



BÀI TẬP CHUYÊN ĐỀ

FUNDAMENTALS OF PYTHON (PYTHON CƠ BẢN)

BÀI 1: Tổng quan Lập trình Python



Mục tiêu chính:

- *Biết cách cài đặt Python và các công cụ lập trình IDE*
- *Làm quen với môi trường làm việc với Python*

1.1. Thực hành cài đặt môi trường Python

✓ **Hướng dẫn:**

- Môi trường Windows: <https://youtu.be/3NMb4IH6LIk>
- Môi trường MacOS: <https://youtu.be/rm7Ikl2BQjM>

1.2. Thực hành cài đặt các IDE (**I**ntegrated **D**evelopment **E**nvironment)

✓ **Yêu cầu:** Học viên có thể làm việc với các công cụ sau:

- Visual Studio Code
- Pycharm
- Jupyter Notebook
- Google Colaboratory
- ...

✓ **Tham khảo:**

- Môi trường Windows: <https://youtu.be/fI7yL22OU9o>
- Môi trường MacOS: <https://youtu.be/eNuDfQiatro>

BÀI 2: Biến và các kiểu dữ liệu cơ sở



Mục tiêu chính: Cung cấp cho học viên kiến thức và kỹ năng làm việc với:

- Định danh (Identifier)
- Từ khóa (Keyword)
- Biến (Variable)
- Các kiểu dữ liệu: Integer, Boolean, Float, String

2.1. Đỗ vui

- ✓ **Yêu cầu 1:** Thêm dấu ngoặc đơn vào biểu thức A để nó có giá trị là 1.

$$A = (5 - 3) // 2$$

- ✓ **Yêu cầu 2:** Thêm dấu ngoặc đơn vào biểu thức sau để nó có giá trị là 0.

$$B = 8 - 3 * 2 - (1 + 1)$$

2.2. Tính tiền hàng

- ✓ **Yêu cầu:** Xây dựng chương trình tính tiền hàng theo công thức:

$$\text{Thành tiền} = \text{Số lượng} * \text{Đơn giá}$$

- ✓ **Thuật giải:**

❖ Nhập:

- Số lượng
- Đơn giá

❖ Xuất:

- Thành tiền

❖ Xử lý:

- Thành tiền = Số lượng * Đơn giá

- ✓ **Minh họa:**

Số lượng: 15

Đơn giá: 50000

Thành tiền = 15 * 50,000 = 750,000

2.3. Đổi nhiệt độ từ độ C sang độ F

- ✓ **Yêu cầu:** Viết chương trình đổi nhiệt độ từ độ C sang độ F theo công thức:

$$\text{Độ F} = 9/5 * \text{Độ C} + 32$$

- ✓ **Thuật giải:**

- ❖ Nhập:
 - Độ C
- ❖ Xuất:
 - Độ F
- ❖ Xử lý:
 - $\text{Độ F} = 9/5 * \text{Độ C} + 32$

✓ **Minh họa:**

Nhập độ C: 27
 27.00 độ C = 80.60 độ F

2.4. Xử lý chuỗi

- ✓ **Yêu cầu:** Thực hiện các yêu cầu sau (tham khảo thêm hình minh họa):
- Nhập vào 1 chuỗi bất kỳ, xuất ra chiều dài của chuỗi vừa nhập.
 - Nhập vào vị trí index bất kỳ (nằm trong phạm vi của chuỗi vừa nhập), xuất ra ký tự tương ứng với vị trí index đó.
 - Nhập vào khoảng (range) index để lấy ra 5 ký tự trong chuỗi.

✓ **Thuật giải:**

- ❖ Nhập:
 - Chuỗi
 - Index
 - Khoảng (range) index
- ❖ Xuất:
 - Chiều dài chuỗi đã nhập
 - 1 ký tự theo index đã nhập
 - 5 ký tự theo range index đã nhập

✓ **Minh họa:**

Nhập chuỗi: *python for everyone*
 Chuỗi "python for everyone" có chiều dài: 19

Nhập vào index (số nguyên từ 0 đến 18): 7
 Ký tự tại vị trí index 7 là: f

Nhập range index để lấy 5 ký tự: 1,6
 5 ký tự từ vị trí index 1 đến 6 là: ython

2.5. Tính tiền lãi gửi tiết kiệm

- ✓ **Yêu cầu:** Viết chương trình tính tiền lãi và tính tổng số tiền nhận được sau khi hết thời hạn gửi tiền:

$$\text{Tiền lãi} = (\text{Số tiền gửi} * \text{Số tháng}) * (\text{Lãi suất năm} / 12)$$

$$\text{Tổng số tiền} = \text{Số tiền gửi} + \text{Tiền lãi}$$

- ✓ **Thuật giải:**

- ❖ Nhập:

- Lãi suất
- Số tiền gửi
- Số tháng gửi

- ❖ Xuất:

- Tiền lãi
- Tổng tiền nhận được

- ❖ Xử lý:

- $\text{Tiền lãi} = (\text{Số tiền gửi} * \text{Số tháng}) * (\text{Lãi suất năm} / 12)$
- $\text{Tổng số tiền} = \text{Số tiền gửi} + \text{Tiền lãi}$

- ✓ **Minh họa:**

```
Lãi xuất 1 năm (%): 7.6
Số tiền gửi: 10000000
Số tháng gửi: 6
----- Kết Quả -----
Tiền lãi = 380,000.0
Tổng tiền = 10,000,000 + 380,000.0 = 10,380,000.0
```

BÀI 3: Toán tử



Mục tiêu chính: Cung cấp cho học viên kiến thức và kỹ năng:

- Sử dụng các toán tử trong Python

3.1. Tính trung bình cộng

- ✓ **Yêu cầu:** Viết chương trình nhập vào 2 số nguyên từ bàn phím, xuất ra màn hình kết quả tổng và trung bình cộng của 2 giá trị vừa nhập.
- ✓ **Thuật giải:**
 - ❖ Nhập:
 - 2 số nguyên
 - ❖ Xuất:
 - Tổng 2 số nguyên
 - Trung bình cộng của 2 số nguyên
- ✓ **Minh họa:**

```
Nhập 2 giá trị a, b: 3,5
Tổng = 3 + 5 = 8
Trung bình cộng = (3 + 5) / 2 = 4.0
```

3.2. Tính giá trị biểu thức

- ✓ **Yêu cầu:** Xây dựng chương trình tính và in ra tổng của biểu thức S.

$$S = 1 + x + x^3/3 + x^5/5$$

- ✓ **Thuật giải:**
 - ❖ Nhập:
 - x
 - ❖ Xuất:
 - S
 - ❖ Xử lý:
 - $S = 1 + x + x^3/3 + x^5/5$

- ✓ **Minh họa:**

```
Nhập x: 6
S = 1 + x + x^3/3 + x^5/5 = 1634.2
```

3.3. Tính toán mệnh giá tiền

- ✓ **Yêu cầu:** Viết chương trình nhập vào số tiền muốn đổi, đổi ra các số tờ mệnh giá 500000, 200000, 100000, 50000 và xuất kết quả ra màn hình.

- ✓ **Thuật giải:**

- ❖ Nhập:

- Số tiền muốn đổi

- ❖ Xuất:

- Số tờ với các mệnh giá: 500000, 200000, 100000, 50000
 - Số tiền còn dư

- ✓ **Minh họa:**

Nhập số tiền muốn đổi: 1375000

Số tờ mệnh giá 500,000: 2

Số tờ mệnh giá 200,000: 1

Số tờ mệnh giá 100,000: 1

Số tờ mệnh giá 50,000: 1

Số tiền còn dư: 25,000

3.4. Tính kết quả 1

- ✓ **Yêu cầu:** Hãy cho biết kết quả xuất ra của đoạn chương trình sau:

```
result = 1 + 2
print('result =', result) 3
original_result = result
result = result - 1 2
print('result =', result) 2
original_result = result 2
result = result * 2 4
original_result = result 4
print('result =', result) 4
result = result ** 3 64
original_result = result 64
print('result =', result) 64
result = result + 8 72
original_result = result 72
print('result =', result) 72
result = result % 7 2
original_result = result 2
print('result =', result) 2
result = result // 2 1
original_result = result 1
print('result =', result) 1
```

Sau khi tính kết quả, hãy viết và chạy đoạn chương trình này để kiểm tra lại.

3.5. Tính kết quả 2

- ✓ **Yêu cầu:** Hãy cho biết kết quả xuất ra của đoạn chương trình sau:

```
result = 5
print('result =', result) 5
result -= 1 4
print('result =', result) 4
result += 3 7
print('result =', result) 7
result = - result -7
print('result =', result) -7
result = True true
print('not result =', not result) false
```

Sau khi tính kết quả, hãy viết và chạy đoạn chương trình này để kiểm tra lại.

3.6. Tính kết quả 3

- ✓ **Yêu cầu:** Hãy cho biết kết quả xuất ra của đoạn chương trình sau:

```
x = 10
y = 4
print('x = %d, y = %d'%(x,y)) x = 10, y= 4
equivalence = x == y False
print('x==y is', equivalence) False
equivalence = x != y True
print('x!=y is', equivalence) true
equivalence = x > y true
print('x>y is', equivalence) true

x = 8
y = 9
print('x = %d, y = %d'%(x,y)) x = 8 y= 9
equivalence = x >= y false
print('x>=y is', equivalence) false
equivalence = x < y true
print('x<y is', equivalence) true
equivalence = x <= y true
print('x<=y is', equivalence) true
```

Sau khi tính kết quả, hãy viết và chạy đoạn chương trình này để kiểm tra lại.

3.7. Tính kết quả 4

- ✓ **Yêu cầu:** Hãy cho biết kết quả xuất ra của đoạn chương trình sau:

```
x = 15
y = 12
print('Binary of x =', bin(x), ', Binary of y =', bin(y))
print('x & y =', bin(x & y))
print('x | y =', bin(x | y))
print('x ^ y =', bin(x ^ y))
print('~x =', bin(~x))
```



```
print('x << 2 =', bin(x << 2))
print('y >> 2 =', bin(y >> 2))
```

Sau khi tính kết quả, hãy viết và chạy đoạn chương trình này để kiểm tra lại.

3.8. Tính kết quả 5

- ✓ **Yêu cầu:** Hãy cho biết kết quả xuất ra của đoạn chương trình sau:

```
x = True
y = False
print('x and y :', x and y) False
print('x or y :', x or y) True
print('not x :', not x) False
print('x is y :', x is y) False
print('x is not y :', x is not y) True
```

Sau khi tính kết quả, hãy viết và chạy đoạn chương trình này để kiểm tra lại.

BÀI 4: Cấu trúc điều kiện



Mục tiêu chính: Cung cấp cho học viên kiến thức và kỹ năng:

- Sử dụng các dạng cấu trúc điều kiện

4.1. Tìm giá trị tuyệt đối

✓ **Yêu cầu:** Viết chương trình tìm giá trị tuyệt đối của một số

✓ **Thuật giải:**

❖ Nhập:

▪ x

❖ Xuất:

▪ |x|

✓ **Minh họa:**

```
Nhập x: -10  
|-10| = 10
```

4.2. Tìm giá trị lớn nhất – nhỏ nhất

✓ **Yêu cầu:** Cho người dùng nhập vào 4 giá trị a, b, c, d. Viết chương trình tìm giá trị lớn nhất, nhỏ nhất và hiển thị ra màn hình.

✓ **Thuật giải:**

❖ Nhập:

▪ a, b, c, d

❖ Xuất:

▪ Số lớn nhất

▪ Số nhỏ nhất

✓ **Minh họa:**

```
Nhập a, b, c, d: 10, 7, 5, 8  
Giá trị lớn nhất: 10  
Giá trị nhỏ nhất: 5
```

4.3. Giải phương trình bậc nhất

✓ **Yêu cầu:** Viết chương trình để giải phương trình bậc nhất $ax + b = 0$

✓ **Thuật giải:**

❖ Nhập:

- a và b
- ❖ Xuất:
 - Nếu a khác 0 → Phương trình có nghiệm duy nhất $x = -b/a$
 - Nếu a bằng 0 và b bằng 0 → Phương trình có vô số nghiệm
 - Nếu a bằng 0 và b khác 0 → Phương trình vô nghiệm

✓ **Minh họa:**

Nhập a: -2
 Nhập b: 9
 Phương trình có 1 nghiệm $x = 4.5$

4.4. Xét năm nhuận

- ✓ **Yêu cầu:** Viết chương trình cho người dùng nhập vào một năm, xét năm đó có phải là năm nhuận hay không.
- ✓ **Mô tả:** có 2 trường hợp:
 - Năm nhuận là năm chia hết cho 4 và không chia hết cho 100.
 - Năm nhuận là năm chia hết cho 400.
- ✓ **Thuật giải:**
 - ❖ Nhập:
 - Năm
 - ❖ Xuất:
 - Xuất ra chuỗi kết quả là năm nhuận hoặc không là năm nhuận.

✓ **Minh họa:**

Nhập năm cần xét: 2023
 Năm 2023 không là năm nhuận

Nhập năm cần xét: 2024
 Năm 2024 là năm nhuận

4.5. Tính cước taxi

- ✓ **Yêu cầu:** Viết chương trình tính cước taxi theo biểu phí cơ bản như sau:

Loại xe	Giá mở cửa (trong phạm vi 500m)	Trong phạm vi 30 km	Từ km thứ 31 trở đi
4 chỗ	11.000	17.600	14.500
7 chỗ	12.000	19.600	17.100

- Thời gian chờ: 5 phút đầu miễn phí, từ phút thứ 6 trở đi tính 750đ/phút.

✓ **Thuật giải:**

- ❖ Nhập:

- Loại xe
- Thời gian chờ
- Số km đã đi
- ❖ Xuất:
 - Tiền chờ
 - Tiền di chuyển
 - Tổng tiền cước phải trả (Tiền di chuyển + Tiền chờ)

✓ **Minh họa:**

```
Loại xe: 4
Thời gian chờ (phút): 8
Số km di chuyển: 50

Tiền chờ: 2,250
Tiền di chuyển: 820,200.0
Tiền cước = 820,200.0 + 2,250 = 822,450.0
```

```
Loại xe: 7
Thời gian chờ (phút): 10
Số km di chuyển: 151

Tiền chờ: 3,750
Tiền di chuyển: 2,659,300.0
Tiền cước = 2,659,300.0 + 3,750 = 2,663,050.0
```

4.6. Tính tiền điện

- ✓ **Yêu cầu:** Tính tiền điện dành cho hộ gia đình phải trả khi biết số Kwh tiêu thụ và cách thức tính theo quy định như sau:

Mức sử dụng trong tháng	Giá bán điện (đồng/kWh)
Bậc 1: Cho kWh từ 0 - 50	1.678
Bậc 2: Cho kWh từ 51 - 100	1.734
Bậc 3: Cho kWh từ 101 - 200	2.014
Bậc 4: Cho kWh từ 201 - 300	2.536
Bậc 5: Cho kWh từ 301 - 400	2.834
Bậc 6: Cho kWh từ 401 trở lên	2.927

(Nguồn tham khảo: <https://www.evn.com.vn/c3/evn-va-khach-hang/Bieu-gia-ban-le-dien-9-79.aspx>)

✓ **Thuật giải:**

- ❖ Nhập:
 - Số kWh
- ❖ Xuất:
 - Tổng số tiền

✓ **Minh họa:**

Số kw tiêu thụ: 398
Tiền điện phải trả: 903,332

4.7. Tính tiền thuê phòng resort

- ✓ **Yêu cầu:** Tính tiền thuê phòng của resort theo bảng giá sau:

Mã loại	Tên loại phòng	Giá 1 đêm	Ở từ 2 - 3 đêm	Ở từ 4 đêm
1	Standard	1,260,000	Giảm 25% so với giá 1 đêm	Giảm 30% so với giá 1 đêm
2	Superior Garden View	1,550,000		
3	Superior Ocean View	1,830,000		
4	Garden View Bungalow	1,830,000		
5	Pool View Bungalow	2,120,000		
6	Family Room	2,120,000		
7	Beach Front Bungalow	2,540,000		
8	VIP sea view	4,800,000		

- ✓ **Thuật giải:**

- ❖ Nhập:
 - Loại phòng
 - Số đêm
- ❖ Xuất:
 - Số tiền

- ✓ **Minh họa:**

Nhập loại phòng (từ 1 đến 8): 6
Nhập số đêm: 4
Thành tiền = 5,936,000.0 VNĐ

4.8. Tính thuế thu nhập cá nhân

- ✓ **Yêu cầu:** Viết chương trình tính thuế thu nhập cá nhân (thuế/năm)

- ✓ **Thuật giải:**

- ❖ Nhập:
 - Tổng thu nhập trong năm
 - Số người phụ thuộc
- ❖ Xuất:
 - Số tiền giảm trừ: Là số tiền không phải đóng thuế
 - Số tiền chịu thuế (hay Thu nhập tính thuế - TNTT) = Tổng thu nhập - Số tiền giảm trừ

- Số tiền thuế phải đóng: Căn cứ theo Số tiền chịu thuế (Thu nhập tính thuế) và tính toán dựa theo quy tắc:

Bậc thuế	Thu nhập tính thuế (TNTT) / năm (triệu đồng)	Thu nhập tính thuế (TNTT) / tháng (triệu đồng)	Thuế suất (%)
1	Đến 60	Đến 5	5%
2	Trên 60 đến 120	Trên 5 đến 10	10%
3	Trên 120 đến 216	Trên 10 đến 18	15%
4	Trên 216 đến 384	Trên 18 đến 32	20%
5	Trên 384 đến 624	Trên 32 đến 52	25%
6	Trên 624 đến 960	Trên 52 đến 80	30%
7	Trên 960	Trên 80	35%

(Biểu thuế lũy tiến từng phần)

1. Giảm trừ gia cảnh là gì?

Theo **Luật Thuế thu nhập cá nhân**, giảm trừ gia cảnh là số tiền được trừ vào thu nhập chịu thuế trước khi tính thuế đối với thu nhập từ kinh doanh, tiền lương, tiền công của đối tượng nộp thuế là cá nhân cư trú.

Việc xác định mức giảm trừ gia cảnh đối với người phụ thuộc thực hiện theo nguyên tắc mỗi người phụ thuộc chỉ được tính giảm trừ một lần vào một đối tượng nộp thuế.

2. Mức giảm trừ gia cảnh năm 2023

Giảm trừ gia cảnh gồm: giảm trừ gia cảnh cho bản thân người nộp thuế và giảm trừ gia cảnh cho người phụ thuộc.

Mức giảm trừ gia cảnh năm 2023 được thực hiện theo **Nghị quyết 954/2020/UBTVQH14**, như sau:

- Mức giảm trừ đối với đối tượng nộp thuế là 11 triệu đồng/tháng (132 triệu đồng/năm);
- Mức giảm trừ đối với mỗi người phụ thuộc là 4,4 triệu đồng/tháng.

Nguồn tham khảo: <https://thuvienphapluat.vn/chinh-sach-phap-luat-moi/vn/thoi-su-phap-luat/tu-van-phap-luat/39480/muc-giam-tru-gia-can>

✓ Minh họa:

Số tiền thu nhập trong năm: 250000000
Số người phụ thuộc: 1

Số tiền giảm trừ: 184,800,000
Số tiền chịu thuế: 65,200,000
Tiền thuế: 3,520,000

BÀI 5: Cấu trúc lặp



Mục tiêu chính: Cung cấp cho học viên kiến thức và kỹ năng:

- Sử dụng vòng lặp *while*
- Sử dụng vòng lặp *for*

5.1. Count down

- ✓ **Yêu cầu:** Nhập vào một số nguyên n. Thực hiện việc count down:
- ✓ **Thuật giải:**
 - ❖ Nhập:
 - n
 - ❖ Xuất:
 - Count down
- ✓ **Minh họa:**

```
Nhập số: 10
10
9
8
7
6
5
4
3
2
1
Start!!!
```

5.2. Tính tổng n số nguyên nhập vào

- ✓ **Yêu cầu:** Xây dựng chương trình tính tổng n số nguyên nhập vào:
- ✓ **Thuật giải:**
 - ❖ Nhập:
 - n số nguyên
 - Các giá trị số nguyên trong vòng lặp
 - ❖ Xuất:
 - Tổng n số nguyên
- ✓ **Minh họa:**

```
***** CHƯƠNG TRÌNH TÍNH TỔNG N SỐ NGUYÊN *****
Nhập n: 5
Nhập số nguyên thứ 1: 10
Nhập số nguyên thứ 2: 56
Nhập số nguyên thứ 3: 22
Nhập số nguyên thứ 4: 17
Nhập số nguyên thứ 5: 90
Tổng = 195
```

5.3. Xuất chuỗi

- ✓ **Yêu cầu:** Xây dựng chương trình cho nhập vào các ký tự cho đến khi nào không muốn nhập nữa (dừng lại ấn phím "0"), xuất ra các ký tự đã nhập
- ✓ **Thuật giải:**
 - ❖ Nhập:
 - Ký tự chuỗi
 - ❖ Xuất:
 - Các ký tự đã nhập
- ✓ **Minh họa:**

```
Nhập ký tự (dừng lại ấn phím 0): p
Nhập ký tự (dừng lại ấn phím 0): y
Nhập ký tự (dừng lại ấn phím 0): t
Nhập ký tự (dừng lại ấn phím 0): h
Nhập ký tự (dừng lại ấn phím 0): o
Nhập ký tự (dừng lại ấn phím 0): n
Nhập ký tự (dừng lại ấn phím 0): 0
Các ký tự đã nhập: python
```

5.4. Kiểm tra số nguyên tố

- ✓ **Yêu cầu:** Nhập vào một số x kiểm tra xem x có phải là số nguyên tố hay không. (Gợi ý: Số nguyên tố là số chỉ chia hết cho 1 và chính nó)
- ✓ **Thuật giải:**
 - ❖ Nhập:
 - x
 - ❖ Xuất:
 - Là số nguyên tố / Không là số nguyên tố
- ✓ **Minh họa:**

```
Nhập số cần kiểm tra: 6
Số 6 không là số nguyên tố

Nhập số cần kiểm tra: 7
Số 7 là số nguyên tố
```


5.5. Tính S

✓ **Yêu cầu:** Nhập vào một số nguyên n và một số thực x . Tính và in ra kết quả sau: $S = (x^2 + 1)^n$

✓ **Thuật giải:**

❖ Nhập:

- n
- x

❖ Xuất:

- S

❖ Xử lý:

- Sử dụng vòng lặp để tính giá trị biểu thức S

✓ **Minh họa:**

```
Nhập n: 4
Nhập x: 5
S = (x * x + 1) ^ n = 456976
```

5.6. Tính A

✓ **Yêu cầu:** Nhập vào một số nguyên n và một số thực x . Tính và in ra kết quả của biểu thức sau:

$$A = (x^2 + x + 1)^n + (x^2 - x + 1)^n$$

✓ **Thuật giải:**

❖ Nhập:

- n
- x

❖ Xuất:

- A

❖ Xử lý:

- Sử dụng vòng lặp để tính giá trị biểu thức A

✓ **Minh họa:**

```
Nhập n: 3
Nhập x: 2
A = (x2 + x + 1) ^ n + (x2 - x + 1) ^ n = 370
```

5.7. Tính giá trị biểu thức

✓ **Yêu cầu:** Nhập vào 2 số nguyên a và b ($a < b$), tính các biểu thức sau đây:

- A = tổng các số lẻ từ a đến b
- B = tổng các số chẵn từ a đến b
- C = tích các số từ a đến b

- D = tích các số chia hết cho 3 từ a đến b
- E = tổng các số nguyên tố từ a đến b

✓ **Thuật giải:**

- ❖ Nhập:
 - a, b
- ❖ Xuất:
 - A, B, C, D, E
- ❖ Xử lý:
 - Sử dụng vòng lặp

✓ **Minh họa:**

```
Nhập a: 5
Nhập b: 12
A = 5 + 7 + 9 + 11 + = 32
B = 6 + 8 + 10 + 12 + = 36
C = 5 * 6 * 7 * 8 * 9 * 10 * 11 * 12 * = 19958400
D = 6 * 9 * 12 * = 648
E = 5 + 7 + 11 + = 23
```

5.8. In bảng cửu chương

- ✓ **Yêu cầu:** Viết chương trình in bảng cửu chương từ n (số bắt đầu) đến m (số kết thúc)

✓ **Thuật giải:**

- ❖ Nhập:
 - n, m
- ❖ Xuất:
 - Bảng cửu chương trong khoảng từ n đến m
- ❖ Xử lý:
 - Sử dụng vòng lặp

✓ **Minh họa:**

```
Từ cửu chương: 3
Đến cửu chương: 8
```

3 x 1 = 3	4 x 1 = 4	5 x 1 = 5	6 x 1 = 6	7 x 1 = 7	8 x 1 = 8
3 x 2 = 6	4 x 2 = 8	5 x 2 = 10	6 x 2 = 12	7 x 2 = 14	8 x 2 = 16
3 x 3 = 9	4 x 3 = 12	5 x 3 = 15	6 x 3 = 18	7 x 3 = 21	8 x 3 = 24
3 x 4 = 12	4 x 4 = 16	5 x 4 = 20	6 x 4 = 24	7 x 4 = 28	8 x 4 = 32
3 x 5 = 15	4 x 5 = 20	5 x 5 = 25	6 x 5 = 30	7 x 5 = 35	8 x 5 = 40
3 x 6 = 18	4 x 6 = 24	5 x 6 = 30	6 x 6 = 36	7 x 6 = 42	8 x 6 = 48
3 x 7 = 21	4 x 7 = 28	5 x 7 = 35	6 x 7 = 42	7 x 7 = 49	8 x 7 = 56
3 x 8 = 24	4 x 8 = 32	5 x 8 = 40	6 x 8 = 48	7 x 8 = 56	8 x 8 = 64
3 x 9 = 27	4 x 9 = 36	5 x 9 = 45	6 x 9 = 54	7 x 9 = 63	8 x 9 = 72
3 x 10 = 30	4 x 10 = 40	5 x 10 = 50	6 x 10 = 60	7 x 10 = 70	8 x 10 = 80

BÀI 6: Number – String – Datetime



Mục tiêu chính: Cung cấp cho học viên kiến thức và kỹ năng:

- Sử dụng các hàm xử lý số học (thư viện *math*, *random*)
- Sử dụng các hàm xử lý chuỗi
- Sử dụng các hàm xử lý ngày giờ

6.1. Tính bán kính hình tròn

- ✓ **Yêu cầu:** Nhập vào diện tích hình tròn. Tính và in ra bán kính của hình tròn:

Diện tích = $\text{PI} * \text{Bán kính} * \text{Bán kính}$

=> Bán kính = ?

- ✓ **Thuật giải:**

❖ Nhập:

- Diện tích hình tròn

❖ Xuất:

- Bán kính hình tròn

- ✓ **Minh họa:**

```
Nhập diện tích hình tròn: 113  
Bán kính hình tròn: 6.0
```

6.2. Giải phương trình bậc hai

- ✓ **Yêu cầu:** Xây dựng chương trình giải phương trình bậc hai:

$$ax^2 + bx + c = 0$$

- ✓ **Thuật giải:**

❖ Nhập:

- a, b, c

❖ Xuất:

- Nghiệm

- ✓ **Quy tắc xử lý:**

- Nếu a bằng 0: giải phương trình bậc nhất: $bx + c = 0$
 - Nếu b khác 0: phương trình có nghiệm $x = -c/b$
 - Nếu b bằng 0, c khác 0: phương trình vô nghiệm
 - Nếu b bằng 0, c bằng 0: phương trình vô số nghiệm
- Nếu a khác 0:
 - Tính $\Delta = b^2 - 4ac$
 - Nếu $\Delta < 0$ thì phương trình vô nghiệm.

- Nếu $\Delta = 0$ thì phương trình có nghiệm kép : $x_1 = x_2 = \frac{-b}{2a}$
- Nếu $\Delta > 0$ thì phương trình có 2 nghiệm phân biệt : $x_{1,2} = \frac{-b \pm \sqrt{\Delta}}{2a}$

✓ **Minh họa:**

```

Nhập a: 2
Nhập b: -7
Nhập c: 3
Phương trình có hai nghiệm phân biệt:
x1 = 3.0
x2 = 0.5

Nhập a: 6
Nhập b: 1
Nhập c: 5
Phương trình vô nghiệm

Nhập a: 1
Nhập b: -8
Nhập c: 16
Phương trình có nghiệm kép x1 = x2 = 4.0

Nhập a: 0
Nhập b: 2
Nhập c: 4
Nghiệm = -2.0

```

6.3. Tính diện tích tam giác

- ✓ **Yêu cầu:** Viết chương trình tính diện tích tam giác bằng công thức Heron:

Gọi **S** là diện tích và độ dài 3 cạnh tam giác lần lượt là **a, b, c**

$$S = \sqrt{p(p-a)(p-b)(p-c)}$$

với **p** là nửa chu vi của tam giác:

$$p = \frac{a+b+c}{2}$$

- ✓ **Thuật giải:**

- ❖ Nhập:
 - a, b, c
- ❖ Xuất:
 - S

✓ **Minh họa:**

Nhập chiều dài 3 cạnh tam giác: 4, 5, 6
 Diện tích tam giác: 9.921567416492215

Nhập chiều dài 3 cạnh tam giác: 1, 2, 3
 Chiều dài 3 cạnh tam giác không hợp lệ

6.4. Tìm kiếm thức uống

- ✓ **Yêu cầu:** Viết chương trình cho phép nhập vào từ khóa (không phân biệt HOA thường), thực hiện chức năng tìm kiếm và hiển thị kết quả ra màn hình. Cho danh sách:

```
danh_sach_thuc_uong = ['Bạc Xiu Đá', 'Freeze Trà Xanh',
                        'Trà Thạch Vải', 'Trà Thanh Đào',
                        'Cappuccino', 'Cà Phê Sữa Đá']
```

✓ **Thuật giải:**

- ❖ Nhập:
 - Từ khóa
- ❖ Xuất:
 - Kết quả tìm kiếm

✓ **Minh họa:**

Nhập từ khóa cần tìm: trà

Tìm thấy 3 thức uống với từ khóa "trà":
 Freeze Trà Xanh, Trà Thạch Vải, Trà Thanh Đào

Nhập từ khóa cần tìm: nhãn

Không tìm thấy thức uống với từ khóa "nhãn"

6.5. Tìm số giá trị số nguyên và tính tổng

- ✓ **Yêu cầu:** Viết chương trình tìm kiếm các từ trong chuỗi văn bản sau là kiểu số nguyên và thực hiện tính tổng các giá trị số nguyên đó

Python 1.0 đã được ra mắt vào năm 1994 với các hàm mới để dễ dàng xử lý danh sách dữ liệu, chẳng hạn như ánh xạ, lọc và lược bỏ.

Python 2.0 đã được ra mắt vào ngày 16 tháng 10 năm 2000, với các tính năng hữu ích mới cho lập trình viên, chẳng hạn như hỗ trợ ký tự Unicode và cách xử lý chi tiết một danh sách nhanh chóng hơn.

Python 3.0 đã được ra mắt vào ngày 3 tháng 12 năm 2008. Phiên bản này bao gồm các tính năng như hàm in và hỗ trợ nhiều hơn cho việc phân chia số và xử lý lỗi.

✓ **Thuật giải:**

- ❖ Nhập:
 - Cho sẵn dữ liệu cung cấp

❖ **Xuất:**

- Đếm số lượng từ là số nguyên
- Tính tổng các giá trị số nguyên tìm được

✓ **Minh họa:**

Tìm thấy 10 từ trong chuỗi là số nguyên
 Tổng các giá trị số nguyên là:
 $1 + 1994 + 2 + 16 + 10 + 2000 + 3 + 3 + 12 + 2008 = 6049$

6.6. Làm việc với datetime

✓ **Yêu cầu:** Viết chương trình cho phép người dùng nhập vào ngày tháng năm sinh. Xuất ra màn hình kết quả:

- Ngày sinh nhật kế tiếp là ngày mấy?
- Còn bao nhiêu ngày (*so_ngay*) nữa là đến ngày sinh nhật lần thứ (*so_lan_sinh_nhat*) của người dùng?

✓ **Thuật giải:**

❖ **Nhập:**

- Ngày tháng năm sinh theo dd/MM/yyyy

❖ **Xuất:**

- Thực hiện theo các yêu cầu trên

✓ **Quy tắc xử lý:** Kiểm tra nếu như ngày sinh nhật trong năm hiện tại đã qua thì xuất ngày sinh nhật kế tiếp sẽ là thuộc năm tiếp theo.

✓ **Minh họa:**

Nhập ngày sinh (dd/MM/yyyy): **31/12/1991**
 Ngày sinh nhật kế tiếp của bạn: 31/12/2022
 Còn 4 ngày nữa là đến ngày sinh nhật lần thứ 31 của bạn

Nhập ngày sinh (dd/MM/yyyy): **11/11/1991**
 Ngày sinh nhật kế tiếp của bạn: 11/11/2023
 Còn 319 ngày nữa là đến ngày sinh nhật lần thứ 32 của bạn

(Ngày hiện tại thực hiện trong ví dụ này là: 27/12/2022)

6.7. Làm việc với calendar

✓ **Yêu cầu:** Viết chương trình cho phép nhập vào tháng (*thang*) và năm (*nam*). Thực hiện các yêu cầu sau:

- Kiểm tra năm nhập vào có phải là năm nhuận không?
- Tháng và năm nhập vào có bao nhiêu ngày?
- Ngày cuối cùng của tháng và năm nhập vào là thứ mấy?
- Ngày Chủ nhật đầu tiên của tháng và năm nhập vào là ngày mấy?
- In ra lịch của tháng đó

✓ **Thuật giải:**

❖ Nhập:

- thang, nam

❖ Xuất:

- Thực hiện theo các yêu cầu trên

✓ **Minh họa:**

Nhập tháng: **1**

Nhập năm: **2023**

Năm 2023 không là năm nhuận

Tháng 1 năm 2023 có 31 ngày

Ngày cuối cùng của tháng 1 năm 2023 là Thứ ba

Ngày Chủ nhật đầu tiên của tháng 1 năm 2023 là ngày 1

January 2023

Mo Tu We Th Fr Sa Su

1

2 3 4 5 6 7 8

9 10 11 12 13 14 15

16 17 18 19 20 21 22

23 24 25 26 27 28 29

30 31

BÀI 7: List – Tuple – Dictionary – Set



Mục tiêu chính: Cung cấp cho học viên kiến thức và kỹ năng sử dụng:

- List
- Tuple
- Dictionary
- Set

7.1. Danh sách thú

- ✓ **Yêu cầu:** Viết chương trình tìm thú trong danh sách thú. Xuất ra kết quả tìm thấy hoặc không tìm thấy thú và in ra màn hình.
- ✓ **Thuật giải:**
 - ❖ Nhập:
 - Tên thú cần tìm
 - ❖ Xuất:
 - Tìm thấy (Nếu tìm thấy thì chỉ ra vị trí index)
 - Hoặc không tìm thấy
- ✓ **Minh họa:**

```
Nhập thú cần tìm: elephant
Tìm thấy "elephant" trong danh sách tại vị trí index = 4

Nhập thú cần tìm: lion
Không tìm thấy "lion" trong danh sách
```

7.2. Tìm từ duy nhất

- ✓ **Yêu cầu:** Viết chương trình tìm từ duy nhất được sử dụng trong câu sau:
`I am a teacher and I love to inspire and teach people`
- ✓ **Thuật giải:**
 - ❖ Nhập:
 - Sử dụng chuỗi trên hoặc cho phép nhập 1 chuỗi từ bàn phím.
 - ❖ Xuất:
 - Danh sách các từ duy nhất trong câu.
- ✓ **Minh họa:**

```
Các từ duy nhất được sử dụng trong câu:
['am', 'a', 'teacher', 'love', 'to', 'inspire', 'teach', 'people']
```

7.3. List numbers

- ✓ **Yêu cầu:** Viết chương trình xử lý list theo các yêu cầu sau:
 - Tạo list: Cho phép người dùng lần lượt nhập các phần tử số cho list cho đến khi không muốn nhập nữa.
 - Tính tổng các phần tử trong list.
 - Nhập vào giá trị số nguyên x.
 - x có xuất hiện trong list hay không không? Nếu có thì cho biết x xuất hiện bao nhiêu lần?
 - x có lớn hơn tất cả các số trong list không? Nếu không thì x nhỏ hơn những số nào trong list? (In ra tất cả các số lớn hơn x).
 - Tìm và in ra các phần tử âm / phần tử dương trong list (sắp xếp tăng dần).
 - Tìm và in ra tất cả các số nguyên tố có trong list (sắp xếp tăng dần).
 - Tính trung bình cộng của các phần tử âm / phần tử dương / số nguyên tố trong list.
- ✓ **Thuật giải:**
 - ❖ Nhập:
 - Các phần tử trong list
 - ❖ Xuất:
 - Thực hiện theo các yêu cầu trên
- ✓ **Minh họa:**

```

Nhập giá trị: 6
Tiếp tục? 1. Có 0. Không 1
Nhập giá trị: -2
Tiếp tục? 1. Có 0. Không 1
Nhập giá trị: 7
Tiếp tục? 1. Có 0. Không 1
Nhập giá trị: -33
Tiếp tục? 1. Có 0. Không 1
Nhập giá trị: 7
Tiếp tục? 1. Có 0. Không 1
Nhập giá trị: 20
Tiếp tục? 1. Có 0. Không 1
Nhập giá trị: -1
Tiếp tục? 1. Có 0. Không 1
Nhập giá trị: 4
Tiếp tục? 1. Có 0. Không 0
-----
Danh sách số đã nhập: [6, -2, 7, -33, 7, 20, -1, 4]
Tổng các phần tử trong list: 8
Nhập x: 7
7 xuất hiện trong list 2 lần
Danh sách số âm: [-33, -2, -1]
Danh sách số dương: [4, 6, 7, 7, 20]
Danh sách số nguyên tố: [7, 7]
Trung bình cộng các số âm: -12.0
Trung bình cộng các số dương: 8.8
Trung bình cộng các số nguyên tố: 7.0
  
```

7.4. Quản lý phim

- ✓ **Yêu cầu:** Viết chương trình quản lý danh sách phim theo các yêu cầu:

Cho trước **danh_sach_phim** như sau:

```
danh_sach_phim = ['Dòng chảy của nước', 'Đảo độc đắc', 'Tiểu đội gấu bay',
                  'Âm lượng hủy diệt', 'Búp bê gọi hồn', 'Tro tàn rực rỡ', 'Nữ chiến binh Amazon', 'Khí con lon ton thế giới']
```

- Tạo thực đơn có 4 chức năng: *Đọc danh sách, Thêm phim mới, Cập nhật phim, Xóa phim*
- Hiển thị danh sách phim

```
----- QUẢN LÝ PHIM -----
Bạn muốn làm gì?
1. Đọc danh sách
2. Thêm phim mới
3. Cập nhật phim
4. Xóa phim
Mời bạn chọn: 1

Có 6 phim trong danh sách
STT      TÊN PHIM
1        Dòng chảy của nước
2        Đảo độc đắc
3        Tiểu đội gấu bay
4        Âm lượng hủy diệt
5        Búp bê gọi hồn
6        Tro tàn rực rỡ

Bạn có muốn tiếp tục không? (y/n)
Mời bạn chọn:
```

- Thêm phim mới: Cho phép người dùng nhập tên phim cho đến khi nào không muốn nhập nữa

```
----- QUẢN LÝ PHIM -----
Bạn muốn làm gì?
1. Đọc danh sách
2. Thêm phim mới
3. Cập nhật phim
4. Xóa phim
Mời bạn chọn: 2

Nhập tên phim (dừng lại bấm phím "s"): Nữ chiến binh Amazon
Nhập tên phim (dừng lại bấm phím "s"): Khí con Lon ton thế giới
Nhập tên phim (dừng lại bấm phím "s"): s

Bạn có muốn tiếp tục không? (y/n)
Mời bạn chọn:
```

=> Sau khi thêm, kiểm tra lại kết quả bằng cách chọn lại chức năng 1

- Cập nhật/Thay đổi tên phim: Cho người dùng chọn phim cần cập nhật (chọn STT)
- Nếu nhập STT có trong danh sách thì cho phép nhập tên phim và cập nhật vào danh sách

```

----- QUẢN LÝ PHIM -----
Bạn muốn làm gì?
1. Đọc danh sách
2. Thêm phim mới
3. Cập nhật phim
4. Xóa phim
Mời bạn chọn: 3

Có 6 phim trong danh sách
STT      TÊN PHIM
1        Dòng chảy của nước
2        Đảo độc đắc
3        Tiểu đội gấu bay
4        Âm lượng hủy diệt
5        Búp bê gọi hồn
6        Tro tàn rực rỡ

Chọn phim cần cập nhật
Nhập STT: 1
Nhập tên phim (dừng lại bấm phím "s"): Chiến tranh và hòa bình

Bạn có muốn tiếp tục không? (y/n)
Mời bạn chọn:

```

=> Sau khi cập nhật, có thể kiểm tra lại kết quả bằng chức năng 1

- Nếu nhập STT không có trong danh sách thì thông báo ra màn hình

```

----- QUẢN LÝ PHIM -----
Bạn muốn làm gì?
1. Đọc danh sách
2. Thêm phim mới
3. Cập nhật phim
4. Xóa phim
Mời bạn chọn: 3

Có 6 phim trong danh sách
STT      TÊN PHIM
1        Dòng chảy của nước
2        Đảo độc đắc
3        Tiểu đội gấu bay
4        Âm lượng hủy diệt
5        Búp bê gọi hồn
6        Tro tàn rực rỡ

Chọn phim cần cập nhật
Nhập STT: 8

Vui lòng nhập STT trong khoảng từ 1 đến 6

```

- Xóa phim: Cho người dùng chọn phim cần cập nhật (chọn STT)

```

----- QUẢN LÝ PHIM -----
Bạn muốn làm gì?
1. Đọc danh sách
2. Thêm phim mới
3. Cập nhật phim
4. Xóa phim
Mời bạn chọn: 4

Có 6 phim trong danh sách
STT      TÊN PHIM
1        Dòng chảy của nước
2        Đảo độc đắc
3        Tiểu đội gấu bay
4        Âm lượng hủy diệt
5        Búp bê gọi hồn
6        Tro tàn rực rỡ

Chọn phim cần xóa
Nhập STT: 3

Đã xóa phim "Tiểu đội gấu bay" ra khỏi danh sách

Bạn có muốn tiếp tục không? (y/n)
Mời bạn chọn:

```

=> Sau khi xóa, có thể kiểm tra lại kết quả bằng chức năng 1

✓ Thuật giải:

❖ Nhập/Xuất:

- Thực hiện nhập theo yêu cầu từng chức năng của chương trình

7.5. Tuple string

✓ Yêu cầu: Viết chương trình xử lý các yêu cầu sau:

- Cho tuple:

```
tuple_colors = ('red', 'green', 'yellow', 'blue', 'black', 'white', 'pink', 'orange', 'red', 'blue')
```

- Cho người dùng nhập vào *index* ($0 \leq \text{index} \leq \text{len}-1$) và *negative index* ($-\text{len} < \text{negative_index} \leq -1$)
- Cho người dùng nhập vào 1 màu cần tìm (*find_color*).
- Hiển thị giá trị tại vị trí *index*, *negative_index*.
- Hiển thị kết quả số lần xuất hiện của *find_color* đã nhập. Nếu có xuất hiện thì chỉ ra các vị trí index. Ngược lại thì thông báo không tìm thấy.
- Sắp xếp các giá trị trong *tuple_colors* theo thứ tự tăng dần.

✓ **Thuật giải:**

❖ **Nhập:**

- index, negative index, phần tử cần tìm

❖ **Xuất:**

- Theo các yêu cầu trên.

✓ **Minh họa:**

```
Nhập index (từ 0 đến 9): 6
Nhập negative index (từ -10 đến -1): -5
Nhập màu cần tìm: blue

-----
- tuple_colors[6] = pink
- tuple_colors[-5] = white
- "blue" xuất hiện 2 lần trong danh sách tại vị trí index: 3, 9
- Danh sách sau khi sắp xếp tăng dần:
('black', 'blue', 'blue', 'green', 'orange', 'pink', 'red', 'red', 'white', 'yellow')
```

7.6. Từ điển

✓ **Yêu cầu:** Viết chương trình làm việc với từ điển, thực hiện theo các yêu cầu:

Cho trước **tu_dien** như sau:

```
tu_dien = {
    'man': ['đàn ông', 'nam nhi'], 'woman': ['đàn bà', 'phụ nữ'],
    'sun': ['mặt trời'], 'moon': ['mặt trăng'],
    'earth': ['trái đất', 'địa cầu'],
    'mountain': ['núi', 'ngọn núi'], 'table': ['cái bàn'],
    'ball': ['quả bóng'], 'flower': ['hoa', 'bông hoa'],
    'fan': ['cái quạt']
}
```

- Tạo thực đơn có 4 chức năng: *Tra từ điển*, *Thêm từ mới*, *Thêm nghĩa* (từ đang có trong tu_dien), *Xóa từ*
- Tra từ: Cho người dùng nhập vào từ tiếng anh cần tra, xuất ra danh sách nghĩa tiếng việt

```
Bạn muốn làm gì?
1. Tra từ điển
2. Thêm từ
3. Thêm nghĩa
4. Xóa từ
Mời bạn chọn: 1

Nhập từ cần tra: mountain

Từ "mountain" có nghĩa là:
- núi
- ngọn núi

Bạn có muốn tiếp tục không? (y/n)
Mời bạn chọn:
```

- Thêm từ: Cho phép người dùng thêm từ mới và nghĩa vào danh sách *tu_dien*.
 - Kiểm tra nếu từ mới đã tồn tại trong *tu_dien* thì không thực hiện thêm và xuất thông báo ra màn hình.

```
Bạn muốn làm gì?
1. Tra từ điển
2. Thêm từ
3. Thêm nghĩa
4. Xóa từ
Mời bạn chọn: 2

Nhập từ mới: moon
Từ "moon" đã tồn tại trong từ điển

Bạn có muốn tiếp tục không? (y/n)
Mời bạn chọn:
```

- Nếu từ mới không tồn tại trong *tu_dien* thì cho người dùng nhập danh sách nghĩa cho đến khi nào không muốn nhập nữa.

```
Bạn muốn làm gì?
1. Tra từ điển
2. Thêm từ
3. Thêm nghĩa
4. Xóa từ
Mời bạn chọn: 2

Nhập từ mới: house
Nhập nghĩa của từ "house" (dừng lại bấm phím "s"): ngôi nhà
Nhập nghĩa của từ "house" (dừng lại bấm phím "s"): căn nhà
Nhập nghĩa của từ "house" (dừng lại bấm phím "s"): nhà
Nhập nghĩa của từ "house" (dừng lại bấm phím "s"): s

Đã thêm từ "house" vào từ điển thành công

Bạn có muốn tiếp tục không? (y/n)
Mời bạn chọn:
```

=> Sau khi thêm thành công, có thể dùng chức năng tra từ điển để kiểm tra:

```
Nhập từ cần tra: house

Từ "house" có nghĩa là:
- ngôi nhà
- căn nhà
- nhà
```

- Thêm nghĩa: Cho phép người dùng nhập từ tiếng anh để thêm nghĩa

- Nếu từ tiếng anh không tồn tại trong *tu_dien* thì thông báo ra màn hình.

```
Bạn muốn làm gì?
1. Tra từ điển
2. Thêm từ
3. Thêm nghĩa
4. Xóa từ
Mời bạn chọn: 3

Nhập từ tiếng anh cần cập nhật nghĩa: computer

Không tìm thấy từ "computer" trong từ điển

Bạn có muốn tiếp tục không? (y/n)
Mời bạn chọn:
```

- Nếu từ tiếng anh tồn tại trong *tu_dien* thì cho phép người dùng nhập thêm nghĩa cho đến khi không muốn nhập nữa.

```
Bạn muốn làm gì?
1. Tra từ điển
2. Thêm từ
3. Thêm nghĩa
4. Xóa từ
Mời bạn chọn: 3

Nhập từ tiếng anh cần cập nhật nghĩa: ball
Nhập nghĩa của từ "ball" (dừng lại bấm phím "s"): trái banh
Nhập nghĩa của từ "ball" (dừng lại bấm phím "s"): s

Đã thêm nghĩa của từ "ball" vào từ điển thành công

Bạn có muốn tiếp tục không? (y/n)
Mời bạn chọn:
```

=> Sau khi thêm thành công, có thể dùng chức năng tra từ điển để kiểm tra:

```
Nhập từ cần tra: ball

Từ "ball" có nghĩa là:
- quả bóng
- trái banh
```

- Xóa từ: Cho phép người dùng nhập từ tiếng anh cần xóa.
 - Kiểm tra nếu từ tiếng anh tồn tại trong *tu_dien* thì thực hiện xóa.

```

Bạn muốn làm gì?
1. Tra từ điển
2. Thêm từ
3. Thêm nghĩa
4. Xóa từ
Mời bạn chọn: 4

Nhập từ tiếng anh cần xóa: sun

Đã xóa từ "sun" thành công

Bạn có muốn tiếp tục không? (y/n)
Mời bạn chọn:

```

=> Sau khi thêm thành công, có thể dùng chức năng tra từ điển để kiểm tra:

```

Nhập từ cần tra: sun

Không tìm thấy từ "sun" trong từ điển

```

- Nếu không tồn tại trong *tu_dien* thì thông báo ra màn hình.

```

Bạn muốn làm gì?
1. Tra từ điển
2. Thêm từ
3. Thêm nghĩa
4. Xóa từ
Mời bạn chọn: 4

Nhập từ tiếng anh cần xóa: television

Không tìm thấy từ "television" trong từ điển

Bạn có muốn tiếp tục không? (y/n)
Mời bạn chọn:

```

✓ Thuật giải:

❖ Nhập/Xuất:

- Thực hiện nhập theo yêu cầu từng chức năng của chương trình

7.7. Set number

- ✓ **Yêu cầu:** Viết chương trình thực hiện việc xử lý trên set như sau:

Tạo set: Cho phép người dùng nhập vào **set 1** cho đến khi không muốn nhập nữa thì chuyển sang cho người dùng nhập tiếp **set 2**. Sau khi người dùng nhập xong set 2 thì thực hiện các yêu cầu sau:

- In ra set 1 và set 2 vừa tạo.
- In ra chiều dài (số phần tử) của set 1 và set 2.
- Tính tổng các phần tử trong set 1 và set 2.

- Tìm giá trị lớn nhất và nhỏ nhất trong set 1 và set 2.
- Sắp xếp giá trị giảm dần trong 2 set.
- Thực hiện set union của set 1 và set 2, in ra kết quả.
- Thực hiện set intersection của set 1 và set 2, in ra kết quả.
- Thực hiện set difference của set 1 và set 2, in ra kết quả.
- Thực hiện set symmetric difference của set 1 và set 2, in ra kết quả.

✓ **Thuật giải:**

❖ Nhập:

- Các giá trị của set 1 và set 2.

❖ Xuất:

- Thực hiện các yêu cầu trên.

✓ **Minh họa:**

```

Nhập giá trị cho set 1: 5
Tiếp tục nhập set 1? 1. Có 0. Không 1
Nhập giá trị cho set 1: 2
Tiếp tục nhập set 1? 1. Có 0. Không 1
Nhập giá trị cho set 1: 7
Tiếp tục nhập set 1? 1. Có 0. Không 1
Nhập giá trị cho set 1: 4
Tiếp tục nhập set 1? 1. Có 0. Không 0
Nhập giá trị cho set 2: 7
Tiếp tục nhập set 2? 1. Có 0. Không 1
Nhập giá trị cho set 2: 8
Tiếp tục nhập set 2? 1. Có 0. Không 1
Nhập giá trị cho set 2: 10
Tiếp tục nhập set 2? 1. Có 0. Không 1
Nhập giá trị cho set 2: 6
Tiếp tục nhập set 2? 1. Có 0. Không 0
----- SET 1 -----
Set 1: {2, 4, 5, 7}
Chiều dài: 4
Tổng: 18
Max: 7
Min: 2
Sắp xếp giảm dần: [7, 5, 4, 2]
----- SET 2 -----
Set 2: {8, 10, 6, 7}
Chiều dài: 4
Tổng: 31
Max: 10
Min: 6
Sắp xếp giảm dần: [10, 8, 7, 6]
-----
Set 1 union Set 2: {2, 4, 5, 6, 7, 8, 10}
Set 1 intersection Set 2: {7}
Set 1 difference Set 2: {2, 4, 5}
Set 1 symmetric difference Set 2: {2, 4, 5, 6, 8, 10}

```

BÀI 8: Phương thức / Hàm



Mục tiêu chính: Cung cấp cho học viên kiến thức và kỹ năng:

- Xây dựng và gọi phương thức / hàm

8.1. Tính năm âm lịch

- ✓ **Yêu cầu:** Viết chương trình cho người dùng nhập vào năm dương lịch, tính và hiển thị kết quả năm âm lịch.

- ✓ **Thuật giải:**

❖ Nhập:

- Năm dương lịch

❖ Xuất:

- Năm âm lịch

❖ Xử lý: Xây dựng hàm `tin_h_can(nam)` và hàm `tin_h_chi(nam)` theo quy tắc sau:

- Tính can: Năm dương lịch chia lấy dư cho 10, kết quả dựa theo bảng sau:

0	1	2	3	4	5	6	7	8	9
Canh	Tân	Nhâm	Quý	Giáp	Ất	Bính	Đinh	Mậu	Kỷ

- Tính chi: Năm dương lịch chia lấy dư cho 12, kết quả dựa theo bảng sau:

0	1	2	3	4	5	6	7	8	9	10	11
Thân	Dậu	Tuất	Hợi	Tý	Sửu	Dần	Mão	Thìn	Tỵ	Ngọ	Mùi

- ✓ **Minh họa:**

```
Nhập năm: 2022
Năm 2022 âm lịch là năm Nhâm Dần

Nhập năm: 2023
Năm 2023 âm lịch là năm Quý Mão
```

8.2. Tính chỉ số BMI

- ✓ **Yêu cầu:** Viết chương trình tính chỉ số BMI.

- ✓ **Thuật giải:**

❖ Nhập:

- Chiều cao (m)
- Cân nặng (kg)

- ❖ Xuất:
 - Chỉ số BMI của cơ thể
 - Kết luận
- ❖ Xử lý:
 - Xây dựng hàm `ting_bmi(can_nang, chieu_cao)` có giá trị trả về là chỉ số BMI theo công thức: **BMI = Cân nặng / (Chiều cao * Chiều cao)**
 - Xây dựng hàm `danh_gia_bmi(bmi)` có giá trị trả về là một chuỗi theo các điều kiện sau (có thể chọn 1 trong 2 tiêu chuẩn trong bảng):

Phân loại	Tiêu chuẩn của tổ chức Y tế Thế giới WHO	Tiêu chuẩn dành riêng cho người Châu Á
Thiếu cân (Gầy)	< 18.5	< 18.5
Bình thường	18.5 – 24.9	18.5 – 22.9
Thừa cân	25 – 29.9	23 – 24.9
Béo phì	> 30	> 25

✓ **Minh họa:**

```
Nhập cân nặng: 65
Nhập chiều cao: 1.74
Chỉ số BMI: 21.469150482230148
Kết quả đánh giá: Bình thường
```

8.3. Tính chu vi, diện tích

- ✓ **Yêu cầu:** Viết chương trình cho người dùng lựa chọn thông qua thực đơn (như hình bên dưới). Thực hiện tính chu vi, diện tích hình tròn hoặc hình chữ nhật.

```
Bạn muốn làm gì?
1. Tính chu vi/diện tích hình tròn
2. Tính chu vi/diện tích hình chữ nhật
Mời bạn chọn:
```

✓ **Thuật giải:**

- ❖ Nhập:
 - r (bán kính)
 - a, b (chiều dài, chiều rộng) hình chữ nhật
- ❖ Xuất:
 - S, P hình tròn
 - S, P hình chữ nhật
- ❖ Xử lý:
 - Xây dựng hàm tính chu vi và diện tích hình tròn (sử dụng *lambda*):

- Chu vi: $r * 2 * \pi$
- Diện tích: $r^2 * \pi$
- Với: π : sử dụng hằng số `math.pi`
 r : bán kính hình tròn
- Xây dựng hàm tính chu vi và diện tích hình chữ nhật (sử dụng `lambda`):
 - Chu vi: $(a + b) * 2$
 - Diện tích: $a * b$
 - Với: a : chiều dài hình chữ nhật
 b : chiều rộng hình chữ nhật

✓ Minh họa:

```
Bạn muốn làm gì?
1. Tính chu vi/diện tích hình tròn
2. Tính chu vi/diện tích hình chữ nhật
Mời bạn chọn: 1

Nhập bán kính hình tròn: 6
Chu vi hình tròn: 37.7
Diện tích hình tròn: 113.1
```

```
Bạn muốn làm gì?
1. Tính chu vi/diện tích hình tròn
2. Tính chu vi/diện tích hình chữ nhật
Mời bạn chọn: 2

Nhập chiều dài hình chữ nhật: 20
Nhập chiều rộng hình chữ nhật: 13
Chu vi hình chữ nhật: 66
Diện tích hình chữ nhật: 260
```

8.4. Sử dụng `map()`, `filter()`, `reduce()`

- ✓ **Yêu cầu:** Viết chương trình sử dụng các phương thức `map()`, `filter()`, `reduce()` để thực hiện các yêu cầu sau:
 - Cho 2 danh sách điểm HK1 và HK2:


```
list_diem_hk1 = [8.5, 6.7, 9.0, 8.3, 5.0, 4.0, 6.0, 3.7, 9.5, 5.8]
list_diem_hk2 = [7.0, 7.0, 9.2, 6.1, 4.8, 6.4, 8.5, 5.1, 5.6, 7.0]
```
 - Tính và in ra danh sách điểm trung bình.
 - Xử lý và in ra kết quả 'Đậu' hoặc 'Rớt' tương ứng với điểm trung bình đã tính.
 - In ra danh sách các điểm trên trung bình và dưới trung bình.

- Chuyển các giá trị trong danh sách điểm trung bình thành số nguyên, tìm số nguyên tố trong danh sách số nguyên đó.
- Tính tổng các giá trị số nguyên tố.

✓ **Thuật giải:**

- ❖ Nhập:
 - Cho 2 danh điểm HK1 và HK2
- ❖ Xuất:
 - Theo các yêu cầu trên
- ❖ Xử lý:
 - Sử dụng map(), filter(), reduce() để thực hiện

✓ **Minh họa:**

Danh sách ĐTB: [7.5, 6.9, 9.1, 6.8, 4.9, 5.6, 7.7, 4.6, 6.9, 6.6]
 Danh sách KQ theo ĐTB: ['Đậu', 'Đậu', 'Đậu', 'Đậu', 'Rót', 'Đậu', 'Đậu', 'Rót', 'Đậu', 'Đậu']
 Danh sách điểm trên TB: [7.5, 6.9, 9.1, 6.8, 5.6, 7.7, 6.9, 6.6]
 Danh sách điểm dưới TB: [4.9, 4.6]
 Danh sách số nguyên từ ĐTB: [7, 6, 9, 6, 4, 5, 7, 4, 6, 6]
 Danh sách số nguyên tố: [7, 5, 7]
 Tổng các số nguyên tố: 19

8.5. Kiểm tra ngày hợp lệ

- ✓ **Yêu cầu:** Viết chương trình cho người dùng nhập vào 1 chuỗi theo định dạng dd/MM/yyyy, xây dựng hàm kiểm tra các thành phần ngày, tháng, năm nhập vào có hợp lệ hay không.

✓ **Thuật giải:**

- ❖ Nhập:
 - Chuỗi theo định dạng dd/MM/yyyy
- ❖ Xuất:
 - Kiểm tra ngày, tháng, năm hợp lệ
- ❖ Xử lý:
 - Xây dựng hàm *kiem_tra_ngay_thang_nam(chuoi)*:
 - Năm: là số nguyên có 2 hoặc 4 kí số
 - Tháng: là số nguyên từ 1 đến 12
 - Ngày: căn cứ theo tháng
 - Tháng 1, 3, 5, 7, 8, 10, 12: có 31 ngày
 - Tháng 4, 6, 9, 11: có 30 ngày
 - Tháng 2: Nếu là năm nhuận thì có 29 ngày, ngược lại có 28 ngày.
 - Kết quả trả về kiểu Boolean (True/False)

✓ **Minh họa:**

Nhập chuỗi (dd/MM/yyyy): 30/2/2020
Kết quả: False

Nhập chuỗi (dd/MM/yyyy): 11/7/1992
Kết quả: True

Nhập chuỗi (dd/MM/yyyy): 1/1/1
Kết quả: False

8.6. Viết hàm cho các bài tập đã thực hiện trước đó

- ✓ Giải phương trình bậc nhất
- ✓ Giải phương trình bậc hai
- ✓ Tính diện tích hình tam giác
- ✓ ...

BÀI 9: Module – Package



Mục tiêu chính: Cung cấp cho học viên:

- Khái niệm về Module và Package
- Biết cách tra cứu, sử dụng các thư viện chuẩn của Python
- Biết cách cài đặt, quản lý các package của bên thứ ba thông qua PIP

9.1. Tổ chức bài tập 8

- ✓ **Yêu cầu:** Trong bài tập 8, học viên tạo 1 tập tin có tên **thu_vien_bai_8.py** và di chuyển các hàm của bài tập 8 vào đó. Sử dụng kiến thức đã học để sử dụng hàm.
- ✓ Tổ chức bài tập 8

9.2. Xây dựng ứng dụng đơn giản

- ✓ **Yêu cầu:** Viết chương trình thực hiện các chức năng đơn giản như:
 - Tạo mã QR wifi
 - Tạo mã QR nội dung
 - Download tập tin
 - Đọc văn bản từ file PDF
- ✓ **Thuật giải:**
 - ❖ Xử lý:
 - Cài đặt các thư viện cần thiết:
 - `pip install pyqrcode`
 - `pip install pypng`
 - `pip install requests`
 - `pip install pdfplumber`
 - Tạo thư mục làm việc (thư mục gốc), bên trong 4 thư mục và 1 tập tin `app.py` (chương trình chính):
 - Cung cấp tập tin `libs.py`, đặt tập tin này vào thư mục `libs`
 - Viết chương trình xử lý theo 4 yêu cầu trên

- ✓ **Minh họa:**

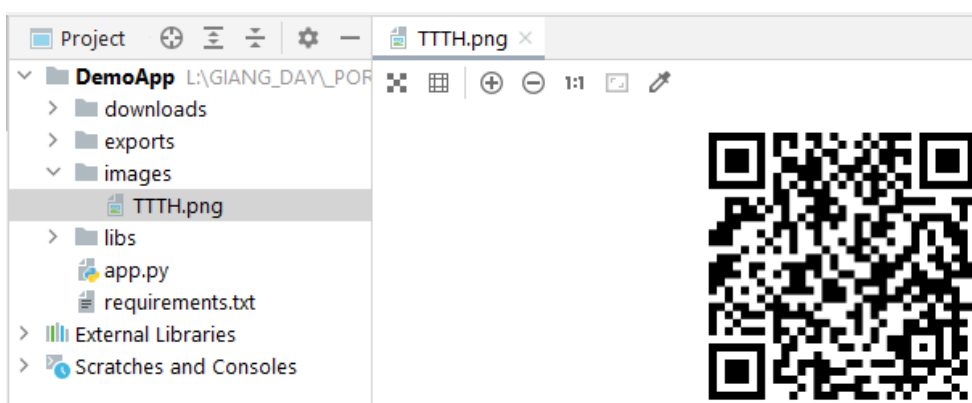
○ Chức năng 1: Tạo mã QR wifi

```
Bạn muốn làm gì?
1. Tạo mã QR wifi
2. Tạo mã QR nội dung
3. Download tập tin
4. Đọc văn bản từ PDF
Mời bạn chọn: 1

Tên wifi (SSID): TTTH
Mật khẩu: ttth2023
Loại bảo mật (WPA hoặc WEP): WPA

Tạo mã QR thành công. Xem tại: images/TTTH.png

Tiếp tục? 1. Có 0. Không
```



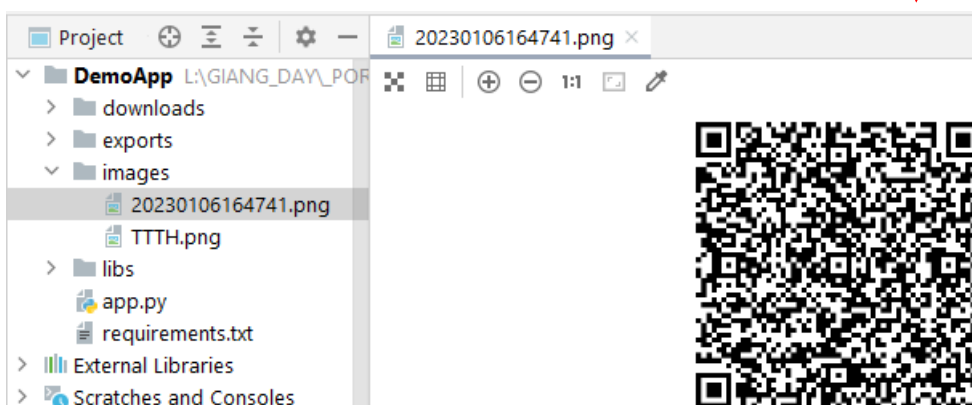
○ Chức năng 2: Tạo mã QR nội dung

```
Bạn muốn làm gì?
1. Tạo mã QR wifi
2. Tạo mã QR nội dung
3. Download tập tin
4. Đọc văn bản từ PDF
Mời bạn chọn: 2

Nhập nội dung: https://csc.edu.vn/data-science-machine-learning

Tạo mã QR thành công. Xem tại: images/20230106164741.png

Tiếp tục? 1. Có 0. Không
```



○ Chức năng 3: Download tập tin

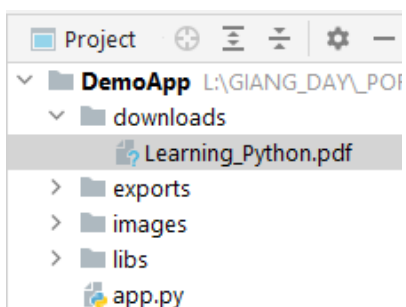
```
Bạn muốn làm gì?
1. Tạo mã QR wifi
2. Tạo mã QR nội dung
3. Download tập tin
4. Đọc văn bản từ PDF
Mời bạn chọn: 3

Nhập link: https://cfm.ehu.es/ricardo/docs/python/Learning\_Python.pdf

Download is starting...
Finished! Total time: 9.61s

Tải tập tin thành công. Xem tại: downloads/Learning_Python.pdf

Tiếp tục? 1. Có 0. Không
```



○ Chức năng 4: Đọc nội dung từ tập tin PDF

```
Bạn muốn làm gì?
1. Tạo mã QR wifi
2. Tạo mã QR nội dung
3. Download tập tin
4. Đọc văn bản từ PDF
Mời bạn chọn: 4

Tập tin PDF: downloads/Learning\_Python.pdf
Extracting page 1/1213
Extracting page 2/1213
Extracting page 3/1213
:
:
Extracting page 1212/1213
Extracting page 1213/1213
Tập tin text: Learning\_Python.txt

Lưu tập tin thành công. Xem tại: exports/Learning_Python.txt

Tiếp tục? 1. Có 0. Không
```

Project		Learning_Python.txt
▼ DemoApp L:\GIANG_DAY\POF	99	IDLE Basics 58
▼ downloads	100	Using IDLE 60
▼ Learning_Python.pdf	101	Advanced IDLE Tools 62
▼ exports	102	Other IDEs 63
▼ Learning_Python.txt	103	Other Launch Options 64
> images	104	viii Table of Contents
> libs		Embedding Calls 64
app.py		
requirements.txt		

BÀI 10: Xử lý ngoại lệ



Mục tiêu chính: Cung cấp cho học viên kiến thức và kỹ năng:

- Xử lý ngoại lệ trong việc xây dựng các xử lý trong chương trình

10.1. Xử lý lỗi xóa phim

- ✓ **Yêu cầu:** Sử dụng khối `try... except...` để xử lý lỗi người dùng chọn thông tin không có trong danh sách.
- ✓ **Thuật giải:**
 - ❖ Nhập:
 - Số thứ tự cần xóa.
 - ❖ Xuất:
 - Nếu số thứ tự có trong danh sách thì thực hiện xóa và thông báo xóa thành công.
 - Ngược lại, thông báo lỗi bằng `except`.
 - ❖ Xử lý: Kiểm tra 2 trường hợp:
 - STT không có trong danh sách => `IndexError: pop index out of range`
 - STT không phải là số => `ValueError: invalid literal for int() with base 10: 's'`
- ✓ **Minh họa:**

- Lỗi STT không có trong danh sách

```
Có 6 phim trong danh sách
STT    TÊN PHIM
1      Dòng chảy của nước
2      Đảo độc đặc
3      Tiểu đội gấu bay
4      Âm lượng hủy diệt
5      Búp bê gọi hồn
6      Tro tàn rực rỡ

Chọn phim cần xóa
Nhập STT: 9

Vui lòng nhập STT trong khoảng từ 1 đến 6
```

- Lỗi STT không phải là số

```
Có 6 phim trong danh sách
STT    TÊN PHIM
1      Dòng chảy của nước
2      Đảo độc đặc
3      Tiểu đội gấu bay
4      Âm lượng hủy diệt
5      Búp bê gọi hồn
6      Tro tàn rực rỡ

Chọn phim cần xóa
Nhập STT: s

Dữ liệu nhập không phải là số
Vui lòng nhập số STT trong khoảng từ 1 đến 6
```

- Xóa thành công

```
Có 6 phim trong danh sách
STT      TÊN PHIM
1        Dòng chảy của nước
2        Đảo độc đắc
3        Tiểu đội gấu bay
4        Âm lượng hủy diệt
5        Búp bê gọi hồn
6        Tro tàn rực rỡ

Chọn phim cần xóa
Nhập STT: 4

Đã xóa phim "Âm lượng hủy diệt" ra khỏi danh sách
```

10.2. Xử lý lỗi chương trình đổi nhiệt độ

- ✓ **Yêu cầu:** Cho người dùng nhập vào nhiệt độ (độ C), thực hiện tính và xuất kết quả sang độ F. Sử dụng khối *try... except...* để xử lý lỗi thông tin nhập.
- ✓ **Thuật giải:**
 - ❖ Nhập:
 - Độ C
 - ❖ Xuất:
 - Nếu dữ liệu nhập hợp lệ thì xuất kết quả chuyển đổi sang độ F.
 - Ngược lại, thông báo lỗi bằng *except*.
 - ❖ Xử lý: Kiểm tra:
 - Dữ liệu nhập không phải là số => `NameError: name 'abc' is not defined`
- ✓ **Minh họa:**
 - Lỗi:

```
Nhập độ C: abc
----- KẾT QUẢ -----
Giá trị nhập không hợp lệ.
Vui lòng nhập số.
```

- Hợp lệ:

```
Nhập độ C: 30
----- KẾT QUẢ -----
30.00 độ C = 86.00 độ F
```

10.3. Xử lý lỗi lưu trữ tập tin

- ✓ **Yêu cầu:** Cho người dùng nhập vào link tập tin cần tải từ Internet, sử dụng khối *try... except...* để xử lý lỗi không tìm thấy đường dẫn lưu trữ.
- ✓ **Thuật giải:**

❖ Nhập:

- Đường link dẫn đến tập tin cần tải.

❖ Xuất:

- Kết quả tải thành công.
- Ngược lại, thông báo lỗi bằng except.

❖ Xử lý: Kiểm tra:

- Đường dẫn lưu trữ tập tin tải về từ Internet:

`FileNotFoundError: [Errno 2] No such file or directory: 'downloads/Learning_Python.pdf'`

✓ **Minh họa:**

Nhập link: https://cfm.ehu.es/ricardo/docs/python/Learning_Python.pdf

Download is starting...

Không tìm thấy đường dẫn lưu trữ: downloads/Learning_Python.pdf

BÀI 11: Tập tin – Thư mục



Mục tiêu chính: Cung cấp cho học viên kiến thức và kỹ năng:

- Làm việc với tập tin text (.txt)
- Làm việc với tập tin csv (.csv)
- Thao tác trên tập tin / thư mục

11.1. Đọc tập tin text và thống kê

- ✓ **Yêu cầu:** Viết chương trình cho người dùng nhập vào đường dẫn đến tập tin text. Thực hiện đọc nội dung và thống kê số lượng dòng, từ và ký tự trong tập tin.
- ✓ **Thuật giải:**
 - ❖ Nhập:
 - Đường dẫn đến tập tin text
 - ❖ Xuất:
 - Nội dung trong tập tin
 - Thống kê số lượng dòng, từ và ký tự
 - ❖ Xử lý:
 - Xây dựng hàm, mỗi hàm xử lý một chức năng riêng biệt
 - Ví dụ: `doc_tap_tin(<cac_tham_so>)`, `thong_ke(<cac_tham_so>)`, ...
- ✓ **Minh họa:**

```
Nhập tên tập tin: du_Lieu/HumptyDumpty.txt
----- NỘI DUNG -----
Humpty Dumpty sat on a wall,
Humpty Dumpty had a great fall.
All the king's horses and all the king's men
Couldn't put Humpty together again.
----- THỐNG KÊ -----
- Số dòng: 4
- Số từ: 26
- Số ký tự: 141
```

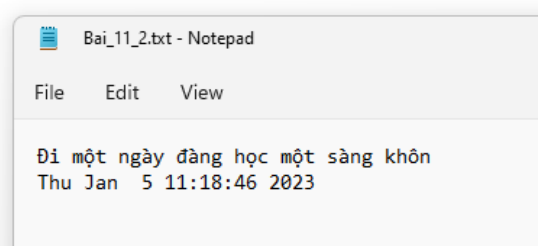
11.2. Ghi tập tin text (1 dòng)

- ✓ **Yêu cầu:** Viết chương trình cho người dùng nhập vào nội dung. Thực hiện lưu thông tin vào đường dẫn chỉ định.
- ✓ **Thuật giải:**
 - ❖ Nhập:
 - Nội dung cần lưu

- ❖ Xuất:
 - Lưu thông tin vào tập tin text
- ❖ Xử lý:
 - Lưu nội dung đã nhập vào tập tin text
 - Lưu thêm vào dòng cuối cùng thông tin thời gian hiện tại tại thời điểm lưu

✓ **Minh họa:**

Nhập nội dung: *Đi một ngày đàng học một sàng khôn*
 Đã lưu nội dung tại "du_lieu/Bai_11_2.txt"



11.3. Ghi tập tin text (nhiều dòng)

- ✓ **Yêu cầu:** Viết chương trình cho người dùng nhập vào danh sách sinh viên cho đến khi không muốn nhập nữa. Thực hiện lưu danh sách sinh viên đã nhập vào tập tin text.
- ✓ **Thuật giải:**
 - ❖ Nhập:
 - Thông tin sinh viên
 - ❖ Xuất:
 - Lưu thông tin vào tập tin text
 - ❖ Xử lý:
 - Lưu danh sách sinh viên đã nhập vào tập tin text

✓ **Minh họa:**

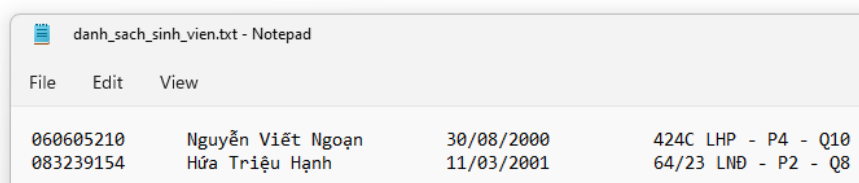
Mã số SV: *060605210*
 Họ tên: *Nguyễn Viết Ngoạn*
 Ngày sinh: *30/08/2000*
 Địa chỉ: *424C LHP - P4 - Q10*

Tiếp tục? 1. Có 0. Không *1*

Mã số SV: *083239154*
 Họ tên: *Hứa Triệu Hạnh*
 Ngày sinh: *11/03/2001*
 Địa chỉ: *64/23 LND - P2 - Q8*

Tiếp tục? 1. Có 0. Không *0*

Đã lưu thông tin tại "du_lieu/danh_sach_sinh_vien.txt"

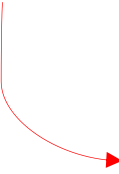


11.4. Tính điểm trung bình

- ✓ **Yêu cầu:** Viết chương trình tính điểm trung bình của danh sách học sinh đọc từ file csv.
- ✓ **Thuật giải:**
 - ❖ Xử lý:
 - Đọc danh sách học sinh từ tập tin csv đã cho (hoc_sinh.csv). Thực hiện tính điểm trung bình từ 3 cột điểm đang có, sau đó tạo thêm 1 cột điểm thứ 4 có tên "Điểm trung bình".
 - Xuất kết quả sau khi tính toán ra 1 tập tin csv khác.

✓ **Minh họa:**

	Họ tên ▼	Môn học 1 ▼	Môn học 2 ▼	Môn học 3 ▼
	Lê Văn Bảo	10	8.5	9
	Trần Minh Ngọc	6.5	7	10
	Lê Thị Minh Tâm	8	5.5	9.5
	Nguyễn Kim Tuyến	6	7.5	9
	Ngô Văn Sáng	5	9	8.5



	Họ tên ▼	Môn học 1 ▼	Môn học 2 ▼	Môn học 3 ▼	Điểm trung bình ▼
	Lê Văn Bảo	10	8.5	9	9.2
	Trần Minh Ngọc	6.5	7	10	7.8
	Lê Thị Minh Tâm	8	5.5	9.5	7.7
	Nguyễn Kim Tuyến	6	7.5	9	7.5
	Ngô Văn Sáng	5	9	8.5	7.5

11.5. Đăng nhập

- ✓ **Yêu cầu:** Viết chương trình cho phép người dùng thực hiện đăng nhập. Xuất ra màn hình trạng thái đăng nhập (Thành công/Thất bại).
- ✓ **Thuật giải:**
 - ❖ Nhập:
 - Tên đăng nhập
 - Mật khẩu
 - ❖ Xuất:
 - Thông báo trạng thái sau khi nhập thông tin đăng nhập.
 - Lưu thông tin nhật ký nếu đăng nhập thành công.
 - ❖ Xử lý:
 - Cho người dùng nhập *Tên đăng nhập* vào *Mật khẩu*
 - So sánh dữ liệu nhập với dữ liệu đọc ra từ tập tin csv đã cho (*nhan_vien.csv*). Nếu thông tin nhập đúng với *Tên đăng nhập* và *Mật khẩu* có trong dữ liệu csv:
 - Xuất ra câu chào. VD: *Xin chào, Nhân viên 1*
 - Lưu lại thông tin nhân viên vừa thực hiện đăng nhập (lưu nhật ký).
 - Ngược lại, nếu thông tin nhập không khớp với dữ liệu trong tập tin csv thì thông báo "*Đăng nhập thất bại*"

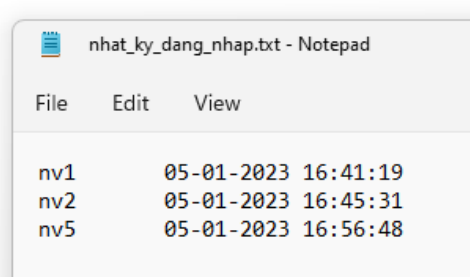
✓ **Minh họa:**

- Dữ liệu cung cấp

	Họ tên ▼	Tên đăng nhập ▼	Mật khẩu ▼	Email ▼
	Nhân Viên 1	nv1	nv01	nv1@csc.hcmus.edu.vn
	Nhân Viên 2	nv2	nv02	nv2@csc.hcmus.edu.vn
	Nhân Viên 3	nv3	nv03	nv3@csc.hcmus.edu.vn
	Nhân Viên 4	nv4	nv04	nv4@csc.hcmus.edu.vn
	Nhân Viên 5	nv5	nv05	nv5@csc.hcmus.edu.vn

- Đăng nhập thành công:

Tên đăng nhập: **nv5**
Mật khẩu: **nv05**
Xin chào, Nhân Viên 5



- Đăng nhập thất bại:

Tên đăng nhập: **nv6**
Mật khẩu: **nv06**
Đăng nhập thất bại

11.6. Tra cứu Zipcode

✓ **Yêu cầu:** Viết chương trình cho người dùng tra cứu Zipcode từ:

- Tên Tỉnh/Thành phố
- Mã Zipcode

✓ **Thuật giải:**

- ❖ Nhập:
 - Tên Tỉnh/Thành phố hoặc Mã Zipcode
- ❖ Xuất:
 - Danh sách kết quả theo thông tin nhập
- ❖ Xử lý:
 - Tập tin dữ liệu cung cấp: *zipcode.csv*
 - Xây dựng chương trình dưới dạng thực đơn lựa chọn. Có 2 lựa chọn:
 - Tra cứu từ Tên Tỉnh/Thành phố
 - Tra cứu từ Mã Zipcode
 - Xuất kết quả tra cứu ra màn hình.

✓ **Minh họa:**

Bạn muốn tra cứu như theo:

1. Tên Tỉnh/Thành phố

2. ZipCode

Mời bạn chọn: **1**

Tên Tỉnh/Thành phố: **an**

Tìm thấy kết quả với từ khóa "an":

TỈNH/THÀNH PHỐ	MÃ ZIPCODE

An Giang	880000
Long An	850000
Nghệ An	460000, 470000

Tiếp tục? 1. Có 0. Không

MỤC LỤC

BÀI 1: Tổng quan Lập trình Python	2
1.1. Thực hành cài đặt môi trường Python	2
1.2. Thực hành cài đặt các IDE (Integrated Development Environment)	2
BÀI 2: Biến và các kiểu dữ liệu cơ sở	3
2.1. Đố vui	3
2.2. Tính tiền hàng	3
2.3. Đổi nhiệt độ từ độ C sang độ F	3
2.4. Xử lý chuỗi	4
2.5. Tính tiền lãi gửi tiết kiệm	5
BÀI 3: Toán tử	6
3.1. Tính trung bình cộng	6
3.2. Tính giá trị biểu thức	6
3.3. Tính toán mệnh giá tiền	6
3.4. Tính kết quả 1	7
3.5. Tính kết quả 2	8
3.6. Tính kết quả 3	8
3.7. Tính kết quả 4	8
3.8. Tính kết quả 5	9
BÀI 4: Cấu trúc điều kiện	10
4.1. Tìm giá trị tuyệt đối	10
4.2. Tìm giá trị lớn nhất – nhỏ nhất	10
4.3. Giải phương trình bậc nhất	10
4.4. Xét năm nhuận	11
4.5. Tính cước taxi	11
4.6. Tính tiền điện	12

4.7.	Tính tiền thuê phòng resort	13
4.8.	Tính thuế thu nhập cá nhân.....	13
BÀI 5: Cấu trúc lặp		15
5.1.	Count down	15
5.2.	Tính tổng n số nguyên nhập vào.....	15
5.3.	Xuất chuỗi	16
5.4.	Kiểm tra số nguyên tố	16
5.5.	Tính S	17
5.6.	Tính A	17
5.7.	Tính giá trị biểu thức.....	17
5.8.	In bảng cửu chương.....	18
BÀI 6: Number – String – Datetime		19
6.1.	Tính bán kính hình tròn	19
6.2.	Giải phương trình bậc hai	19
6.3.	Tính diện tích tam giác	20
6.4.	Tìm kiếm thức uống	21
6.5.	Tìm số giá trị số nguyên và tính tổng	21
6.6.	Làm việc với calendar.....	22
6.7.	Làm việc với datetime	22
BÀI 7: List – Tuple – Dictionary – Set		24
7.1.	Danh sách thú	24
7.2.	Tìm từ duy nhất.....	24
7.3.	List numbers.....	24
7.4.	Quản lý phim.....	26
7.5.	Tuple string	28
7.6.	Từ điển	29

7.7. Set number	32
BÀI 8: Phương thức / Hàm.....	34
8.1. Tính năm âm lịch	34
8.2. Tính chỉ số BMI.....	34
8.3. Tính chu vi, diện tích.....	35
8.4. Sử dụng map(), filter(), reduce()	36
8.5. Kiểm tra ngày hợp lệ.....	37
8.6. Viết hàm cho các bài tập đã thực hiện trước đó	38
BÀI 9: Module – Package	39
9.1. Tổ chức bài tập 8.....	39
9.2. Xây dựng ứng dụng đơn giản	39
BÀI 10: Xử lý ngoại lệ.....	43
10.1. Xử lý lỗi xóa phim.....	43
10.2. Xử lý lỗi chương trình đổi nhiệt độ.....	44
10.3. Xử lý lỗi lưu trữ tập tin.....	44
BÀI 11: Tập tin – Thư mục	46
11.1. Đọc tập tin text và thống kê.....	46
11.2. Ghi tập tin text (1 dòng)	46
11.3. Ghi tập tin text (nhiều dòng)	47
11.4. Tính điểm trung bình	48
11.5. Đăng nhập.....	48
11.6. Tra cứu Zipcode	49