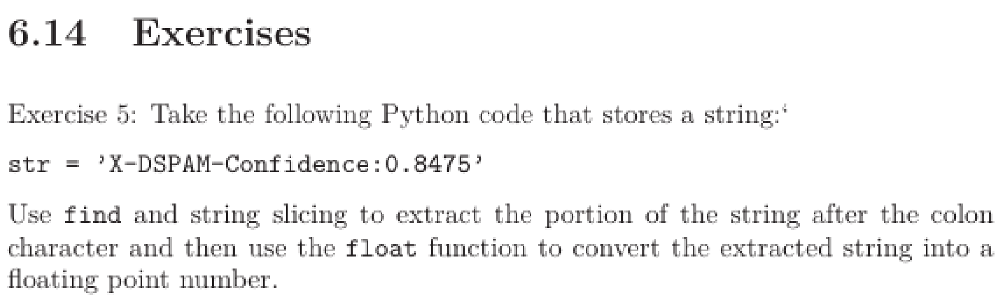
Nguyễn Tấn Trung Dũng – 060206000031

Git: [NTTD-060206000031/BT04]()

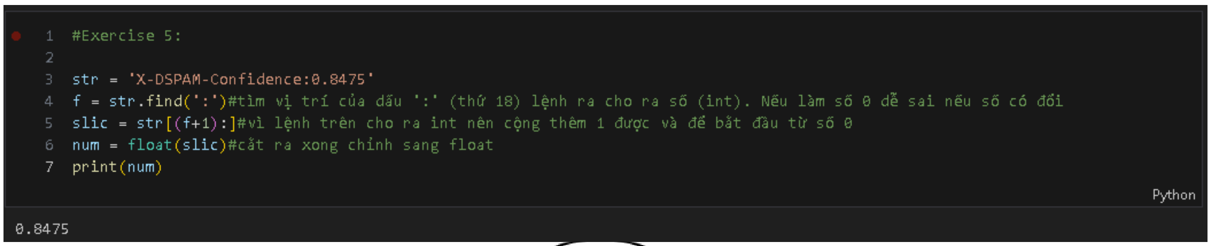


## 6.14 Exercises

Exercise 5: Take the following Python code that stores a string:‘

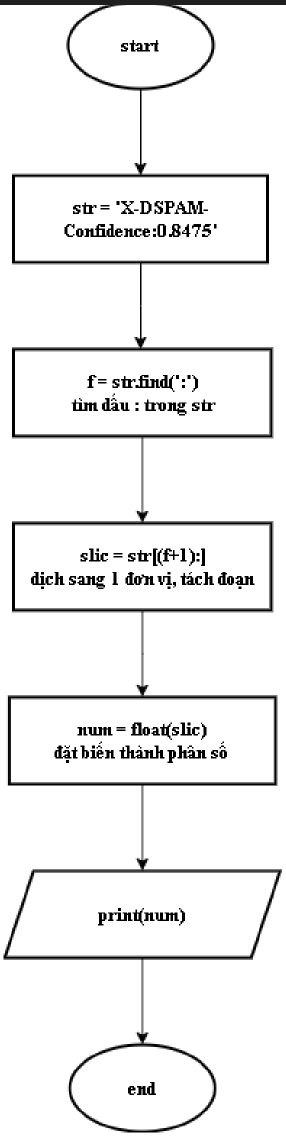
`str = ’X-DSPAM-Confidence:0.8475’`

Use `find` and string slicing to extract the portion of the string after the colon character and then use the `float` function to convert the extracted string into a floating point number.



```python
#Exercise 5:

str = 'X-DSPAM-Confidence:0.8475'
f = str.find(':')#tìm vị trí của dấu ':' (thứ 18) lệnh ra cho ra số (int). Nếu làm số 0 dễ sai nếu số có đối
slic = str[(f+1):]#vì lệnh trên cho ra int nên cộng thêm 1 được và để bắt đầu từ số 0
num = float(slic)#cắt ra xong chỉnh sang float
print(num)
```

```
0.8475
```



Exercise 6:

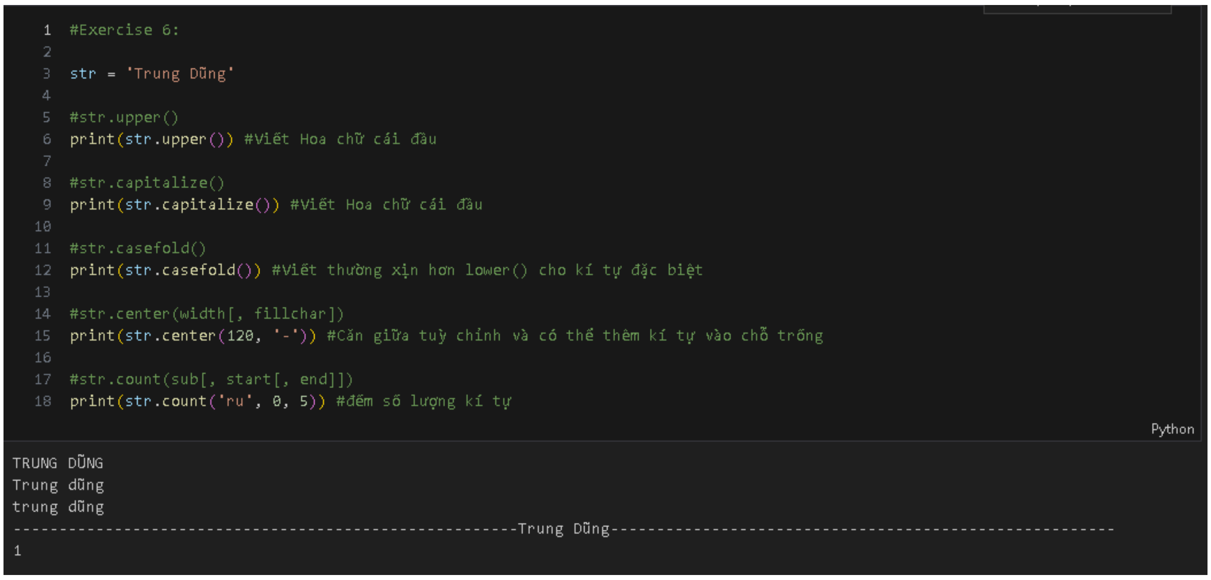
Read the documentation of the string methods at

https://docs.python.org/3.5/library/stdtypes.html#string-methods

You might want to experiment with some of them to make sure you understand how they work. `strip` and `replace` are particularly useful.

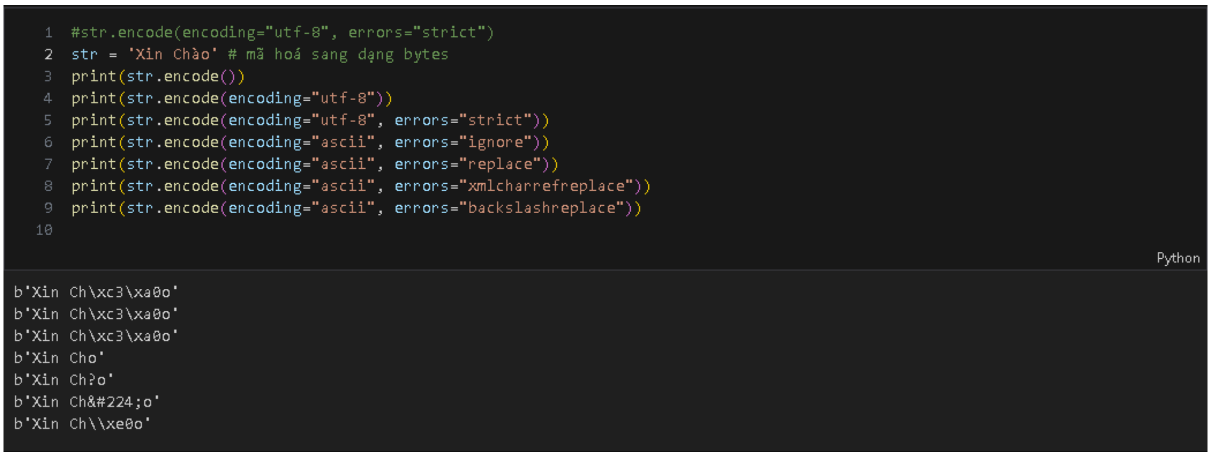
The documentation uses a syntax that might be confusing. For example, in `find(sub[, start[, end]])`, the brackets indicate optional arguments. So `sub` is required, but `start` is optional, and if you include `start`, then `end` is optional.

## Tự học



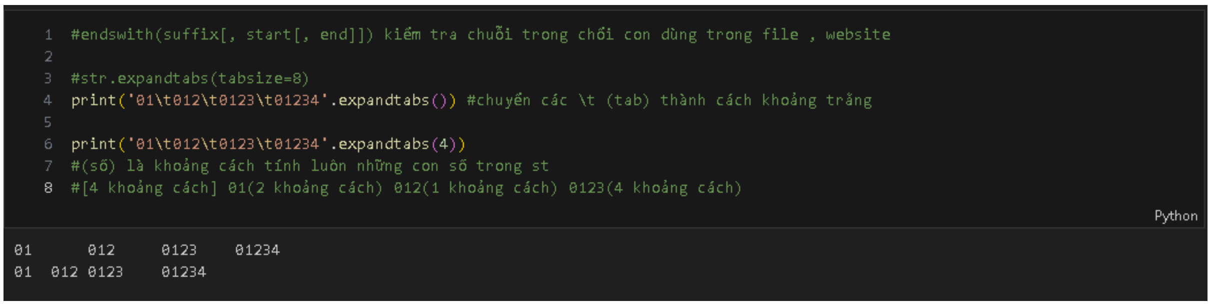
```python
#Exercise 6:

str = 'Trung Dũng'

#str.upper()
print(str.upper()) #Viết Hoa chữ cái đầu

#str.capitalize()
print(str.capitalize()) #Viết Hoa chữ cái đầu

#str.casefold()
print(str.casefold()) #Viết thường xịn hơn lower() cho kí tự đặc biệt

#str.center(width[, fillchar])
print(str.center(120, '-')) #Căn giữa tuỳ chỉnh và có thể thêm kí tự vào chỗ trống

#str.count(sub[, start[, end]])
print(str.count('ru', 0, 5)) #đếm số lượng kí tự
```

```
TRUNG DŨNG
Trung dũng
trung dũng
-------------------------------------------------------Trung Dũng-------------------------------------------------------
1
```



```python
#str.encode(encoding="utf-8", errors="strict")
str = 'Xin Chào' # mã hoá sang dạng bytes
print(str.encode())
print(str.encode(encoding="utf-8"))
print(str.encode(encoding="utf-8", errors="strict"))
print(str.encode(encoding="ascii", errors="ignore"))
print(str.encode(encoding="ascii", errors="replace"))
print(str.encode(encoding="ascii", errors="xmlcharrefreplace"))
print(str.encode(encoding="ascii", errors="backslashreplace"))
```

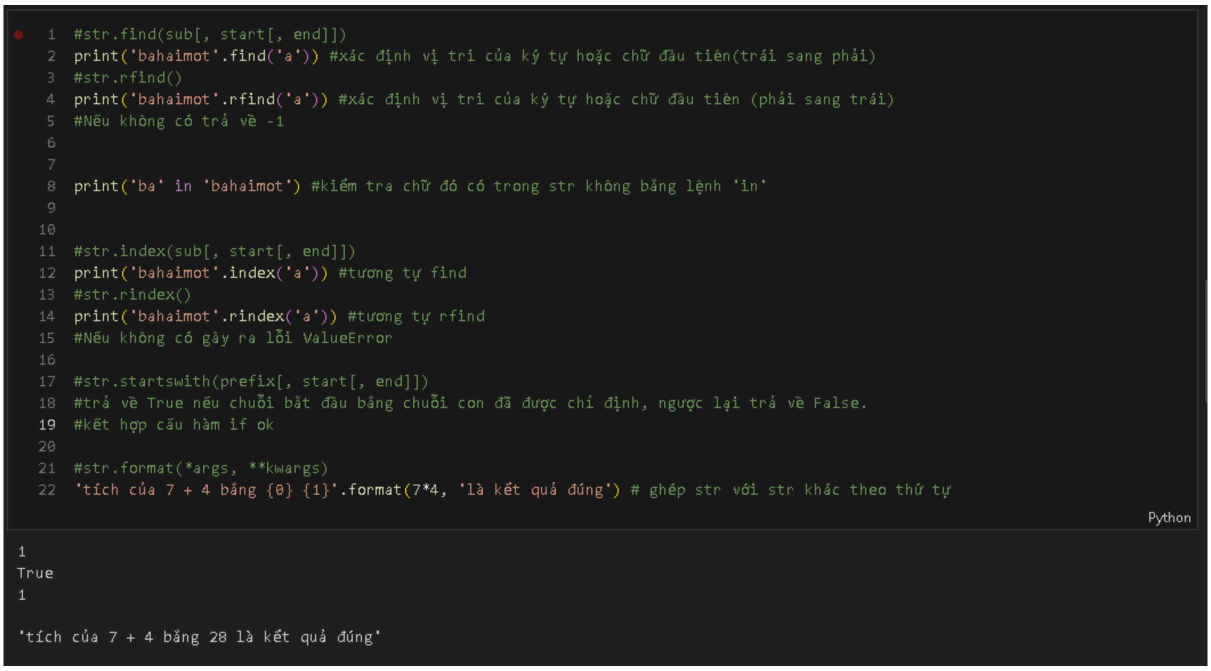
```
b'Xin Ch\xc3\xa0o'
b'Xin Ch\xc3\xa0o'
b'Xin Ch\xc3\xa0o'
b'Xin Cho'
b'Xin Ch?o'
b'Xin Ch&#224;o'
b'Xin Ch\\xe0o'
```



```python
#endswith(suffix[, start[, end]]) kiểm tra chuỗi trong chối con dùng trong file , website

#str.expandtabs(tabsize=8)
print('01\t012\t0123\t01234'.expandtabs()) #chuyển các \t (tab) thành cách khoảng trắng

print('01\t012\t0123\t01234'.expandtabs(4))
#(số) là khoảng cách tính luôn những con số trong st
#[4 khoảng cách] 01(2 khoảng cách) 012(1 khoảng cách) 0123(4 khoảng cách)
```

Python

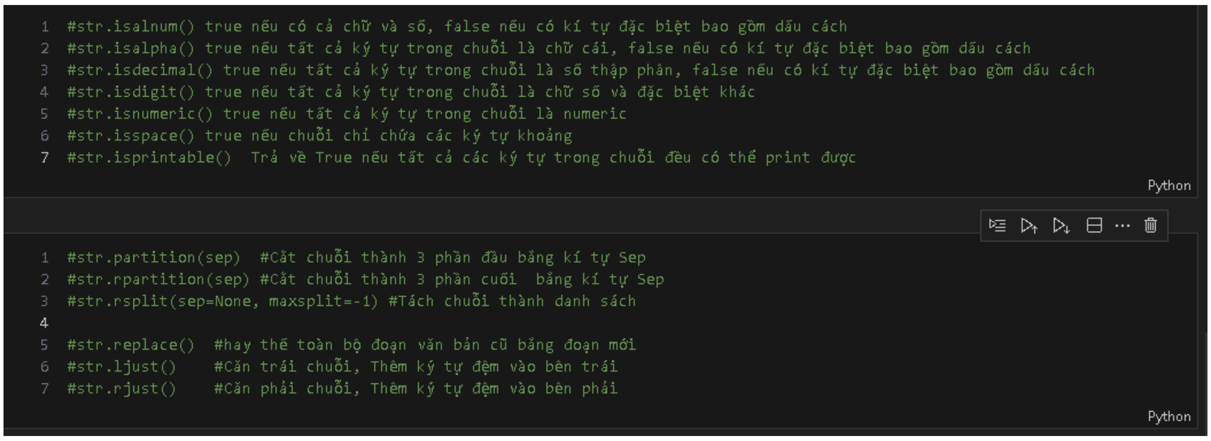
```
01      012     0123    01234
01  012 0123    01234
```



```python
#str.find(sub[, start[, end]])
print('bahaimot'.find('a')) #xác định vị tri của ký tự hoặc chữ đầu tiên(trái sang phải)
#str.rfind()
print('bahaimot'.rfind('a')) #xác định vị tri của ký tự hoặc chữ đầu tiên (phải sang trái)
#Nếu không có trả về -1


print('ba' in 'bahaimot') #kiểm tra chữ đó có trong str không bằng lệnh 'in'


#str.index(sub[, start[, end]])
print('bahaimot'.index('a')) #tương tự find
#str.rindex()
print('bahaimot'.rindex('a')) #tương tự rfind
#Nếu không có gây ra lỗi ValueError

#str.startswith(prefix[, start[, end]])
#trả về True nếu chuỗi bắt đầu bằng chuỗi con đã được chỉ định, ngược lại trả về False.
#kết hợp cấu hàm if ok

#str.format(*args, **kwargs)
'tích của 7 + 4 bằng {0} {1}'.format(7*4, 'là kết quả đúng') # ghép str với str khác theo thứ tự
```

Python

```
1
True
1

'tích của 7 + 4 bằng 28 là kết quả đúng'
```

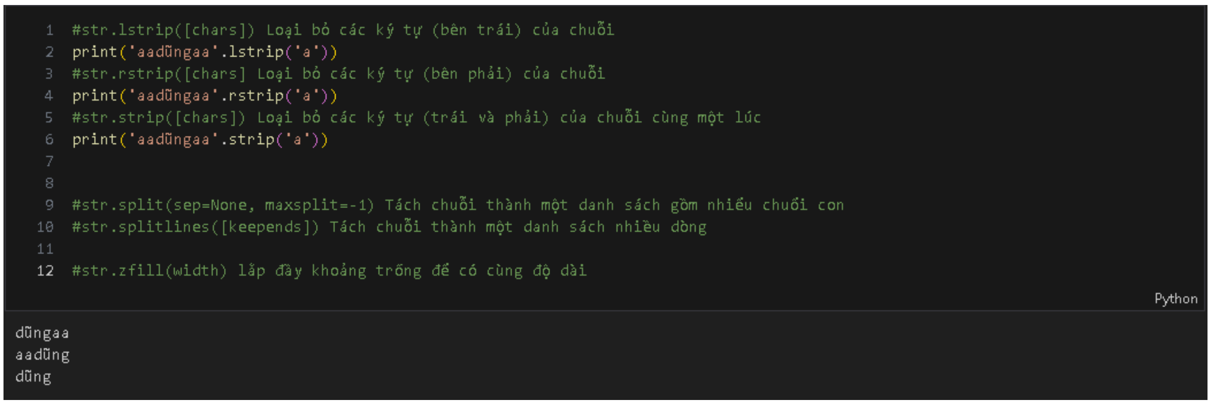


```python
#str.isalnum() true nếu có cả chữ và số, false nếu có kí tự đặc biệt bao gồm dấu cách
#str.isalpha() true nếu tất cả ký tự trong chuỗi là chữ cái, false nếu có kí tự đặc biệt bao gồm dấu cách
#str.isdecimal() true nếu tất cả ký tự trong chuỗi là số thập phân, false nếu có kí tự đặc biệt bao gồm dấu cách
#str.isdigit() true nếu tất cả ký tự trong chuỗi là chữ số và đặc biệt khác
#str.isnumeric() true nếu tất cả ký tự trong chuỗi là numeric
#str.isspace() true nếu chuỗi chỉ chứa các ký tự khoảng
#str.isprintable()  Trả về True nếu tất cả các ký tự trong chuỗi đều có thể print được
```

Python

```python
#str.partition(sep)  #Cắt chuỗi thành 3 phần đầu bằng kí tự Sep
#str.rpartition(sep) #Cắt chuỗi thành 3 phần cuối  bằng kí tự Sep
#str.rsplit(sep=None, maxsplit=-1) #Tách chuỗi thành danh sách

#str.replace()  #hay thế toàn bộ đoạn văn bản cũ bằng đoạn mới
#str.ljust()    #Căn trái chuỗi, Thêm ký tự đệm vào bên trái
#str.rjust()    #Căn phải chuỗi, Thêm ký tự đệm vào bên phải
```

Python



```python
#str.lstrip([chars]) Loại bỏ các ký tự (bên trái) của chuỗi
print('aadũngaa'.lstrip('a'))
#str.rstrip([chars]) Loại bỏ các ký tự (bên phải) của chuỗi
print('aadũngaa'.rstrip('a'))
#str.strip([chars]) Loại bỏ các ký tự (trái và phải) của chuỗi cùng một lúc
print('aadũngaa'.strip('a'))


#str.split(sep=None, maxsplit=-1) Tách chuỗi thành một danh sách gồm nhiều chuỗi con
#str.splitlines([keepends]) Tách chuỗi thành một danh sách nhiều dòng

#str.zfill(width) lấp đầy khoảng trống để có cùng độ dài
```

Python

```
dũngaa
aadũng
dũng
```

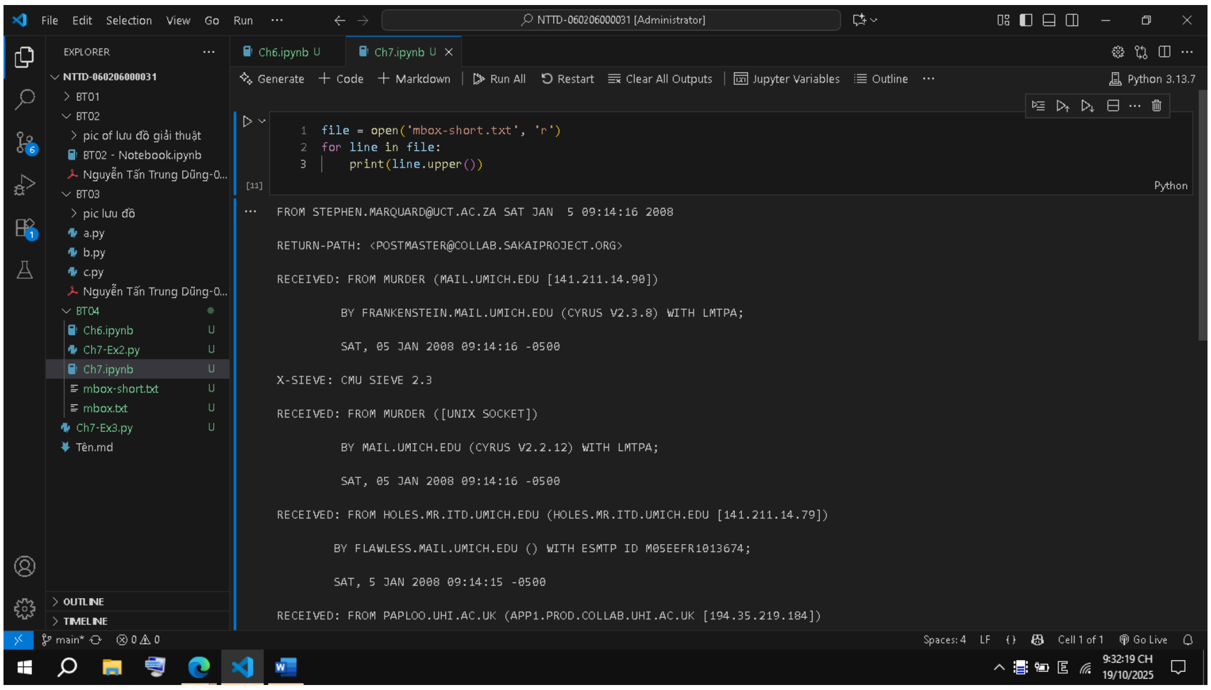
## 7.11 Exercises

Exercise 1: Write a program to read through a file and print the contents of the file (line by line) all in upper case. Executing the program will look as follows:

```
python shout.py
Enter a file name: mbox-short.txt
FROM STEPHEN.MARQUARD@UCT.AC.ZA SAT JAN  5 09:14:16 2008
RETURN-PATH: <POSTMASTER@COLLAB.SAKAIPROJECT.ORG>
RECEIVED: FROM MURDER (MAIL.UMICH.EDU [141.211.14.90])
     BY FRANKENSTEIN.MAIL.UMICH.EDU (CYRUS V2.3.8) WITH LMTPA;
     SAT, 05 JAN 2008 09:14:16 -0500
```

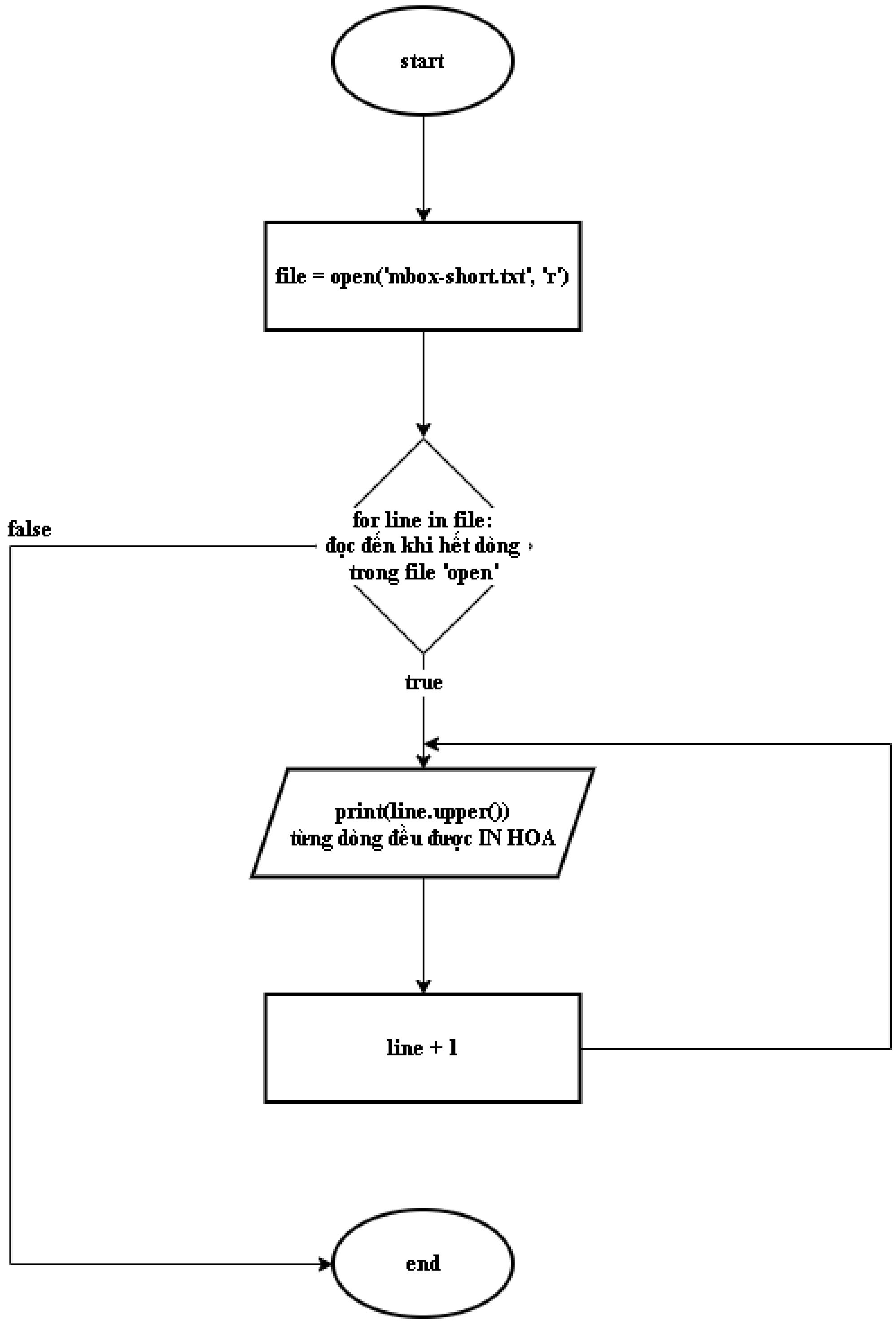
You can download the file from

www.py4e.com/code3/mbox-short.txt



```python
file = open('mbox-short.txt', 'r')
for line in file:
    print(line.upper())
```

```
FROM STEPHEN.MARQUARD@UCT.AC.ZA SAT JAN  5 09:14:16 2008

RETURN-PATH: <POSTMASTER@COLLAB.SAKAIPROJECT.ORG>

RECEIVED: FROM MURDER (MAIL.UMICH.EDU [141.211.14.90])

        BY FRANKENSTEIN.MAIL.UMICH.EDU (CYRUS V2.3.8) WITH LMTPA;

        SAT, 05 JAN 2008 09:14:16 -0500

X-SIEVE: CMU SIEVE 2.3

RECEIVED: FROM MURDER ([UNIX SOCKET])

        BY MAIL.UMICH.EDU (CYRUS V2.2.12) WITH LMTPA;

        SAT, 05 JAN 2008 09:14:16 -0500

RECEIVED: FROM HOLES.MR.ITD.UMICH.EDU (HOLES.MR.ITD.UMICH.EDU [141.211.14.79])

       BY FLAWLESS.MAIL.UMICH.EDU () WITH ESMTP ID M05EEFR1013674;

       SAT, 5 JAN 2008 09:14:15 -0500

RECEIVED: FROM PAPLOO.UHI.AC.UK (APP1.PROD.COLLAB.UHI.AC.UK [194.35.219.184])
```


start
file = open('mbox-short.txt', 'r')
for line in file:
đọc đến khi hết dòng
trong file 'open'
false
true
print(line.upper())
từng dòng đều được IN HOA
line + 1
end

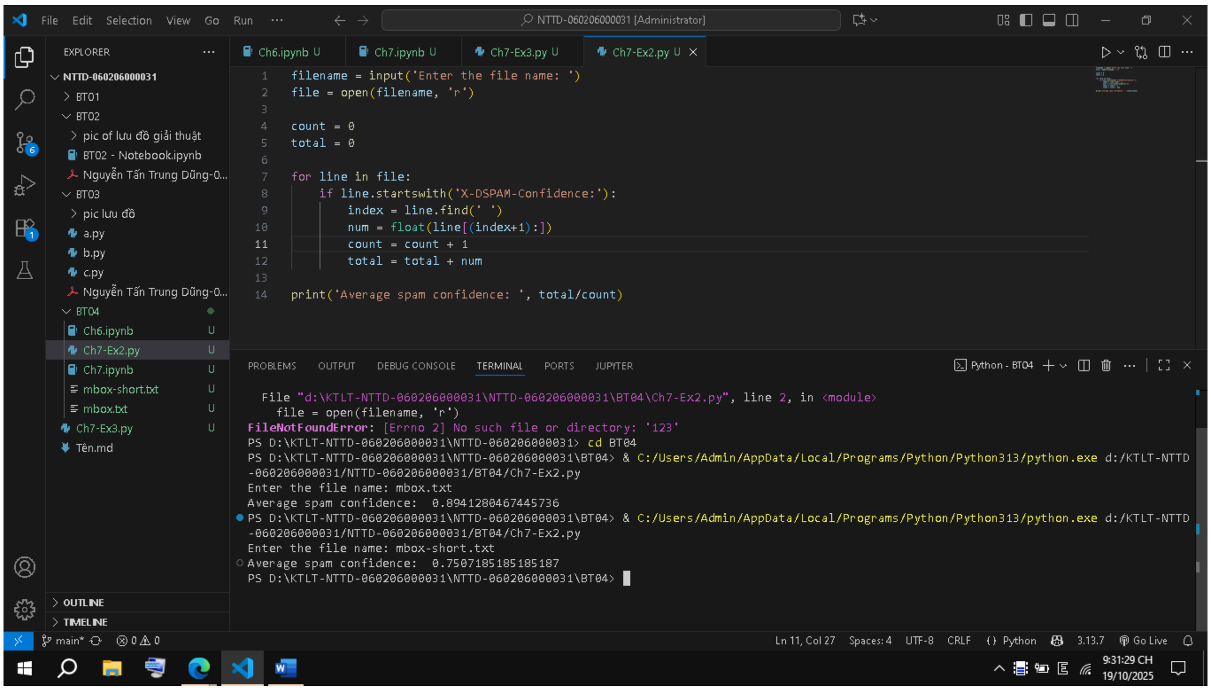
Exercise 2: Write a program to prompt for a file name, and then read through the file and look for lines of the form:

```
X-DSPAM-Confidence:0.8475
```

When you encounter a line that starts with "X-DSPAM-Confidence:" pull apart the line to extract the floating-point number on the line. Count these lines and then compute the total of the spam confidence values from these lines. When you reach the end of the file, print out the average spam confidence.

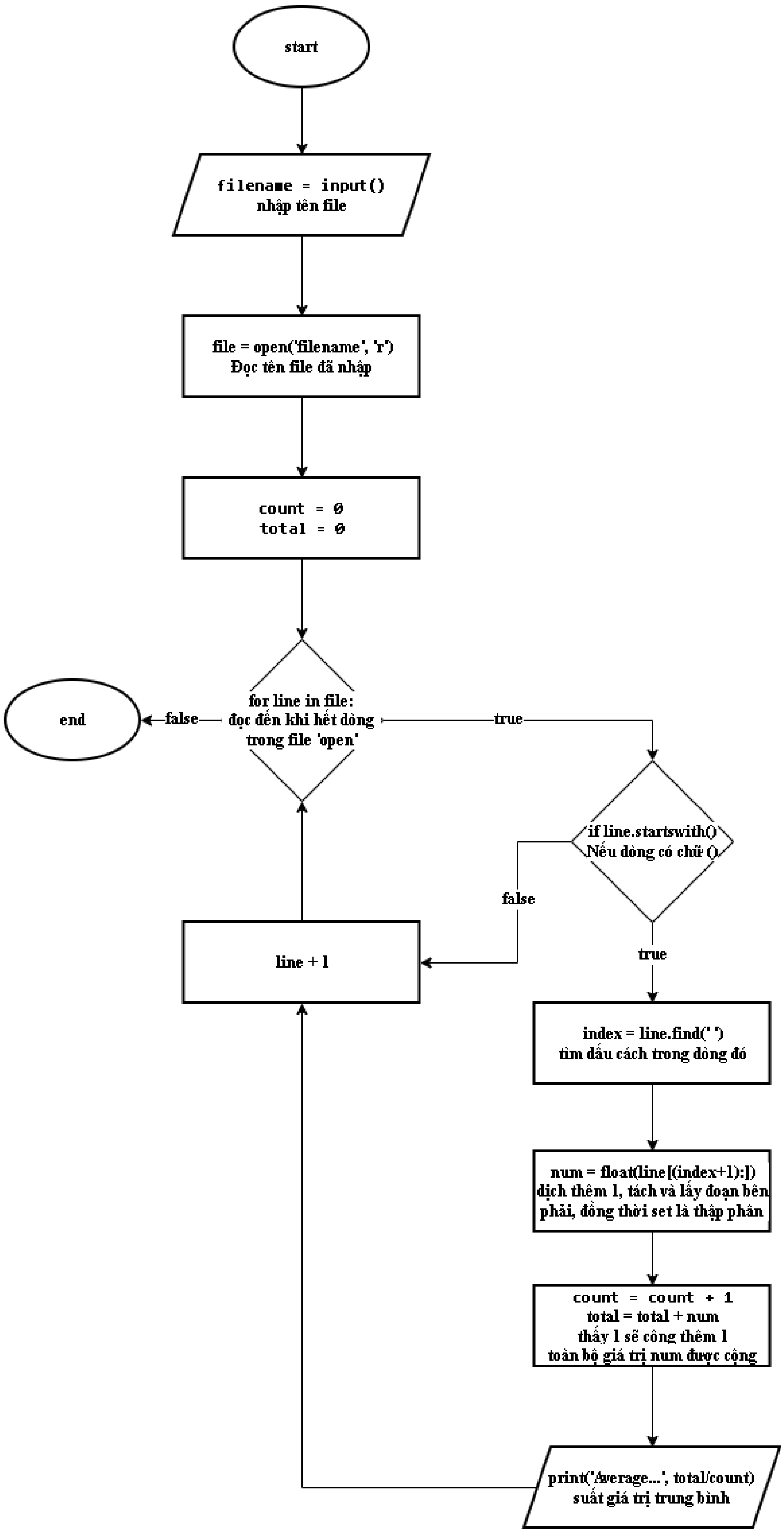
```
Enter the file name: mbox.txt
Average spam confidence: 0.894128046745

Enter the file name: mbox-short.txt
Average spam confidence: 0.750718518519
```

Test your file on the `mbox.txt` and `mbox-short.txt` files.



```python
filename = input('Enter the file name: ')
file = open(filename, 'r')

count = 0
total = 0

for line in file:
    if line.startswith('X-DSPAM-Confidence:'):
        index = line.find(' ')
        num = float(line[(index+1):])
        count = count + 1
        total = total + num

print('Average spam confidence: ', total/count)
```

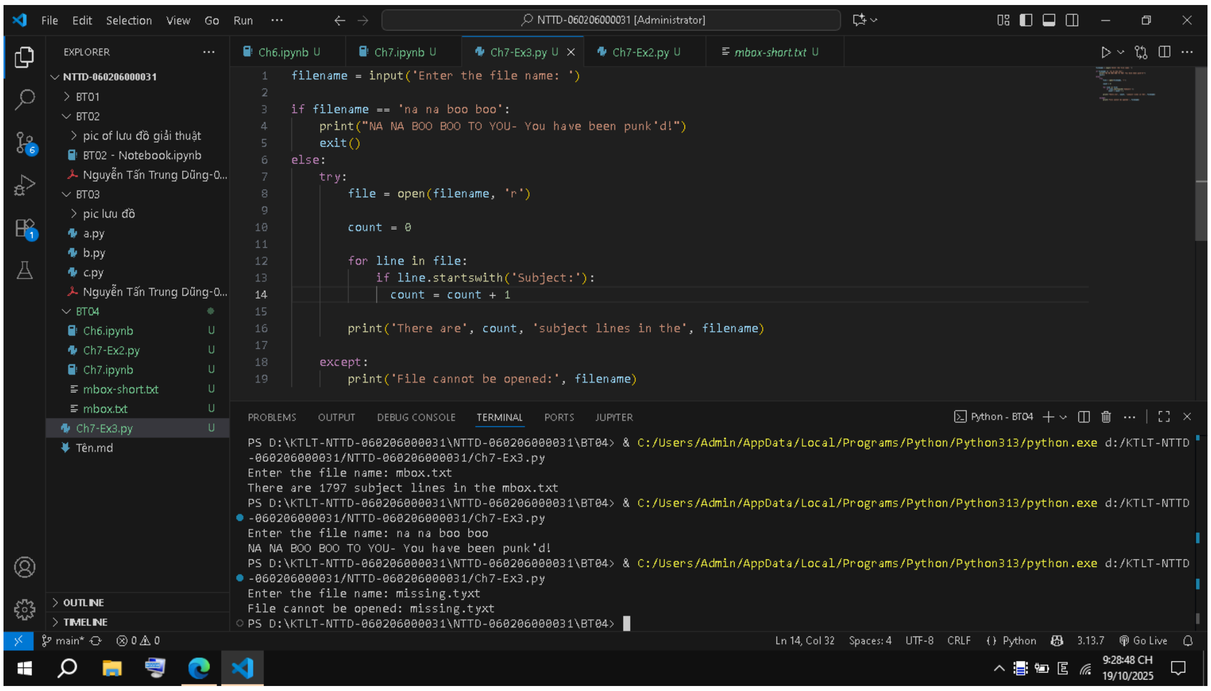
```
  File "d:\KTLT-NTTD-060206000031\NTTD-060206000031\BT04\Ch7-Ex2.py", line 2, in <module>
    file = open(filename, 'r')
FileNotFoundError: [Errno 2] No such file or directory: '123'
PS D:\KTLT-NTTD-060206000031\NTTD-060206000031> cd BT04
PS D:\KTLT-NTTD-060206000031\NTTD-060206000031\BT04> & C:/Users/Admin/AppData/Local/Programs/Python/Python313/python.exe d:/KTLT-NTTD
-060206000031/NTTD-060206000031/BT04/Ch7-Ex2.py
Enter the file name: mbox.txt
Average spam confidence:  0.8941280467445736
PS D:\KTLT-NTTD-060206000031\NTTD-060206000031\BT04> & C:/Users/Admin/AppData/Local/Programs/Python/Python313/python.exe d:/KTLT-NTTD
-060206000031/NTTD-060206000031/BT04/Ch7-Ex2.py
Enter the file name: mbox-short.txt
Average spam confidence:  0.7507185185185187
PS D:\KTLT-NTTD-060206000031\NTTD-060206000031\BT04>
```


start
filename = input()
nhập tên file
file = open('filename', 'r')
Đọc tên file đã nhập
count = 0
total = 0
for line in file:
đọc đến khi hết dòng
trong file 'open'
false
end
true
if line.startswith()
Nếu dòng có chữ ()
false
true
line + 1
index = line.find(' ')
tìm dấu cách trong dòng đó
num = float(line[(index+1):])
dịch thêm 1, tách và lấy đoạn bên
phải, đồng thời set là thập phân
count = count + 1
total = total + num
thấy 1 sẽ cộng thêm 1
toàn bộ giá trị num được cộng
print('Average...', total/count)
suất giá trị trung bình

Exercise 3: Sometimes when programmers get bored or want to have a bit of fun, they add a harmless *Easter Egg* to their program Modify the program that prompts the user for the file name so that it prints a funny message when the user types in the exact file name "na na boo boo". The program should behave normally for all other files which exist and don't exist. Here is a sample execution of the program:

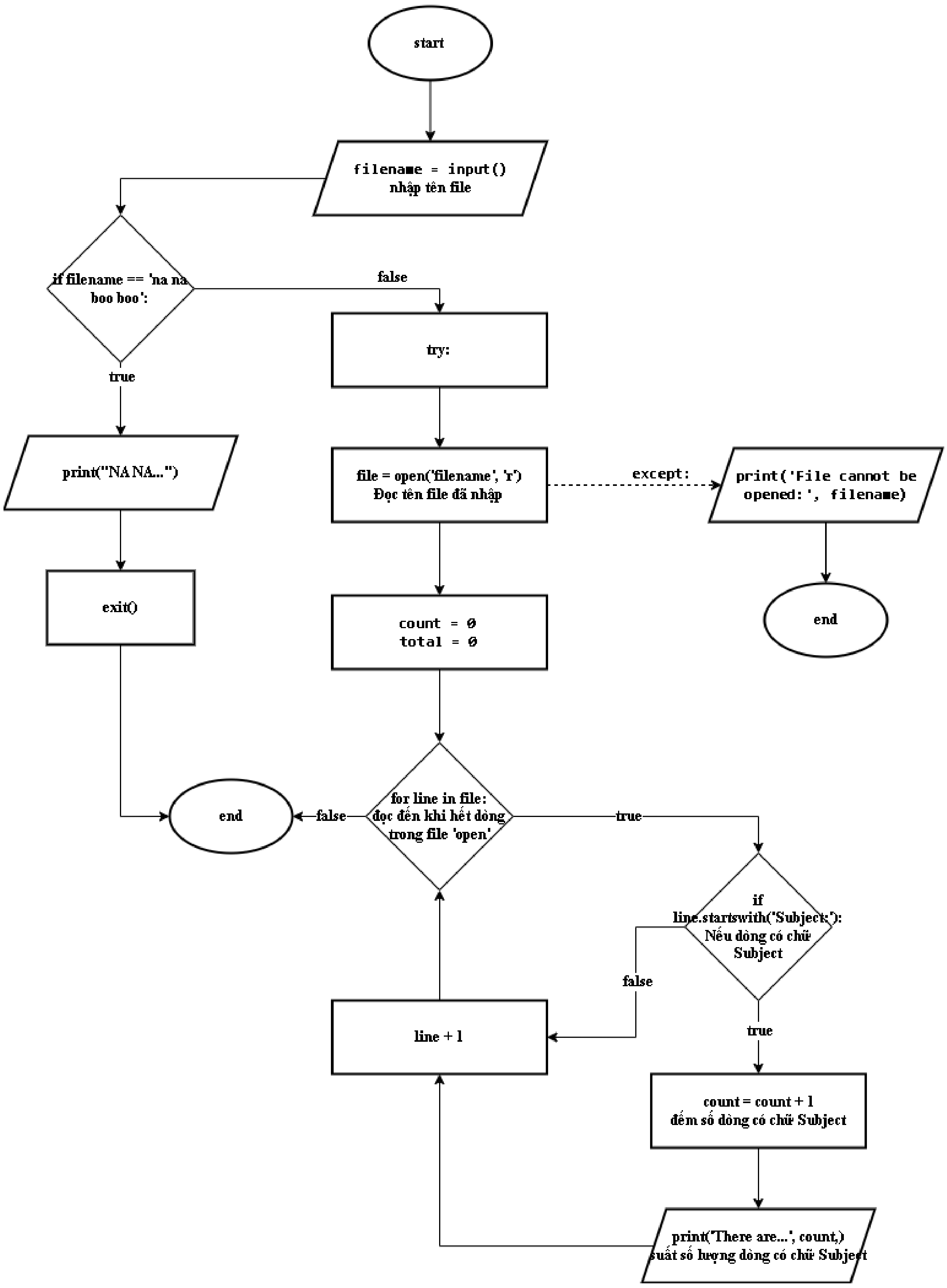
```
python egg.py
Enter the file name: mbox.txt
There were 1797 subject lines in mbox.txt

python egg.py
Enter the file name: missing.tyxt
File cannot be opened: missing.tyxt

python egg.py
Enter the file name: na na boo boo
NA NA BOO BOO TO YOU - You have been punk'd!
```

We are not encouraging you to put Easter Eggs in your programs; this is just an exercise.



```python
filename = input('Enter the file name: ')

if filename == 'na na boo boo':
    print("NA NA BOO BOO TO YOU- You have been punk'd!")
    exit()
else:
    try:
        file = open(filename, 'r')

        count = 0

        for line in file:
            if line.startswith('Subject:'):
                count = count + 1

        print('There are', count, 'subject lines in the', filename)

    except:
        print('File cannot be opened:', filename)
```

```
PS D:\KTLT-NTTD-060206000031\NTTD-060206000031\BT04> & C:/Users/Admin/AppData/Local/Programs/Python/Python313/python.exe d:/KTLT-NTTD
-060206000031/NTTD-060206000031/Ch7-Ex3.py
Enter the file name: mbox.txt
There are 1797 subject lines in the mbox.txt
PS D:\KTLT-NTTD-060206000031\NTTD-060206000031\BT04> & C:/Users/Admin/AppData/Local/Programs/Python/Python313/python.exe d:/KTLT-NTTD
-060206000031/NTTD-060206000031/Ch7-Ex3.py
Enter the file name: na na boo boo
NA NA BOO BOO TO YOU- You have been punk'd!
PS D:\KTLT-NTTD-060206000031\NTTD-060206000031\BT04> & C:/Users/Admin/AppData/Local/Programs/Python/Python313/python.exe d:/KTLT-NTTD
-060206000031/NTTD-060206000031/Ch7-Ex3.py
Enter the file name: missing.tyxt
File cannot be opened: missing.tyxt
PS D:\KTLT-NTTD-060206000031\NTTD-060206000031\BT04>
```



start
filename = input()
nhập tên file
if filename == 'na na boo boo':
false
true
try:
print("NA NA...")
file = open('filename', 'r')
Đọc tên file đã nhập
except:
print('File cannot be opened:', filename)
end
exit()
count = 0
total = 0
end
false
for line in file:
đọc đến khi hết dòng trong file 'open'
true
if line.startswith('Subject:'):
Nếu dòng có chữ Subject
false
line + 1
true
count = count + 1
đếm số dòng có chữ Subject
print('There are...', count,)
xuất số lượng dòng có chữ Subject