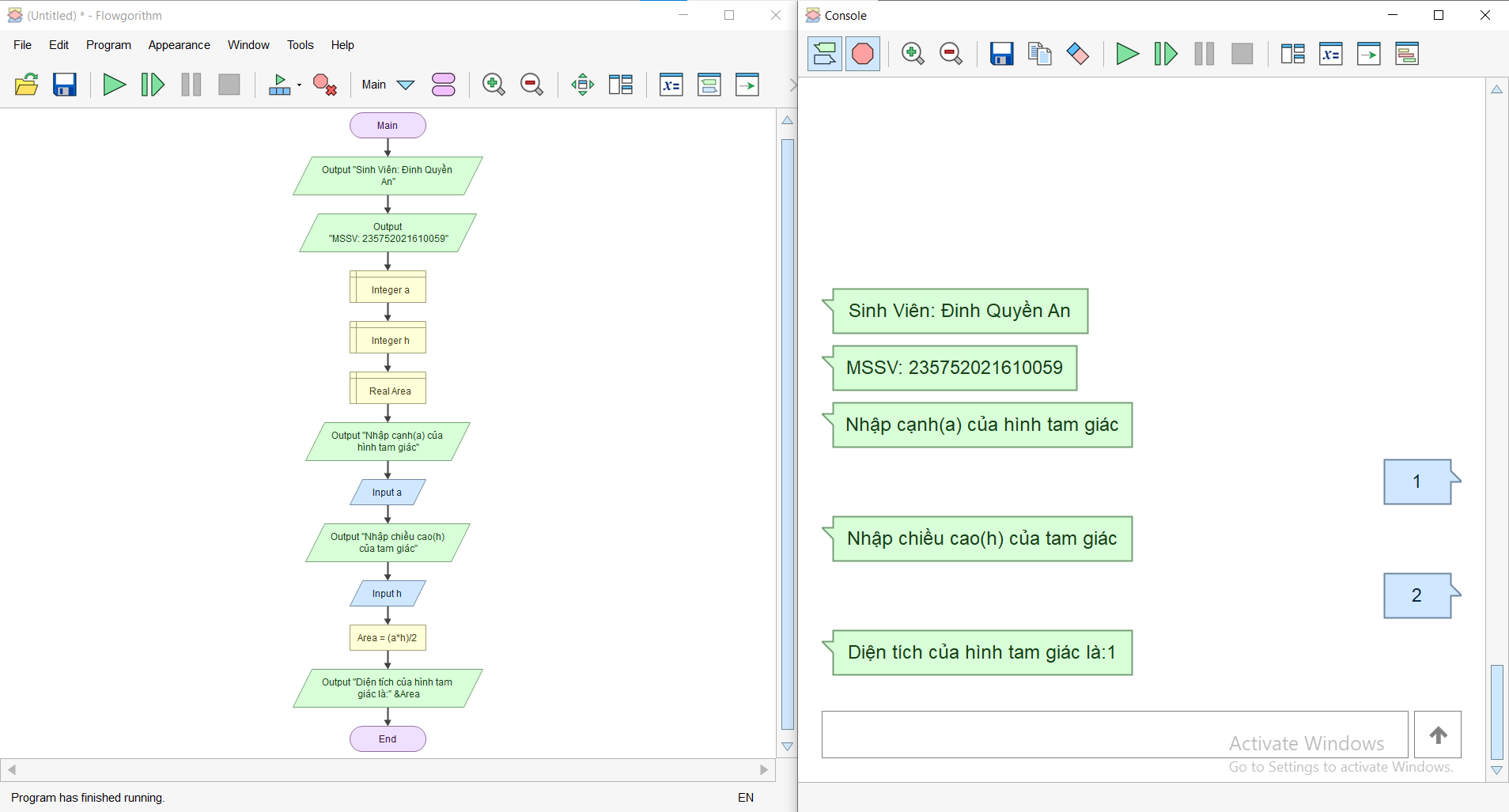
#### Giải thuật bài toán nhập và in số lượng chai sử dụng vòng lặp for

##### Tính diện tích hình tròn sử dụng hàm(chương trình con)



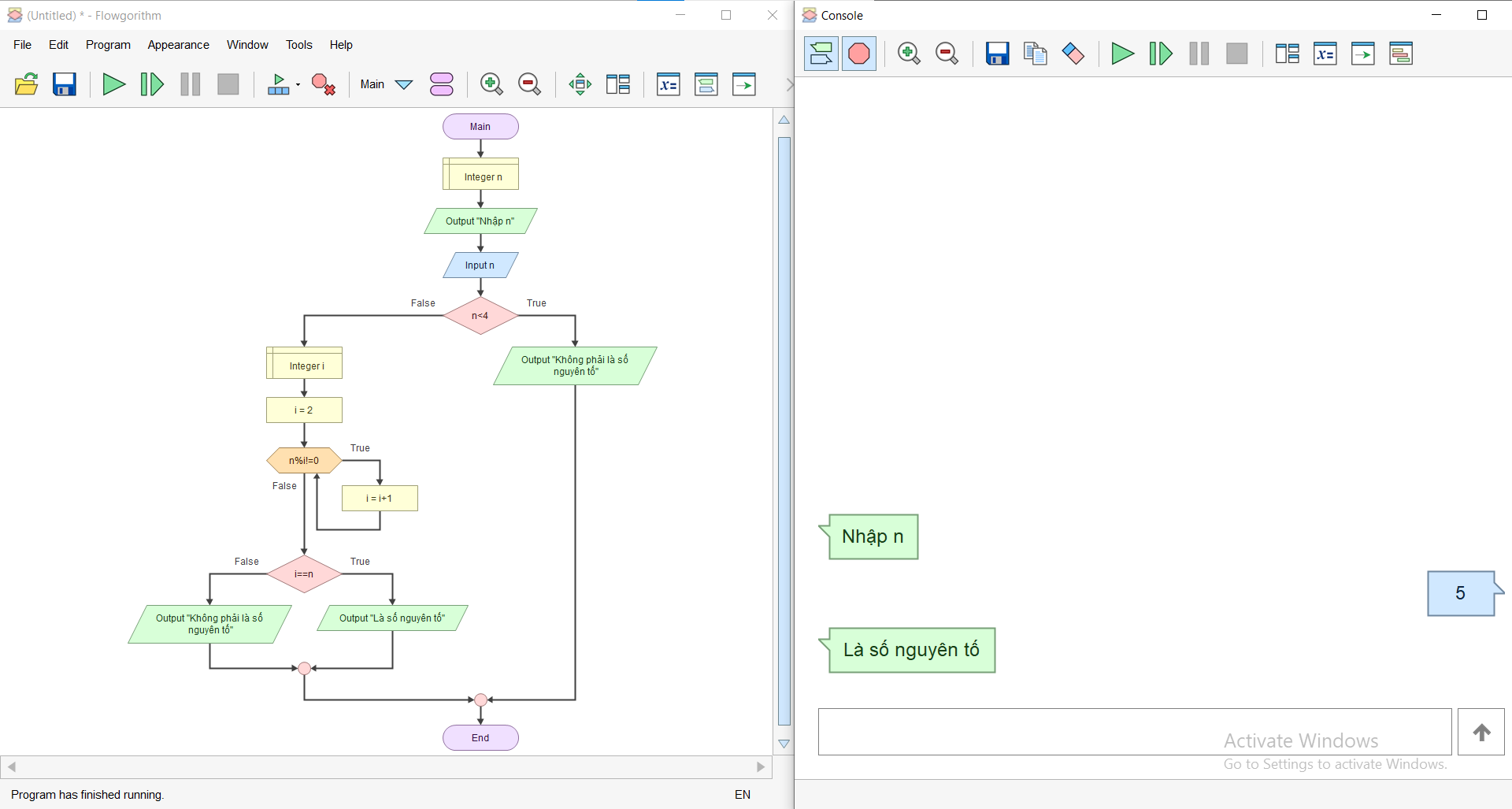
#### Sử dụng hàm cho giải thuật tính diện tích hình tròn

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



#### Sơ đồ giải thuật bài toán tính diện tích hình tròn

##### 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)



#### Xây dựng thuật toán kiểm tra tính nguyên tố

## Câu hỏi kiểm tra

Câu 1: Ý nghĩa các khối sử dụng trong chương trình

* **Khối bắt đầu (Start):** Đánh dấu điểm bắt đầu của chương trình.
* **Khối kết thúc (End):** Đánh dấu điểm kết thúc của chương trình.
* **Khối nhập (Input):** Dùng để nhập dữ liệu từ người dùng, thường là thông qua một hộp thoại.
* **Khối xuất (Output):** Dùng để hiển thị kết quả ra màn hình, cũng thông qua một hộp thoại.
* **Khối gán (Assignment):** Gán giá trị cho một biến.
* **Khối tính toán (Calculation)**: Thực hiện các phép tính toán số học.
* **Khối điều kiện (Decision):** Kiểm tra một điều kiện và thực hiện các nhánh khác nhau tùy thuộc vào kết quả của điều kiện đó (thường sử dụng các từ khóa như if, else).
* **Khối lặp (Loop):** Thực hiện một khối lệnh nhiều lần cho đến khi một điều kiện nào đó không còn đúng (thường sử dụng các từ khóa như while, for).
* **Khối gọi hàm (Call):** Gọi đến một hàm đã được định nghĩa trước đó.

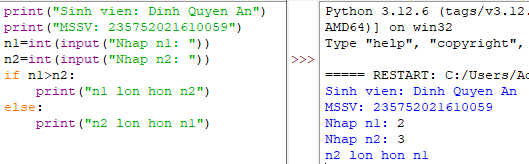
# Các cú pháp, kiểu dữ liệu, lệnh điều khiển trong lập trình Python

## Mục đích

Giúp sinh viên nắm bắt 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.

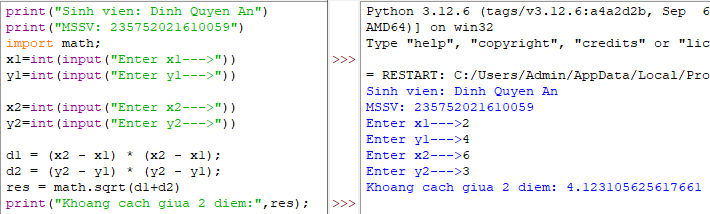
## Các bước thực hiện và kết quả

###### Viết đoạn chương trình sau và sửa lỗi:



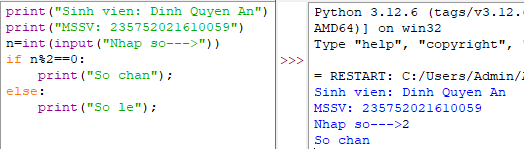
#### Sửa lỗi chương trình

###### Viết chương trình nhập hai điểm và tính khoảng cách:

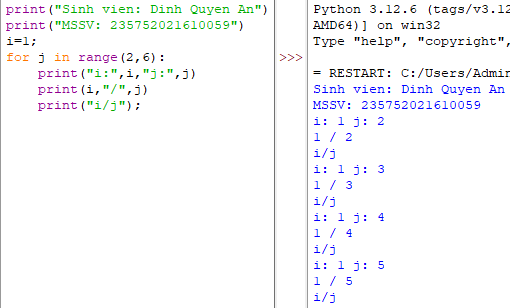


#### Chương trình nhập hai điểm và tính khoảng cách

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

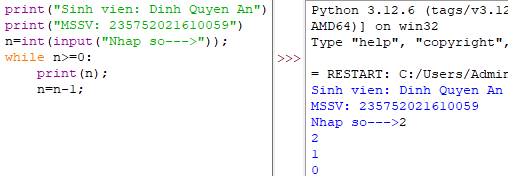


###### Viết chương trình in ra màn hình số nghịch đảo và kết quả dưới dạng thập phân của một dãy số tự nhiên trong khoảng (a,b):



#### Chương trình in ra màn hình số nghịch đảo và kết quả

###### 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:

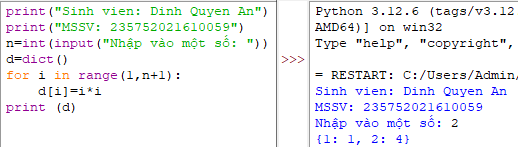


#### Chương trình nhập vào một số tự nhiên n

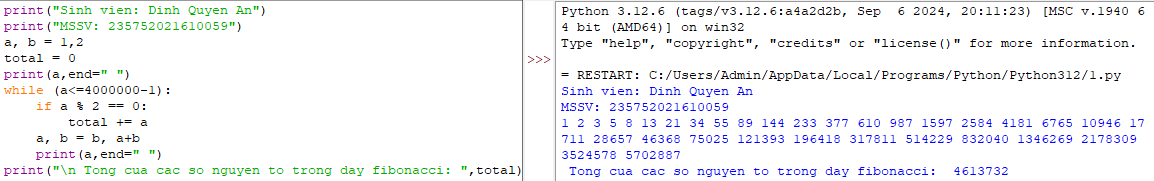
###### Viết chương trình tìm tất cả các số chia hết cho 7 nhưng không phải bội số của 5, nằm trong đoạn 2000 và 3200 (tính cả 2000 và 3200). Các số thu được sẽ được in thành chuỗi trên một dòng, cách nhau bằng dấu phẩy:

###### 

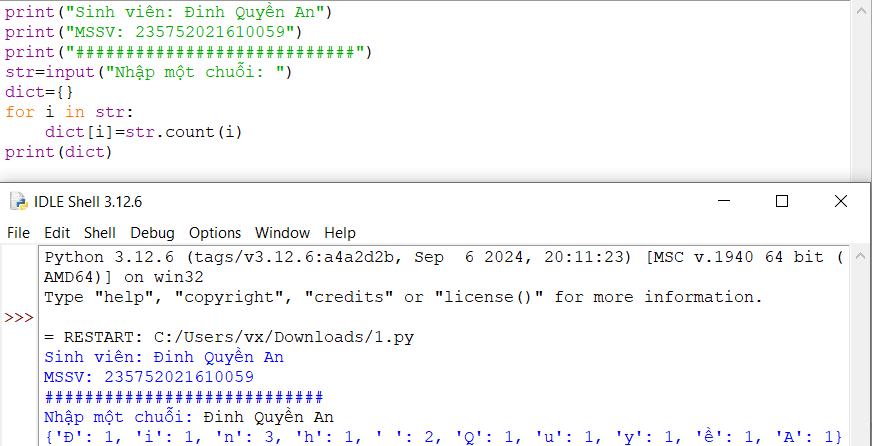
###### Với số nguyên n nhất định, hãy viết chương trình để tạo ra một dictionary chứa (i, i\*i) như là số nguyên từ 1 đến n (bao gồm cả 1 và n) sau đó in ra dictionary này:



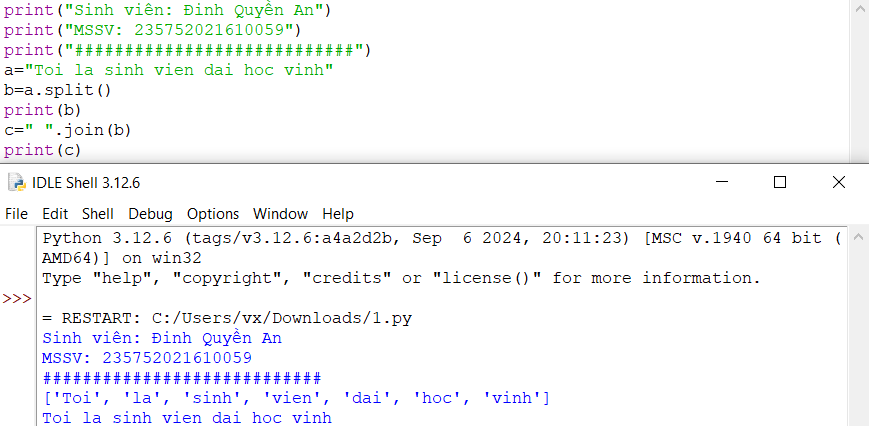
###### Viết chương trình in ra màn hình dãy số Fibonacci nhỏ hơn 4.000.000, tìm tổng các số chẵn trong dãy đã in:



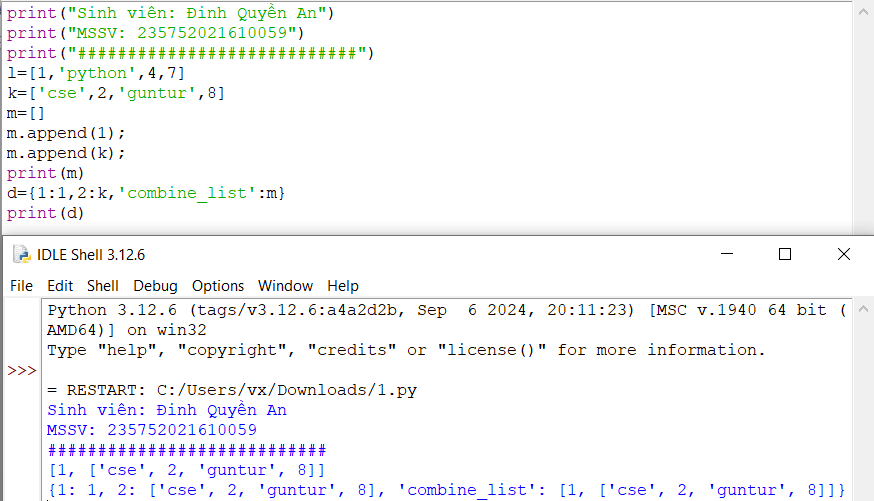
###### Viết chương trình đếm số ký tự trong 1 xâu ký tự nhập vào từ bàn phím, lưu các ký tự vào cấu trúc từ điển:



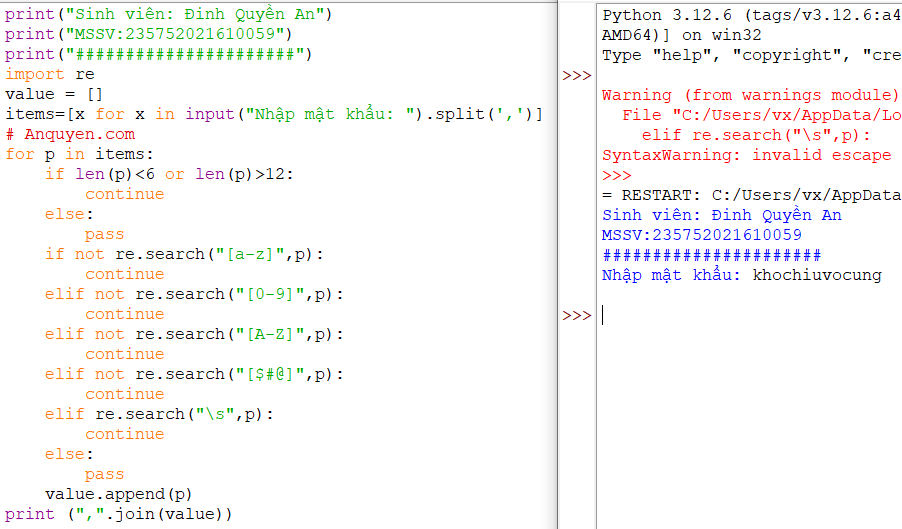
###### Viết chương trình sử dụng các phương thức split và join để tách nhập xâu ký tự:



###### Viết chương trình kết nối các danh sách vào từ điển:



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



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

###### 

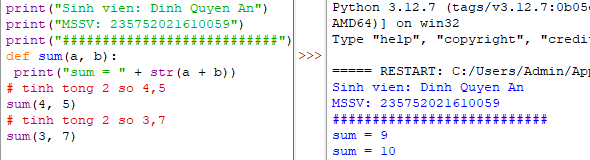
# Lập trình hàm trong Python

## Mục đích

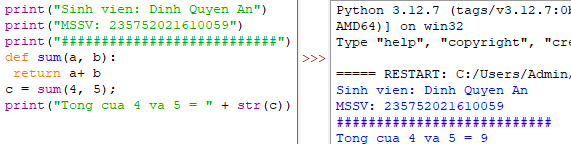
## Giúp sinh viên nắm bắt các kiến thức trong lập trình python sử dụng hàm.

## Các bước thực hiện và kết quả

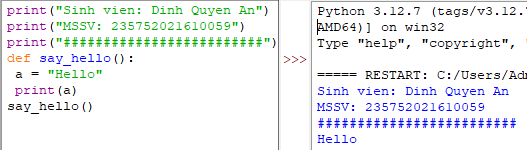
**1.** Viết hàm sum() tính tổng hai số



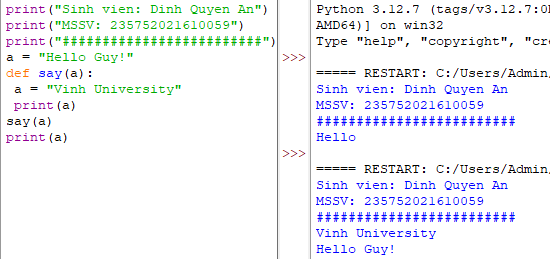
**2.** Viết hàm sum() với kết quả trả về



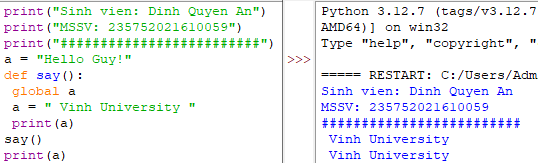
###### **3.** Tìm và sửa lỗi chương trình



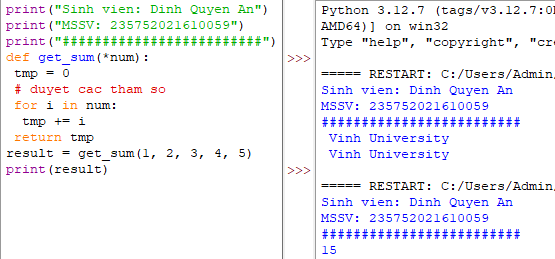
**4.** Viết chương trình có phạm vi biến như sau



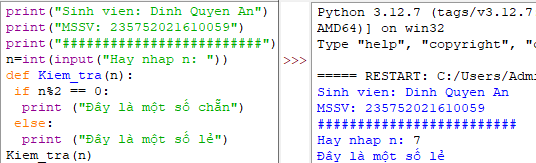
###### **5.** Viết chương trình sau và xem sự thay đổi của biến



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



**7.** 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.



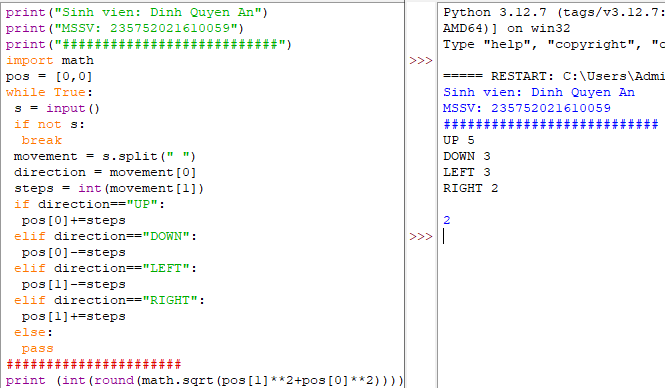
**8.** Một Robot di chuyển trong mặt phẳng bắt đầu từ điểm đầu tiên (0,0). Robot có thể di chuyển theo hướng UP, DOWN, LEFT và RIGHT với những bước nhất định. Dấu di chuyển của robot được đánh hiển thị như sau:

UP 5

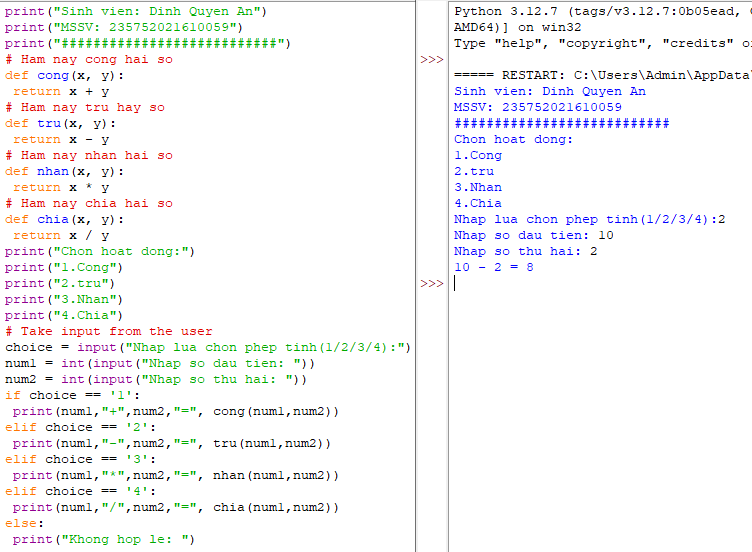
DOWN 3

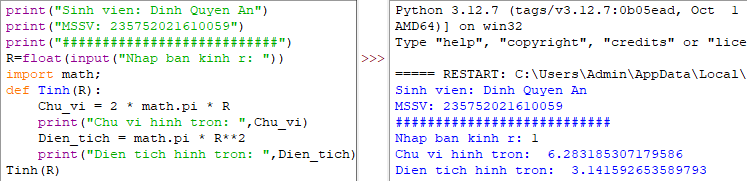
LEFT 3

RIGHT 3



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

**10.** 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.



# Các kiểu dữ liệu có cấu trúc trong Python

## Mục đích

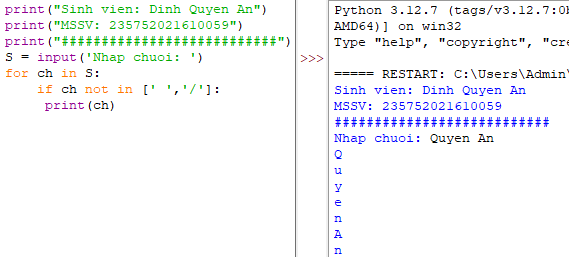
Giúp sinh viên nắm bắt các kiểu dữ liệu có cấu trúc trong lập trình python.

## Các bước thực hiện và kết quả

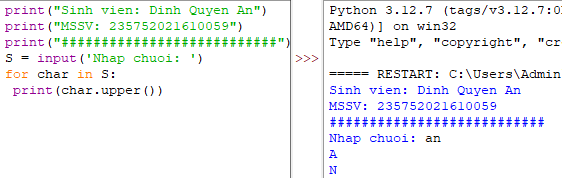
**1.** Nhập chuỗi S và in ra từng kí tự của S, mỗi kí tự trên một dòng.

## 

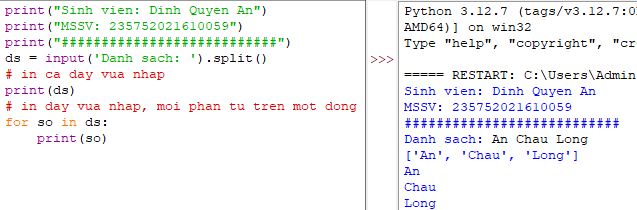
**2.**Chỉnh sửa ví dụ trên: hãy bỏ qua không in ra những kí tự “không nhìn thấy” (dấu space và dấu tab).



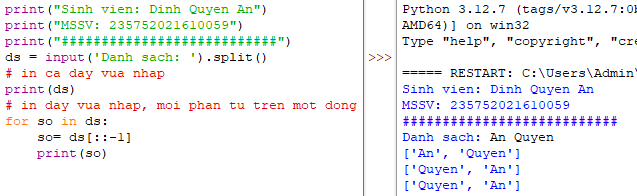
**3.**Chỉnh sửa ví dụ ở bài 1: hãy các kí tự ở dạng IN HOA.



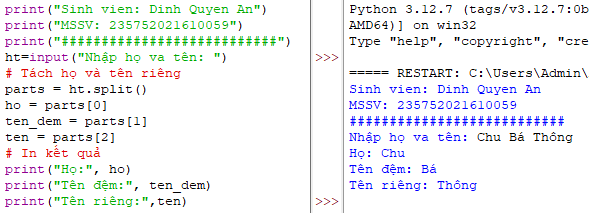
**4.**Nhập một danh sách trên một dòng, mỗi phần tử cách nhau bởi dấu trống hoặc tab, sau đó in ra dãy vừa nhập ra màn hình.



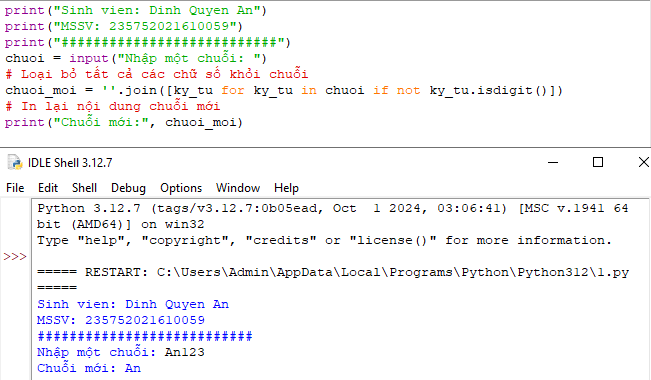
**5.**Chỉnh sửa ví dụ ở bài 4: nhập 1 danh sách các từ từ bàn phím, in ra các từ đó theo thứ tự ngược lại thứ tự vừa nhập (ví dụ nhập “mot hai ba” thì in ra “ba hai mot”).



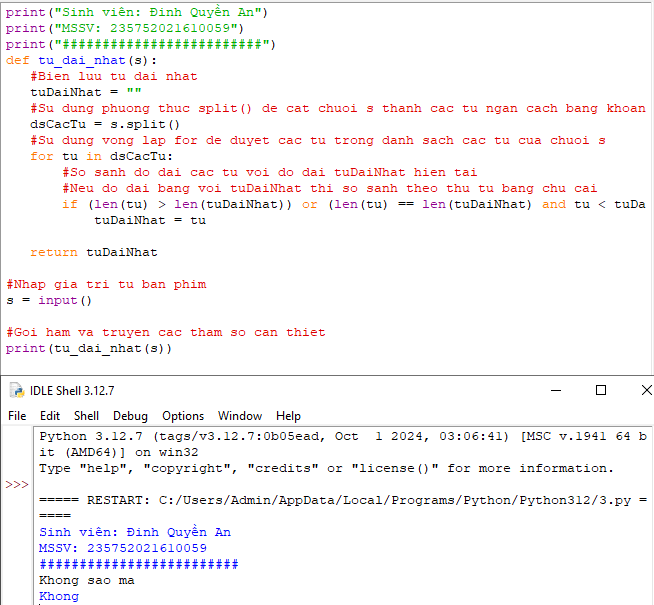
**6.**Nhập một tên người từ bàn phím, hãy tách phần họ và tên riêng của người đó và in chúng ra màn hình (giả thiết họ và tên riêng chỉ gồm một âm).



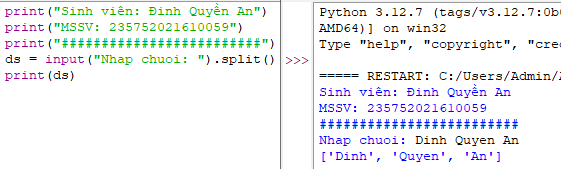
**7.**Nhập một chuỗi từ bàn phím, hãy loại bỏ tất cả các chữ số khỏi chuỗi và in lại nội dung chuỗi mới ra màn hình.



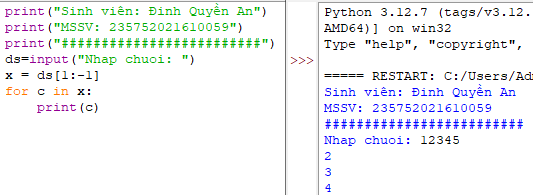
**8.**Nhập một dãy các từ từ bàn phím, hãy in ra từ dài nhất trong dãy vừa nhập, in ra mọi từ có cùng độ dài nhất.



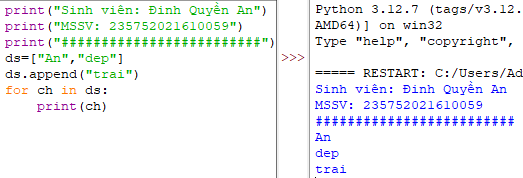
**9.**Nhập một list từ bàn phím.



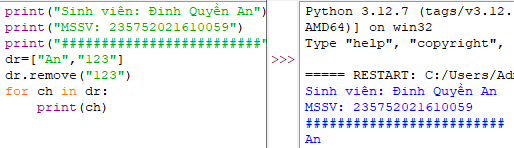
**10.**Cắt list: lấy list nhưng bỏ phần tử đầu và cuối.



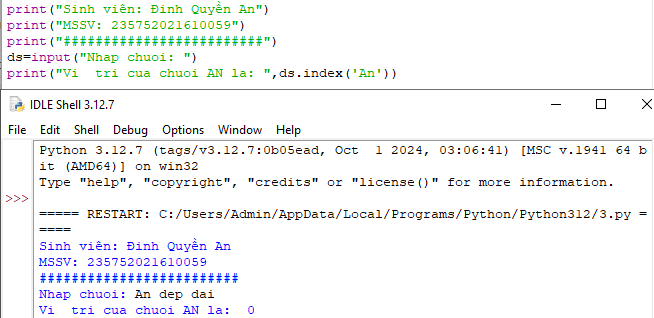
**11.**Thêm phần tử vào list.



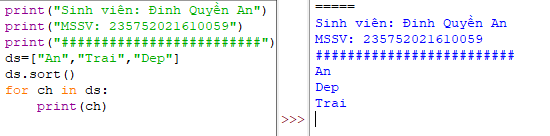
**12.**Bỏ phần tử khỏi list.



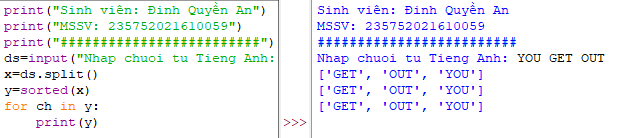
**13.**Tìm kiếm phần tử trong list.



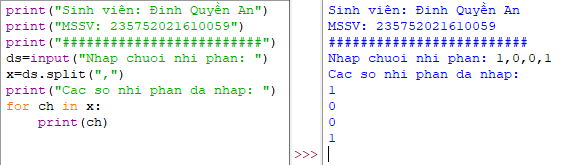
**14.**Sắp xếp các phần tử trong list.



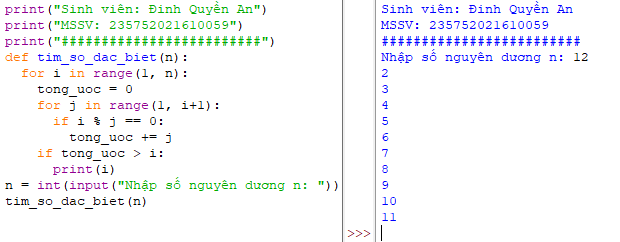
**15**.Người dùng nhập từ bàn phím liên tiếp các từ tiếng Anh viết tách nhau bởi dấu cách. Hãy nhập chuỗi đầu vào và tách thành các từ sau đó in ra màn hình các từ đó theo thứ tự từ điển.



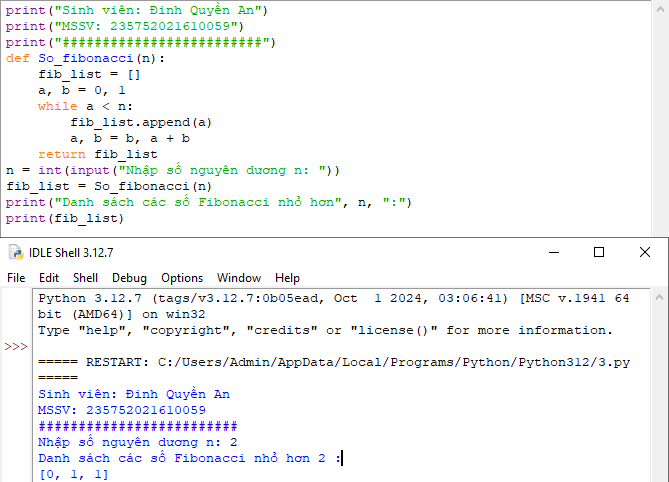
**16.**Người dùng nhập từ bàn phím chuỗi các số nhị phân viết liên tiếp được nối nhau bởi dấu phẩy. Hãy nhập chuỗi đầu vào sau đó in ra những giá trị được nhập.



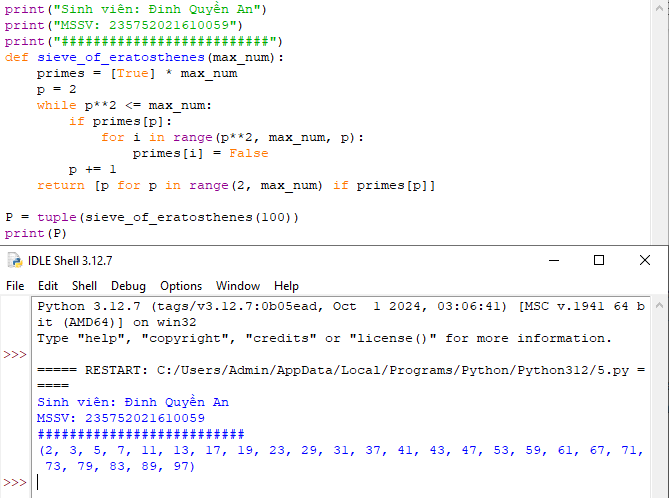
**17.**Nhập số n, in ra màn hình các số nguyên dương nhỏ hơn n có tổng các ước số lớn hơn chính nó.



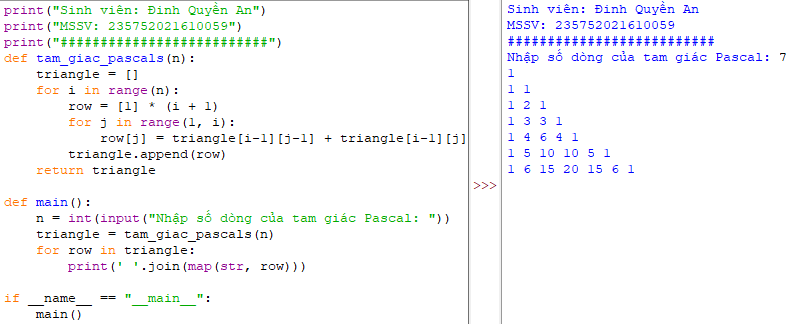
**18.**Hãy nhập số nguyên n, tạo một list gồm các số fibonacci nhỏ hơn n và in ra màn hình.



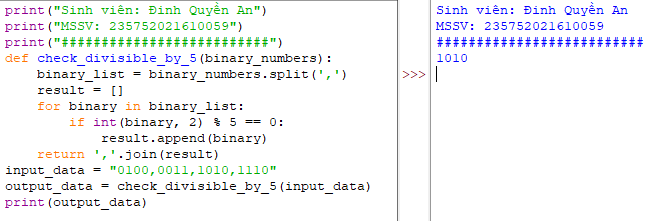
**19.**Hãy tạo ra tuple P gồm các số nguyên tố nhỏ hơn 1 triệu.



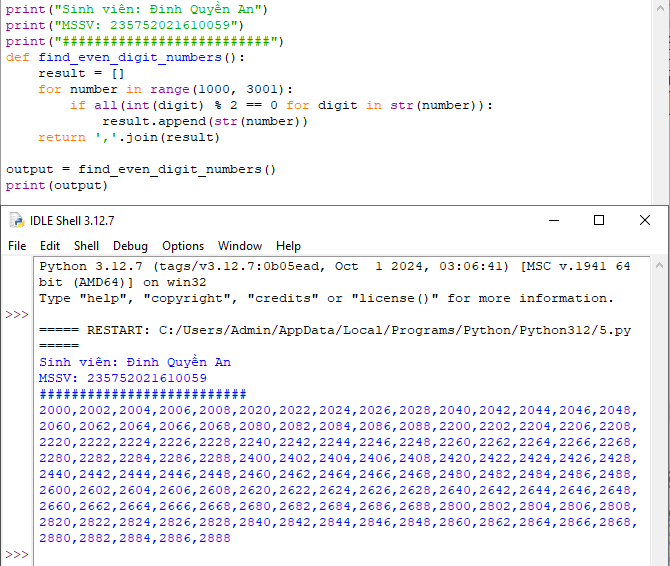
**20.**Nhập n, in n dòng đầu tiên của tam giác pascal.



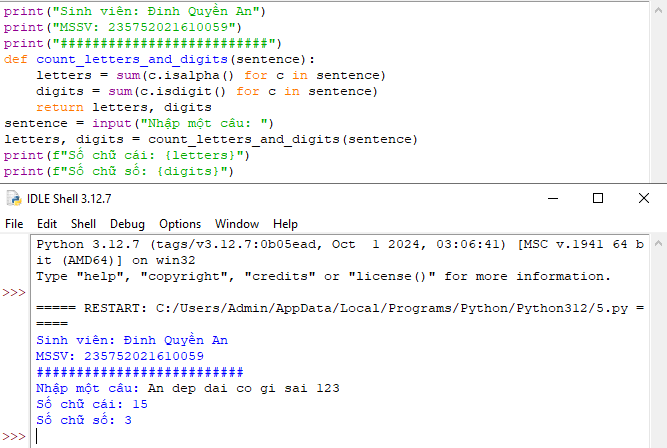
**21.**Viết một chương trình chấp nhận đầu vào là chuỗi các số nhị phân 4 chữ số, phân tách bởi dấu phẩy, kiểm tra xem chúng có chia hết cho 5 không. Sau đó in các số chia hết cho 5 thành dãy phân tách bởi dấu phẩy.



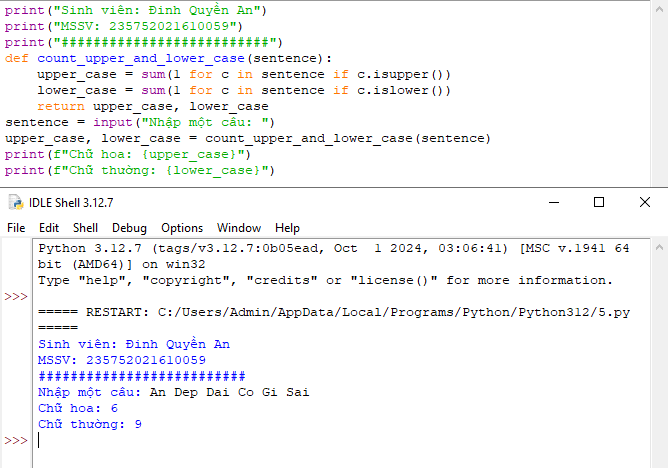
**22.**Viết một chương trình tìm tất cả các số trong đoạn 1000 và 3000 (tính cả 2 số này) sao cho tất cả các chữ số trong số đó là số chẵn. In các số tìm được thành chuỗi cách nhau bởi dấu phẩy, trên một dòng.



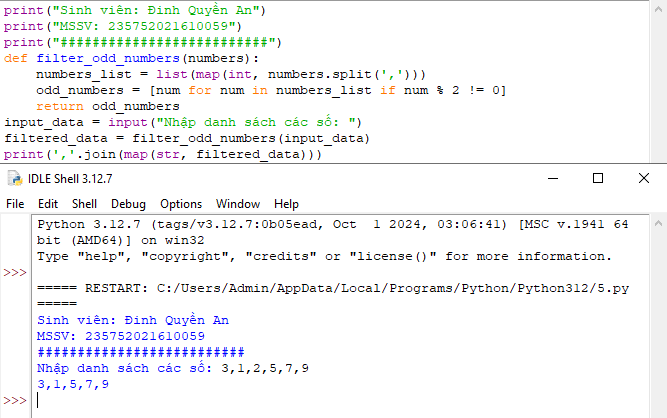
**23.**Viết một chương trình chấp nhận đầu vào là một câu, đếm số chữ cái và chữ số trong câu đó.



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



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



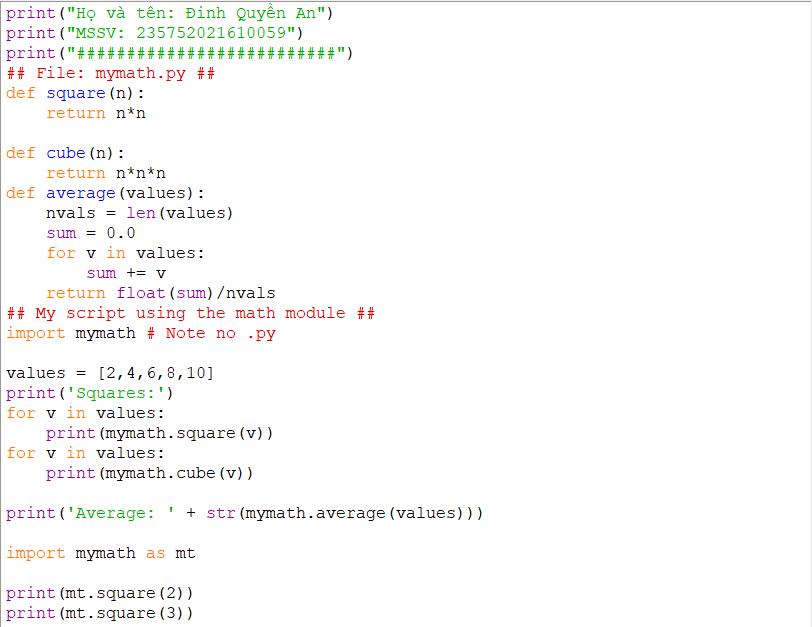
# Thiết kế module trong Python

## Mục đích

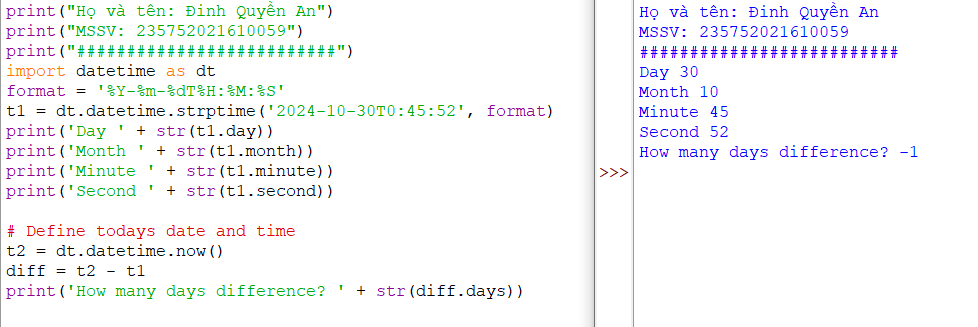
## Giúp sinh viên nắm bắt việc thiết kế module trong lập trình python, sử dụng module thue viện numpy trong các ứng dụng.

## 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.



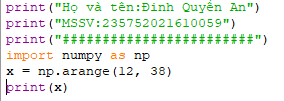
**2.**Sử dụng thư viện tiêu chuẩn của python (datetime)



**3.**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:

[12 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]



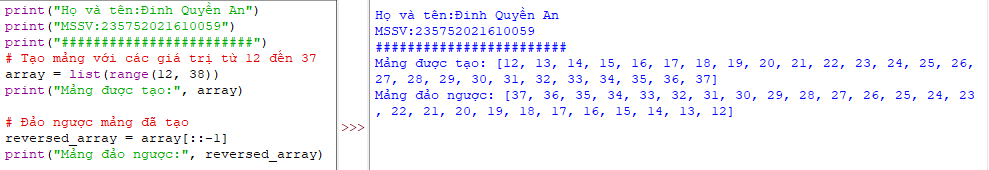
**4.**Viết chương trình để tạo một mảng với các giá trị nằm trong khoảng từ 12 đến 38 và đảo ngược mảng đã tạo (phần tử đầu tiên trở thành cuối cùng) .

Mảng được tạo:

[12 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:

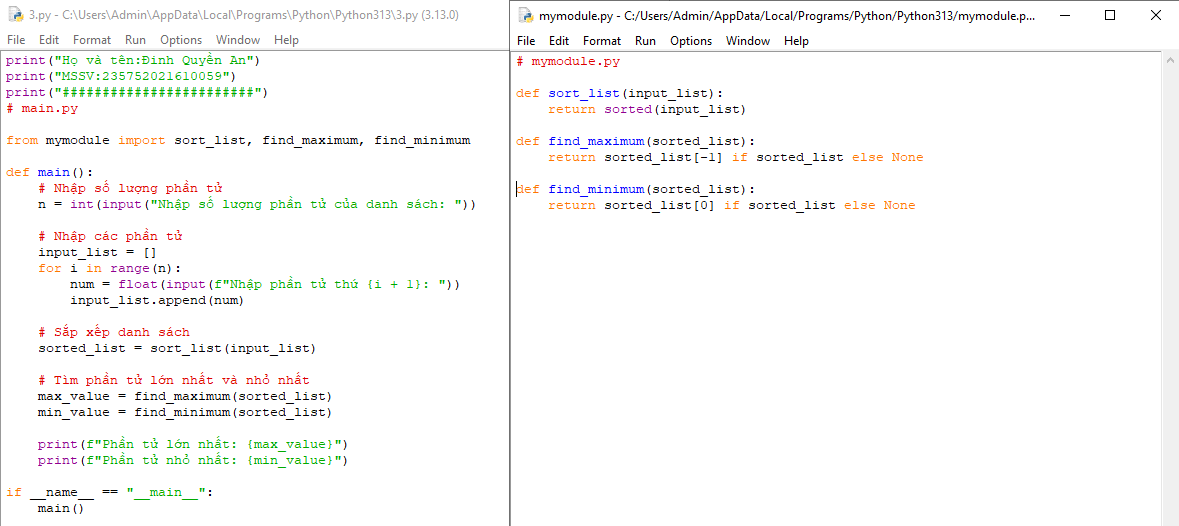
[37 36 35 34 33 32 31 30 29 28 27 26 25 24 23 22 21 20 19 18 17 16 15 14 13 12]



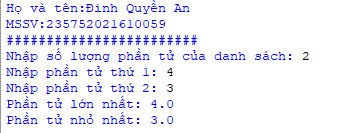
**5.**Viết chương trình tìm phần tử lớn nhất và nhỏ nhất của một danh sách

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

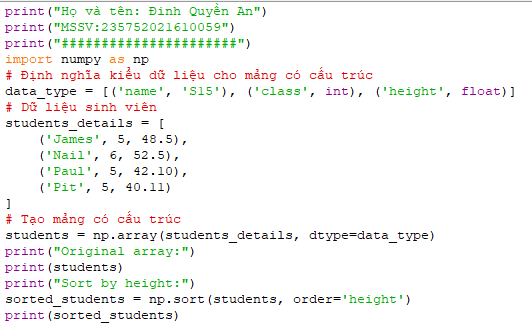
- Phương thức sắp xếp và tìm phần tử lớn nhất được viết thành module



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

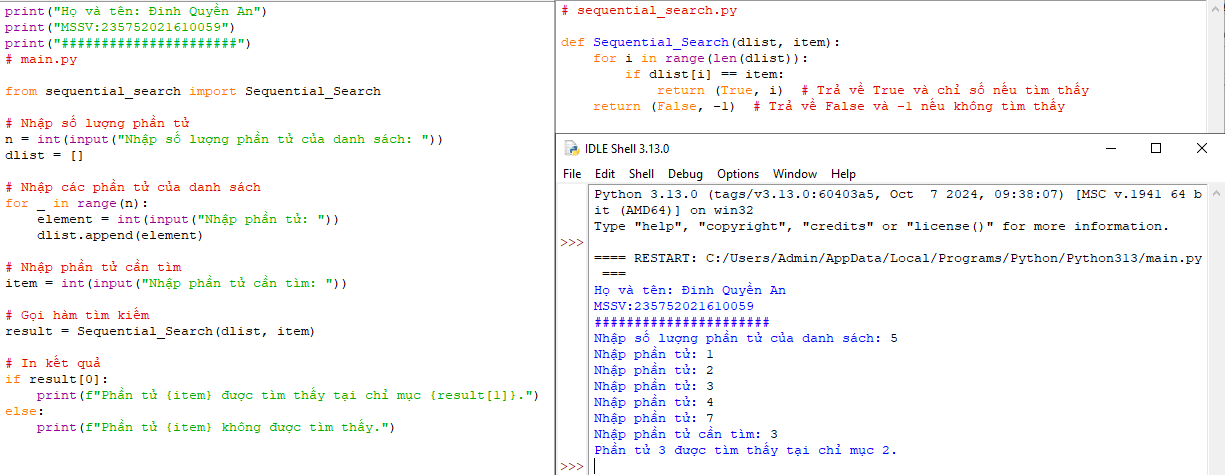


###### **7.**Viết chương trình sử dụng thư viện NumPy để tạo một mảng có cấu trúc từ tên sinh viên, chiều cao, lớp và các kiểu dữ liệu của họ. Bây giờ sắp xếp các mảng thechiều cao.



**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.

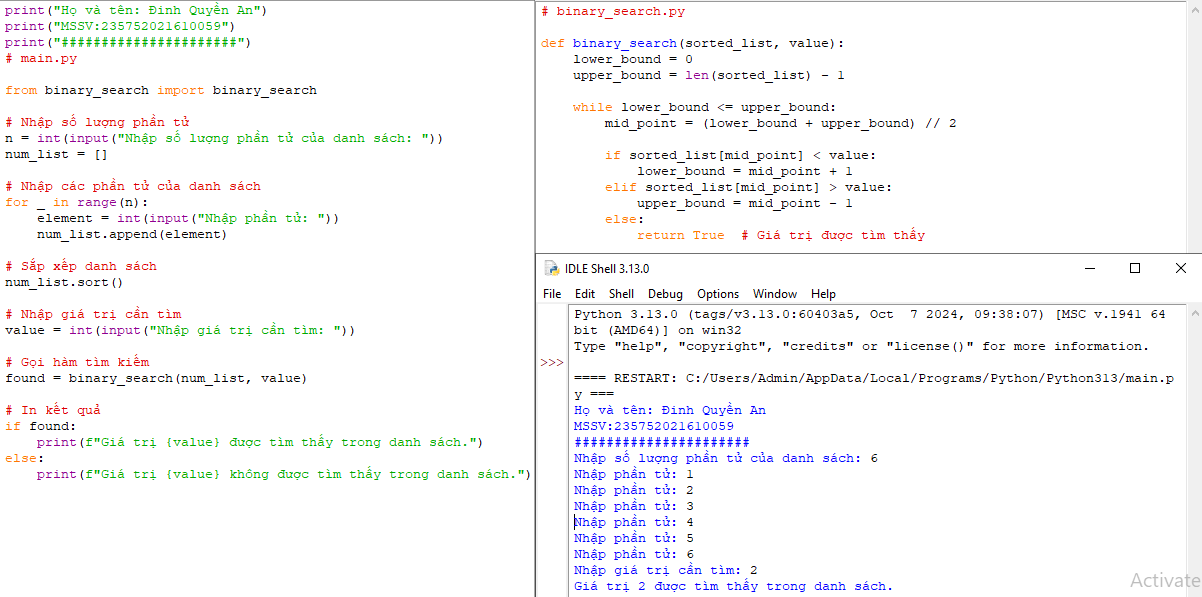
Sequential\_Search([11,23,58,31,56,77,43,12,65,19],31) -> (True, 3)



**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.

binary\_search([1,2,3,5,8], 6) -> False

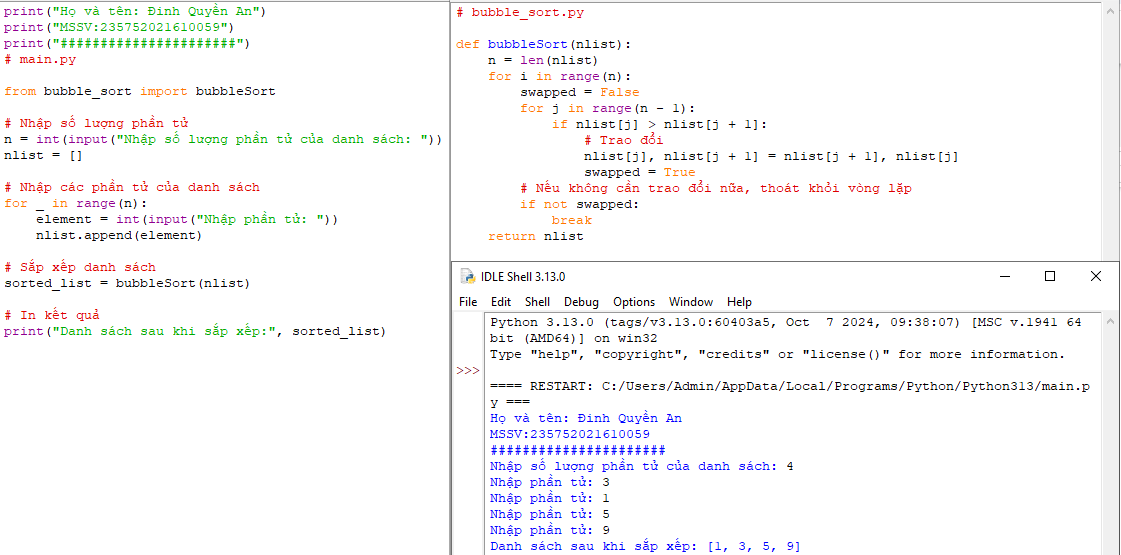
binary\_search([1,2,3,5,8], 5) -> True



**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]



**11.**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:

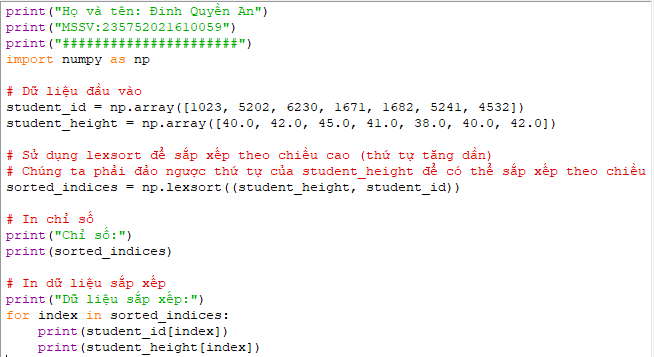
[('James', 5, 48.5 ) ('Nail', 6, 52.5 ) ('Paul', 5, 42.1 ) ('Pit', 5, 40.11)]

Kết quả sắp xếp:

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

###### 

**12.**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()*).



# Lập trình hướng đối tượng trong Python

## Mục đích

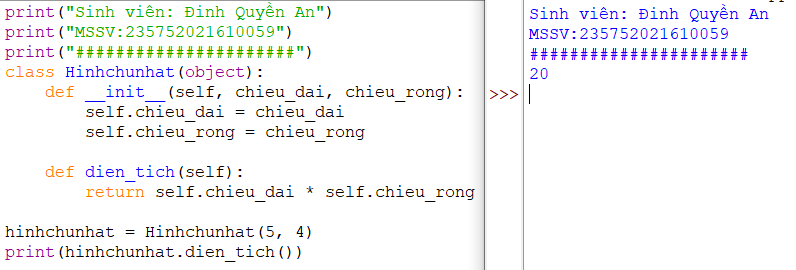
Giúp sinh viên nắm bắt về lập trình hướng đối tượng trong lập trình python.

## 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.

###### 

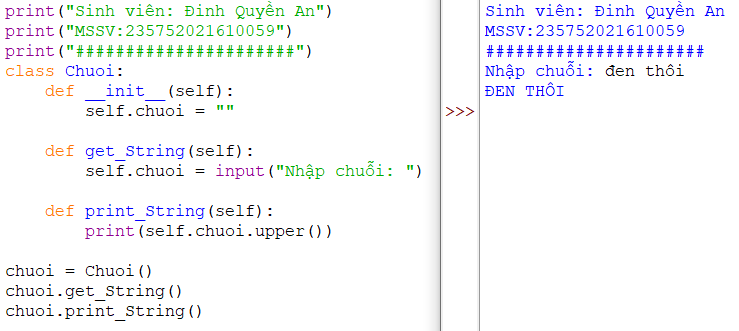
**5.**Viết chương trình Python dưới dạng class để đảo ngược chuỗi từ từng chữ.

Dữ liệu vào : 'hello .py'

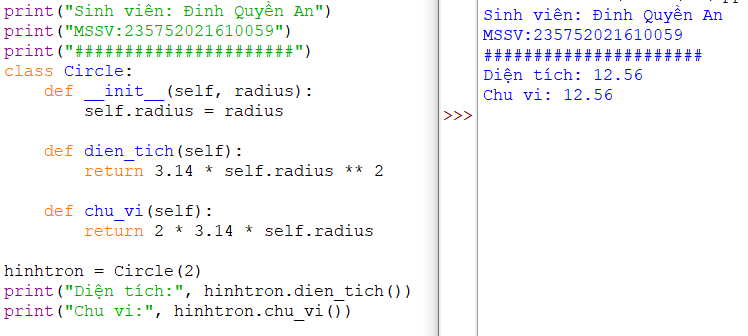
Đầu ra : '.py hello'

#### 

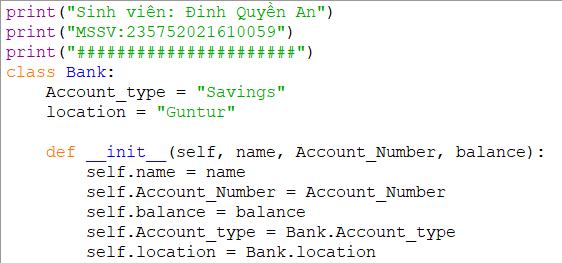
###### **6.**Viết một class Python có hai phương thức get\_String và print\_String. get\_String chấp nhận một chuỗi từ người dùng và print\_String in chuỗi đó bằng chữ in hoa.

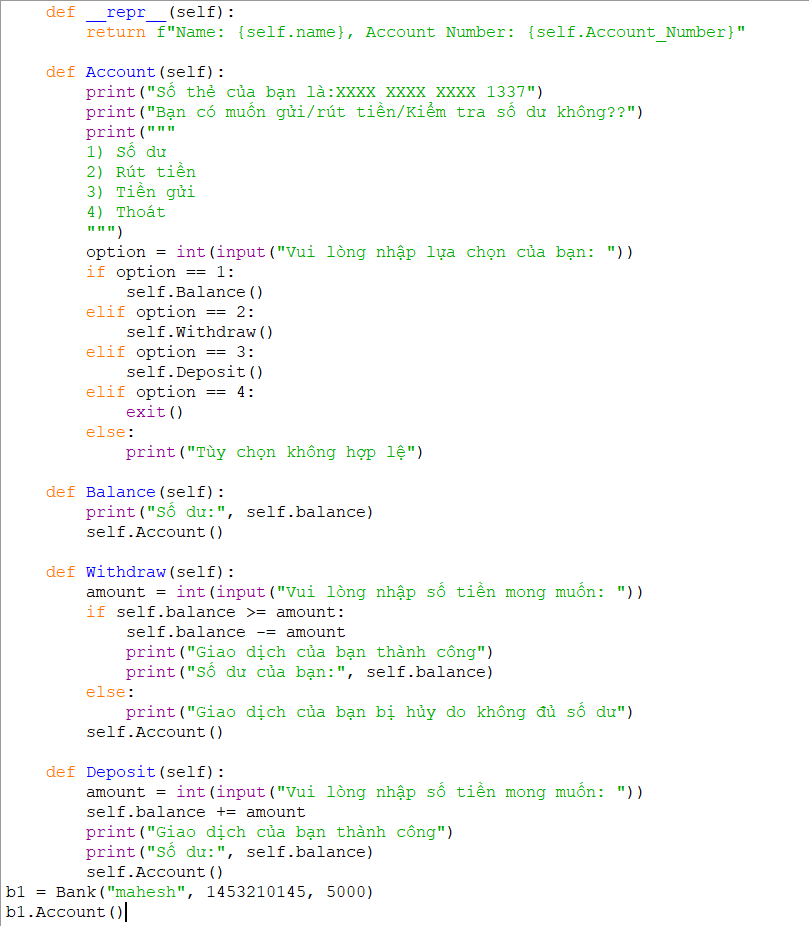


**7.**Viết một class Python có tên Circle được xây dựng theo bán kính và hai phương thức sẽ tính diện tích và chu vi của hình tròn.

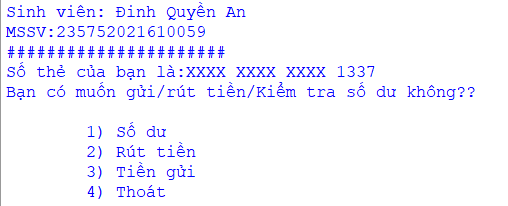


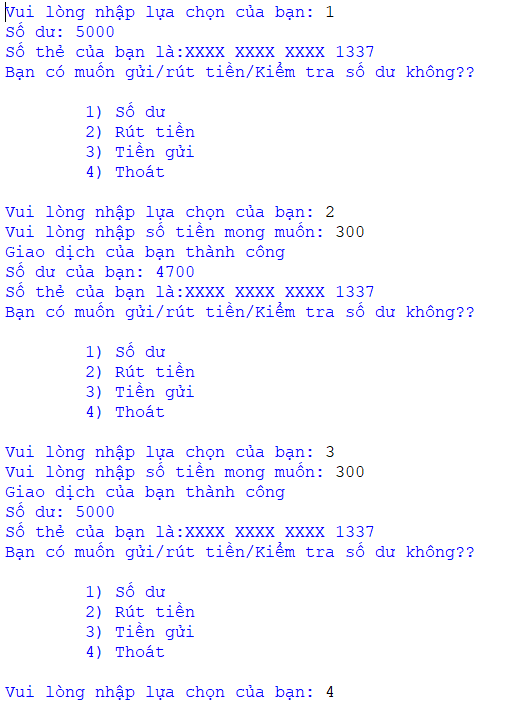
**8.**Chương trình ATM đơn giản





Kết quả:





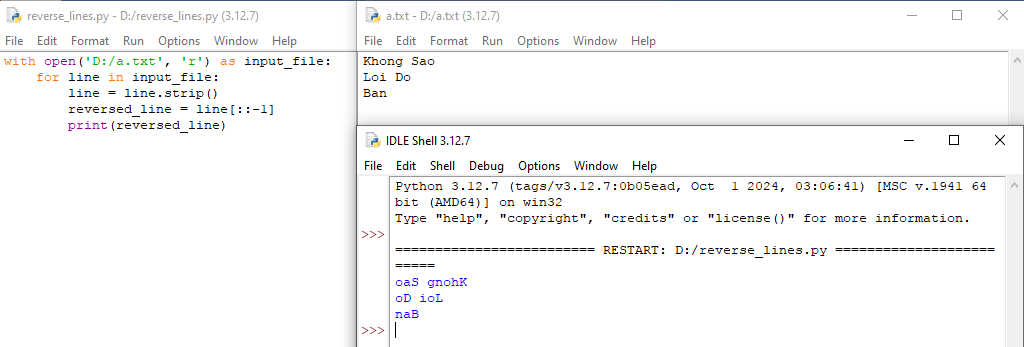
# Thao tác trên tập tin và thư mục trong Python

## Mục đích

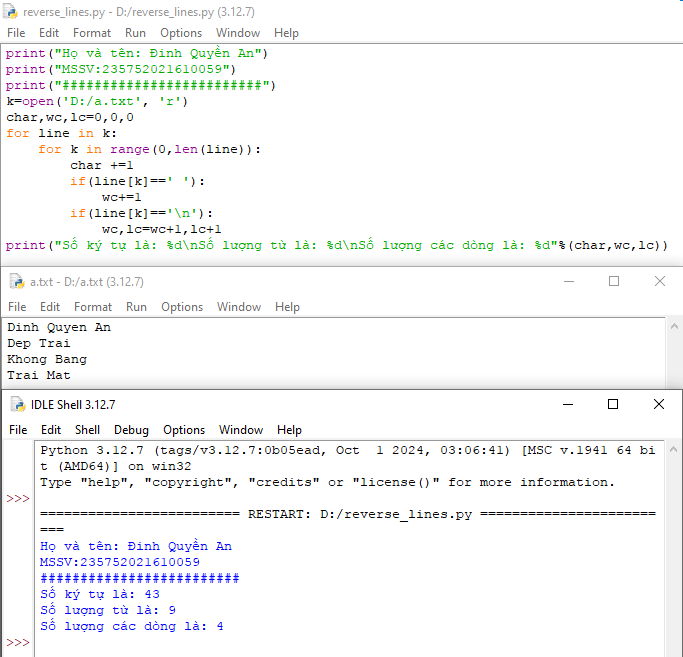
Giúp sinh viên có thể thao tác với các file văn bản sử dụng python

## Các bước thực hiện và kết quả

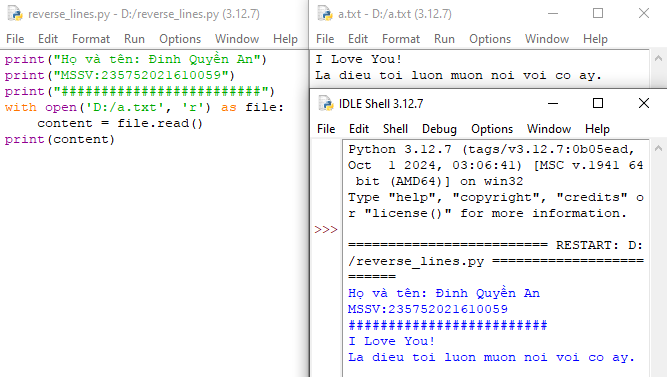
**1.**Chương trình đọc file và in đảo ngược kết quả:



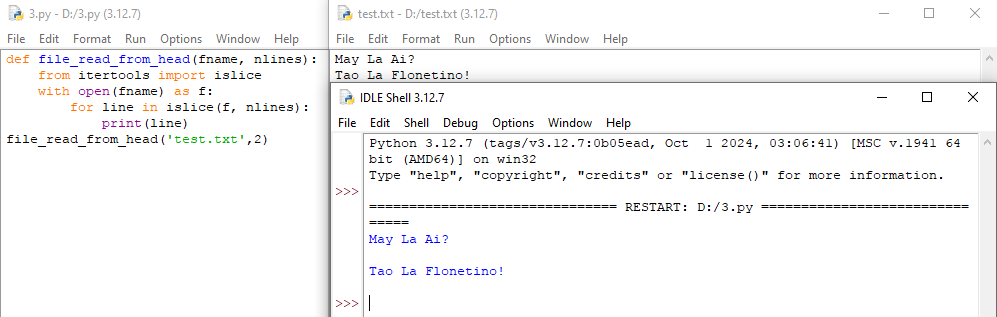
**2.**Chương trình đọc một file,tính số ký tự,số tử và số dòng của file:



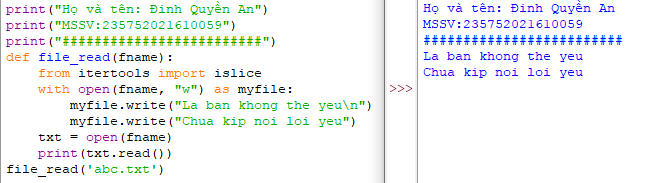
**3.**Viết chương trình Python để đọc toàn bộ tệp văn bản:



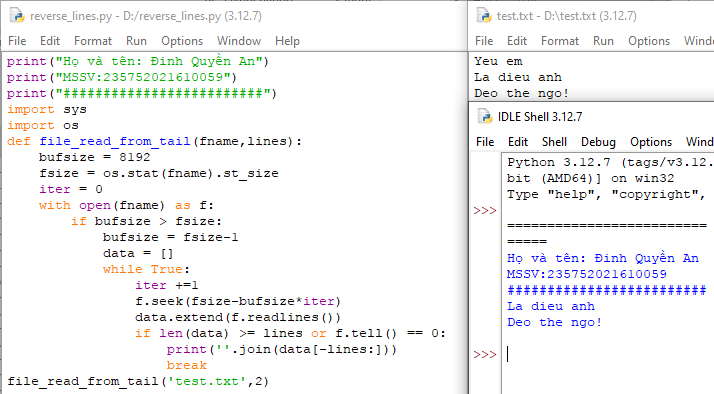
**4.**Chương trình Python để đọc n dòng đầu tiên của tệp:



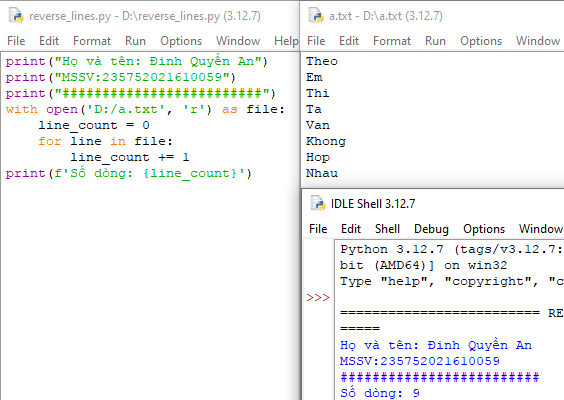
**5.**Chương trình Python để nối văn bản vào tệp và hiển thị văn bản:



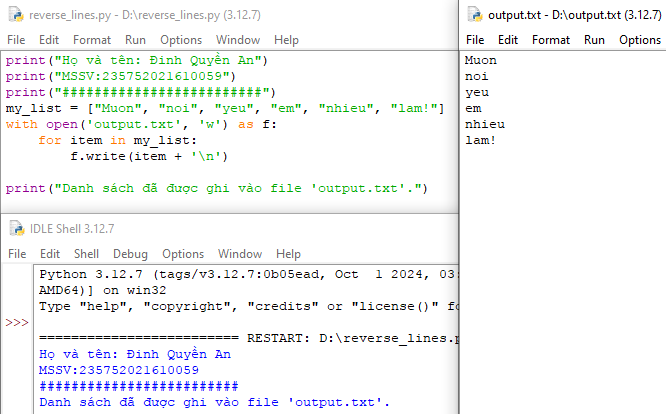
# 6.Chương trình Python để đọc n dòng cuối cùng của tệp



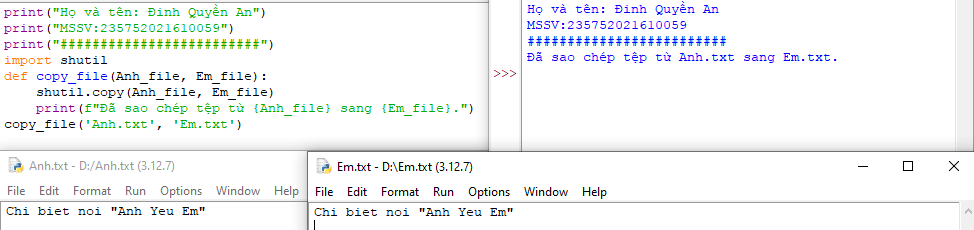
**7.**Viết chương trình Python để đếm số dòng trong tệp văn bản



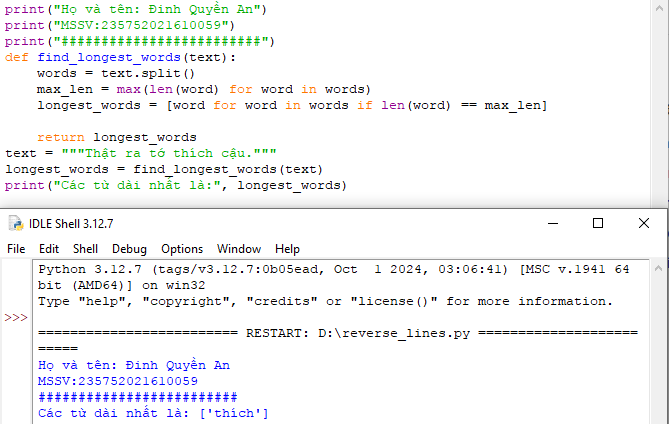
**8.**Viết chương trình Python để viết nội dung danh sách vào tệp.



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



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

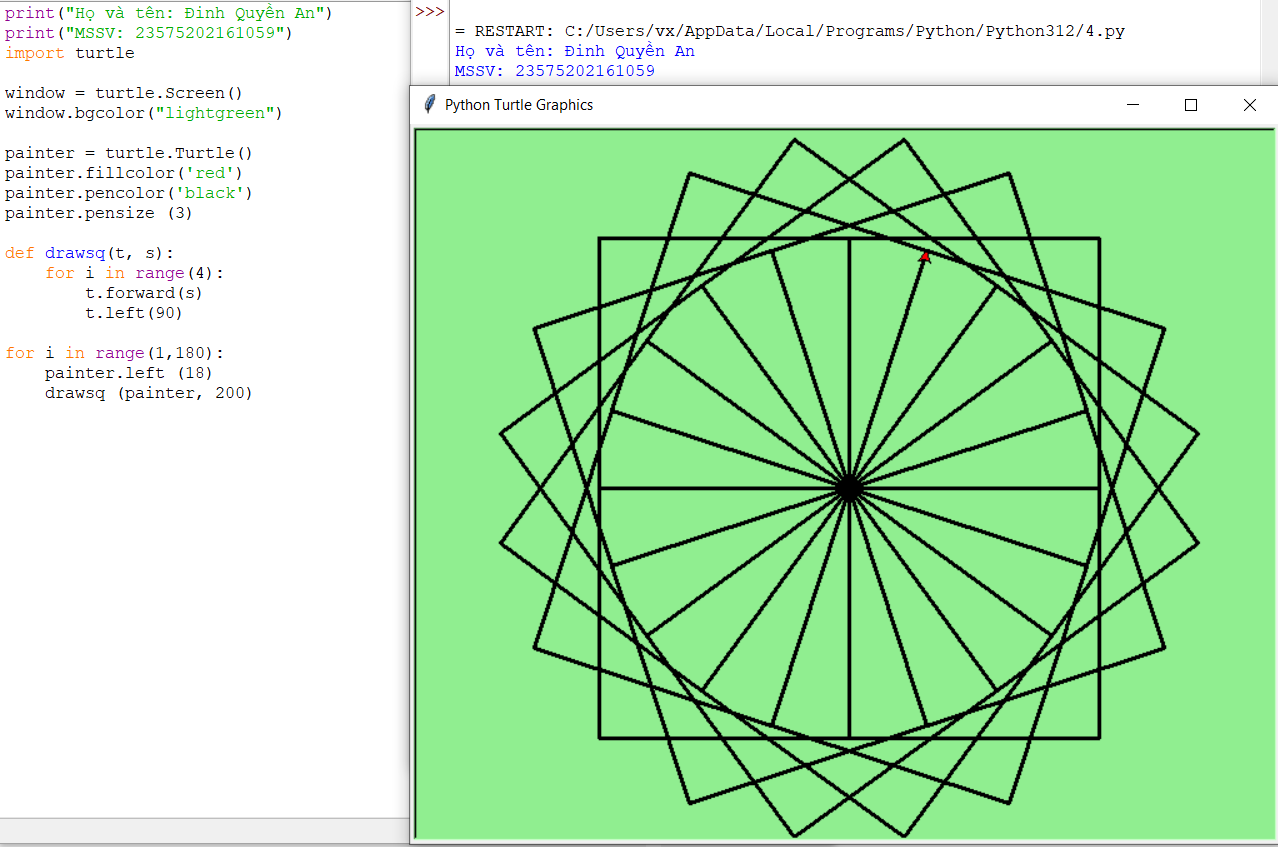


# Lập trình giao diện trong Python

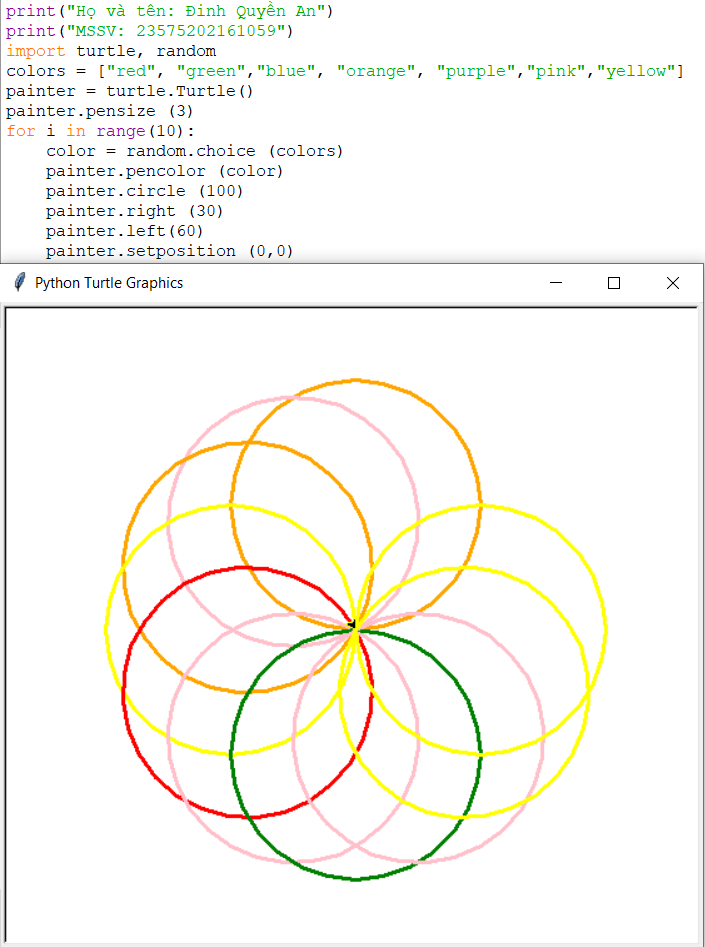
## Mục đích

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

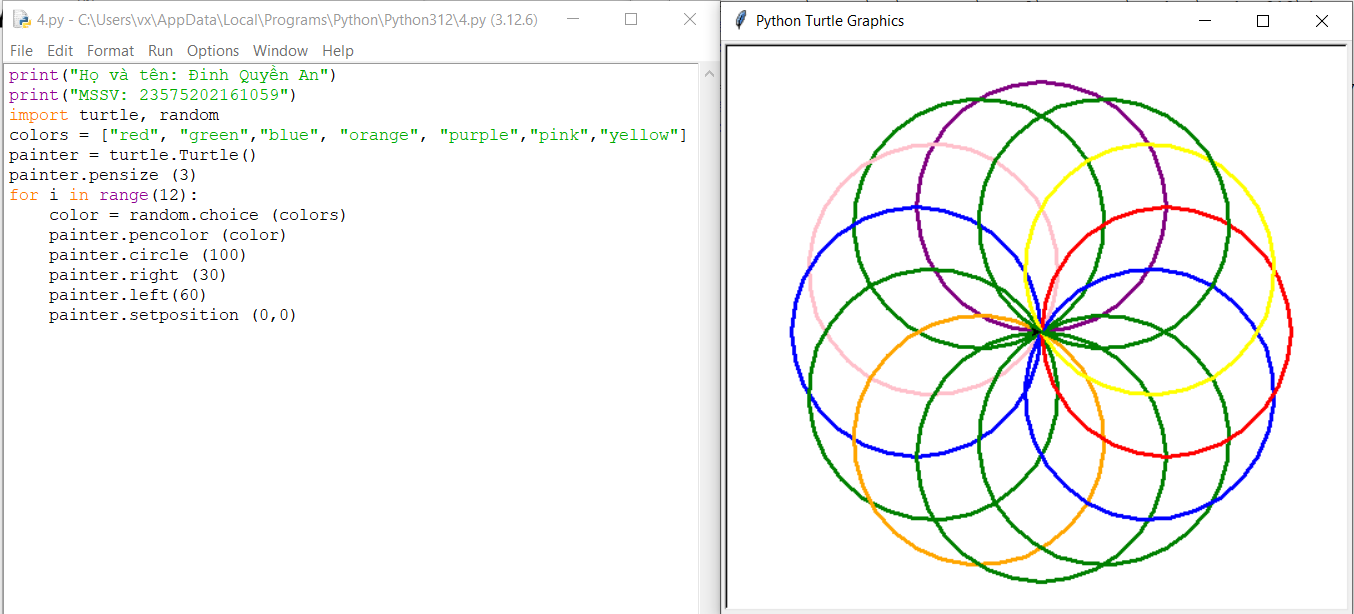
## Các bước thực hiện và kết quả 1. Viết chương trình đồ họa sử dụng thư viện turtle, kiểm tra kết quả và giải thích chương trình:



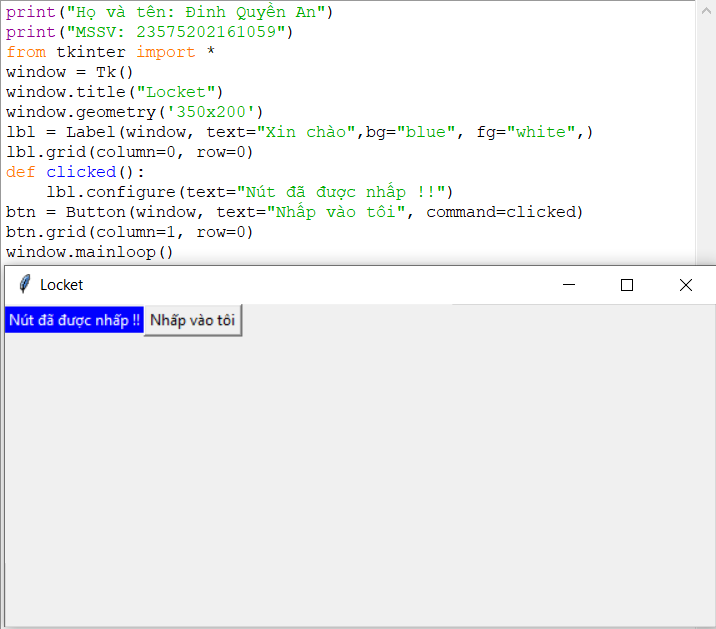
## 2. Viết chương trình đồ họa sử dụng thư viện turtle, kiểm tra kết quả và giải thích chương trình:



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

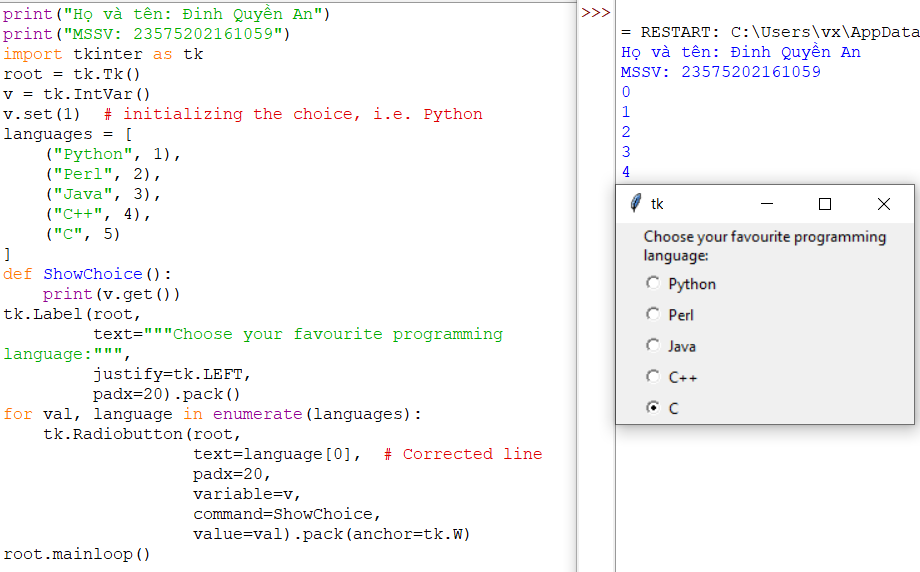


**4.** Viết chương sử dụng thư viện đồ họa tkinter thực hiện:

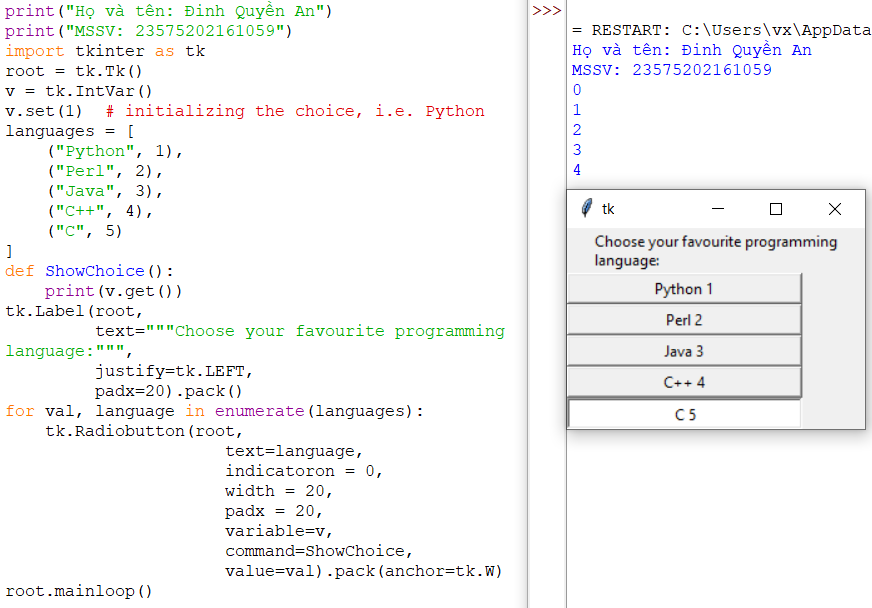


**5.** Sử dụng thư viện tkinter thực hiện:

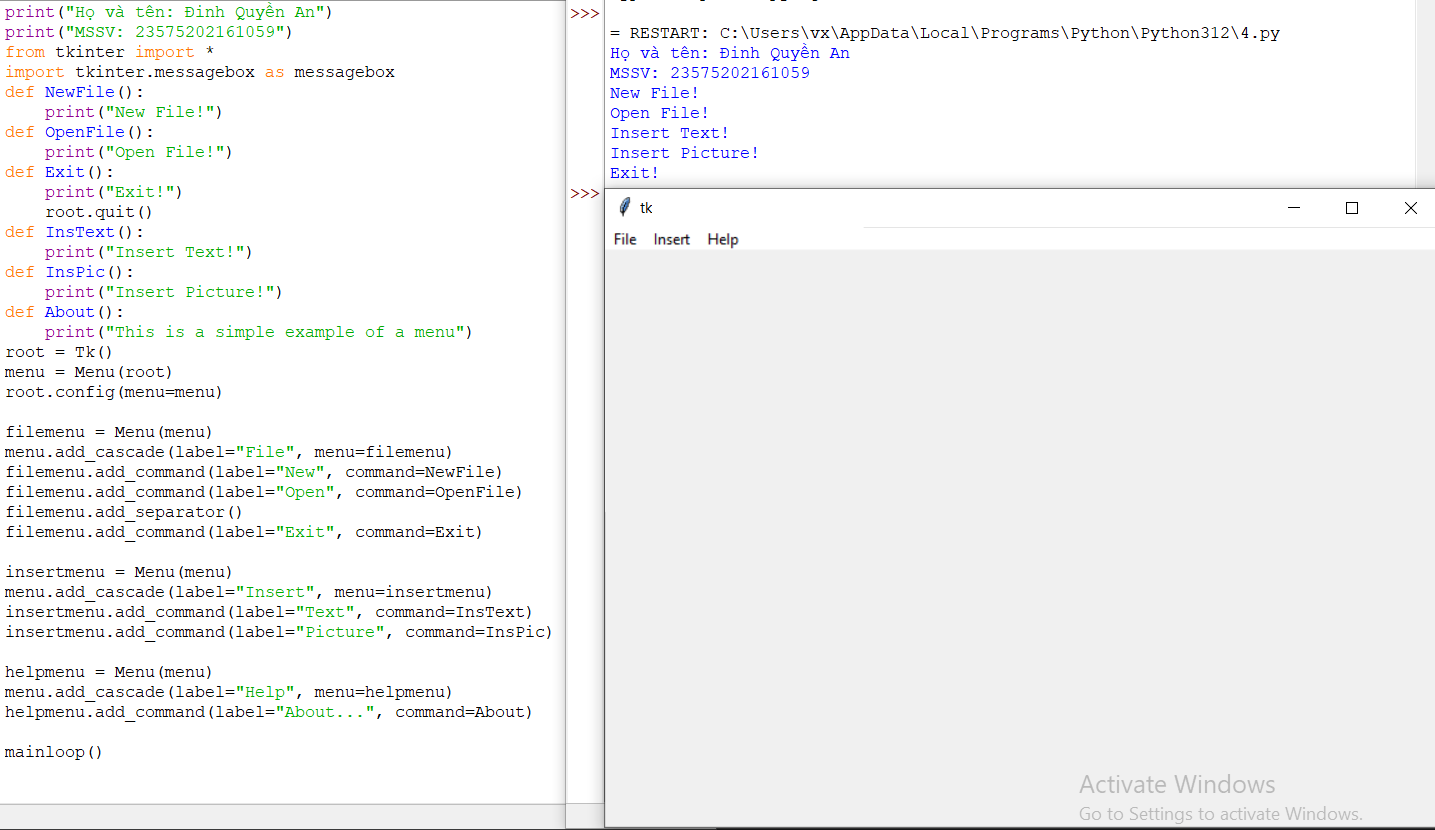
a)Xây dựng các radio button cho phép thực hiện các lựa chọn khác nhau



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

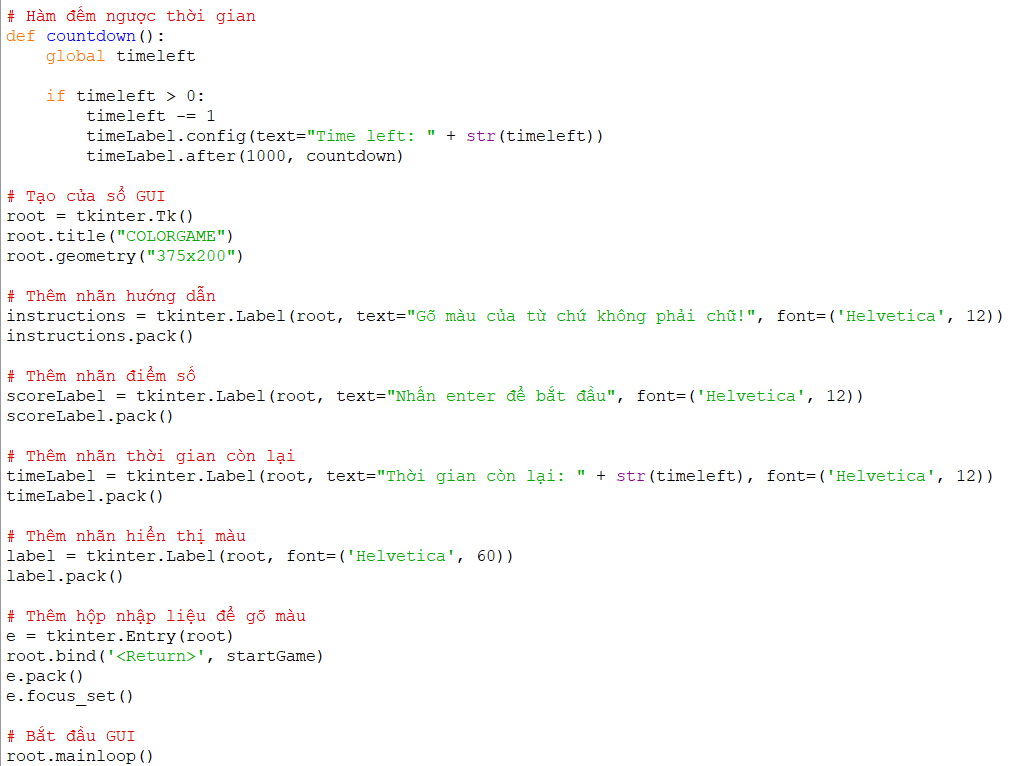


**6.**Viết chương trình thực hiện tạo menu theo các bước sau:



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

## 

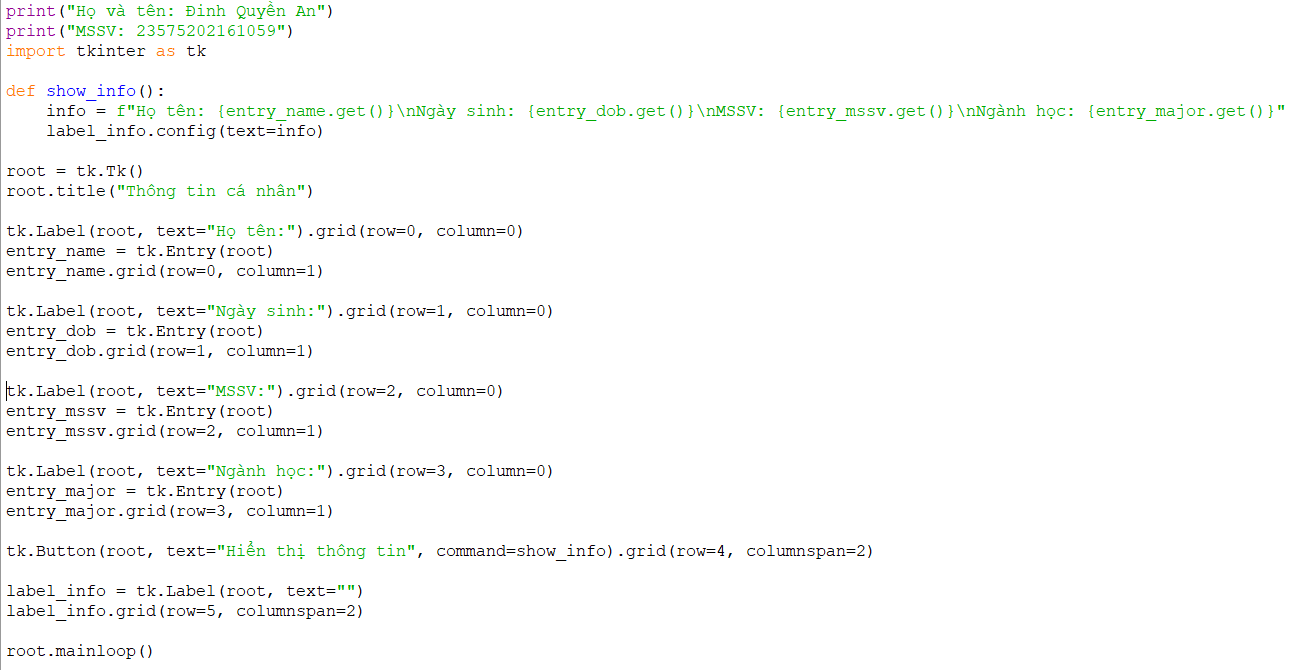


##### Kết quả:

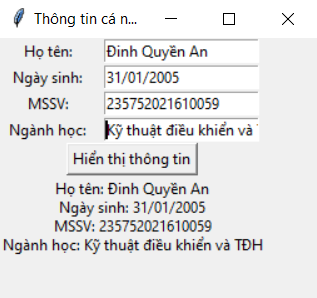
#### 

**8.**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)



##### Kết quả:



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)

**The End!**