

# Các thuật toán thông minh nhân tạo và ứng dụng

## Bài thực hành 1

### Ôn tập về Python

#### 1. Môi trường:

Python3 (Download [link](#))

Anaconda (Download [link](#))

#### 2. Bài tập

*Nhóm bài 1: bài tập cơ bản.*

##### Bài 1.1

Viết chương trình in ra màn hình dòng chữ Hello World

##### Bài 1.2

Viết chương trình nhập nhiệt độ F và chuyển sang nhiệt độ C theo công thức:

$C = 5 * (F - 32) / 9$ , với C: nhiệt độ C; F: nhiệt độ F

##### Bài 1.3

Viết chương trình nhập vào 2 số nguyên, in ra màn hình tổng bình phương của 2 số nguyên đó.

Công thức:  $S = a * a + b * b$

##### Bài 1.4

Viết chương trình nhập vào 2 số nguyên, in ra màn hình hiệu bình phương của 2 số nguyên đó.

Công thức:  $S = a * a - b * b$

*Nhóm bài 2: câu lệnh điều kiện:*

##### Bài 2.1

Viết chương trình nhập vào hai số a, b; giải và biện luận phương trình  $ax + b = 0$

Gợi ý: sử dụng hàm int (string) để chuyển dữ liệu từ định dạng chuỗi sang định dạng số.

Ví dụ: `a = int(input('Nhập a: '))`

**Bài 2.2** Viết chương trình nhập vào ba số a, b, c; giải và biện luận phương trình  $ax^2 + bx + c = 0$

### ***Nhóm bài tập 3: Hàm***

**Bài 3.1** Viết hàm với tham số truyền vào là năm sinh, sử dụng hàm vừa cài đặt, nhập vào năm sinh và in ra tuổi:

Thí dụ nhập 1984 in ra: Bạn sinh năm 1984, vậy bạn 19 tuổi.

**Bài 3.2** Viết hàm với tham số truyền vào là nhiệt độ F, trả về kết quả nhiệt độ C theo công thức. Sử dụng hàm vừa cài đặt, nhập vào độ F và in ra màn hình độ C.

$C = 5 * (F - 32) / 9$ , với C: nhiệt độ C; F: nhiệt độ F

**Bài 3.3** Viết hàm với tham số truyền vào là một tháng và trả về mùa tương ứng trong năm. Sử dụng hàm vừa cài đặt, nhập vào một tháng và in ra màn hình mùa trong năm.

Ví dụ: Người dùng nhập vào tháng 2, in ra màn hình là mùa Xuân.

Từ tháng 1 đến tháng 3: Mùa Xuân

Từ tháng 4 đến tháng 6: Mùa Hạ

Từ tháng 7 đến tháng 9: Mùa Thu

Từ tháng 10 đến tháng 12: Mùa Đông

**Bài 3.4** Viết hàm tìm số lớn nhất của hai số nguyên a và b; sử dụng hàm vừa cài đặt, nhập vào 3 số nguyên a, b, c và tìm số lớn nhất trong 3 số đó.

Ví dụ: Người dùng nhập vào ba số 5, 9, 4. In ra màn hình số lớn nhất là 9.

**Bài 3.5** Viết hàm tính diện tích hình tròn với tham số truyền vào là bán kính; sử dụng hàm vừa cài đặt, nhập vào bán kính và in ra màn hình diện tích hình tròn.

Công thức tính diện tích hình tròn:

$$S = PI * R * R$$

Với  $PI = \text{math.pi}$  (sử dụng thư viện math bằng cách gọi lệnh `import math` ở đầu script)

### ***Nhóm bài tập 4: Vòng lặp***

**Bài 4.1:** Nhập vào một số, tìm tổng các số lẻ từ 1 đến n

Ví dụ Nhập n=20

In ra tổng số lẻ Sum= 1+2+3+5+ ... + 19

**Bài 4.2** Tính tổng các số nguyên tố từ 1 đến 100. Số nguyên tố là số tự nhiên CHỈ chia hết cho 1 và chính nó.

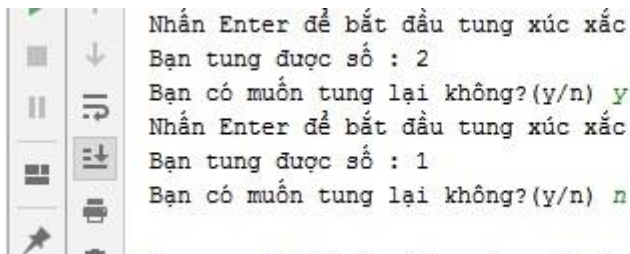
Ví dụ: Số 3 là số nguyên tố, số 9 không phải là số nguyên tố

Sum= 1+3+5+7+11+13+17+ ..... + 97

**Bài 4.3:** Viết chương trình tung xúc xắc. Đưa ra kết quả ngẫu nhiên (từ 1 đến 6) cho người dùng bằng cách sử dụng vòng lặp while. Vòng lặp sẽ tiếp tục cho đến khi người dùng ngừng lại.

**Gợi ý:** num = random.randint(1,6)

Ví dụ: Nhấn Enter để bắt đầu tung xúc xắc.



**Bài 4.4** Nhập vào hai số x và n. Tính dãy số

$$S(x,n) = x + \frac{x^2}{2!} + \frac{x^3}{3!} + \dots + \frac{x^n}{n!}$$

Thí dụ nhập x=2, n=3

In ra kết quả của  $S(2,3) = 2 + \frac{2^2}{2!} + \frac{2^3}{3!}$

**Nhóm bài tập 5: Danh sách (List)**

### Bài 5.1

Viết chương trình tính tổng các phần tử trong một danh sách gồm các số thực

### Bài 5.2

Viết chương trình tính tổng các số chẵn trong một danh sách gồm các số nguyên

### Bài 5.3

Viết chương trình tìm phần tử lớn nhất trong danh sách gồm các số thực

## **Bài 5.4**

Viết chương trình kiểm tra xem 2 hai danh sách có phần tử trùng nhau hay không.

## **Bài 5.5**

Viết chương trình xóa các phần tử trùng trong danh sách

## ***Nhóm bài tập 6: Chuỗi***

### **Bài 6.1**

Viết chương trình để tính chiều dài 1 chuỗi.

### **Bài 6.2**

Viết chương trình để lấy về một chuỗi là chuỗi tạo thành bởi hai ký tự đầu tiên và hai ký tự cuối của một chuỗi cho trước. Nếu chuỗi cho trước này có chiều dài nhỏ hơn hai, trả về chuỗi rỗng:  
Ví dụ: 'w3resource' → 'w3ce', 'w3' → 'w3w3', 'w' → Chuỗi rỗng.

### **Bài 6.3**

Viết chương trình thêm 'ing' vào cuối một chuỗi cho trước. Nếu chuỗi cho trước đã kết thúc bởi 'ing' thêm 'ly' vào cuối. Nếu chuỗi cho trước có độ dài nhỏ hơn 3, để nguyên không thêm gì.

Ví dụ: 'abc' → 'abcing', 'string' → 'stringly'

### **Bài 6.4**

Viết chương trình để xóa ký tự thứ **n** từ một chuỗi không rỗng.

### **Bài 6.5**

Viết chương trình để xóa các ký tự ở vị trí **lẻ**.

## ***Nhóm bài tập 7: Dictionary***

**Bài 7.1** Viết một đoạn lệnh cho phép người dùng thêm một khóa vào từ điển. Trong đó khóa và giá trị do người dùng chỉ định:

Ví dụ:

Từ điển trước khi thêm: {0: 10, 1: 20}

Phần tử cần thêm: key = 2, value = 30

Từ điển sau khi thêm (không cần đúng thứ tự như ví dụ):

{0: 10, 1: 20, 2: 30}

**Bài 7.2** Viết đoạn lệnh để kiểm tra xem một khóa có nằm trong từ điển không. Nếu có, xuất ra “Tồn tại giá trị trong từ điển với giá trị là v” với v là giá trị có trong từ điển. Nếu không, xuất ra là “Không có khóa k trong từ điển” với k là khóa đã nhập.

Ví dụ: {"a": 1, "b": 2}

Tìm “a”

Tồn tại giá trị trong từ điển với giá trị là 1

**Bài 7.3** Viết chương trình xóa một phần tử với khóa cho trước khỏi từ điển. Xuất lại từ điển sau khi xóa.

Ví dụ: {"a": 1, "b": 2, "c": 3.5, "d": "hello"}

Xóa phần tử với khóa: “c”

Từ điển sau khi xóa (không cần đúng thứ tự như ví dụ):

{"a": 1, "b": 2, "d": "hello"}

**Bài 7.4** Viết chương trình kiểm tra xem từ điển có rỗng không. Nếu có, in ra là “Từ điển rỗng”. Nếu không, in ra là “Từ điển không rỗng”.

Ví dụ: {"a": 1, "b": 2, "c": 3.5, "d": "hello"}

Từ điển không rỗng

**Bài 7.5** Viết chương trình tính tổng các giá trị là số nguyên có trong từ điển.

Ví dụ: {"a": 1, "b": 2, "c": 3.5, "d": "hello"}

Tổng là 3

**Nhóm bài tập 8: Bộ (Tuple)**

**Bài 8.1** Viết chương trình tạo ra một tuple từ dữ liệu người dùng nhập vào

Ví dụ:

Các giá trị người dùng nhập:

1

2

3

4

Bộ được tạo là: (1,2,3,4)

**Bài 8.2** Viết chương trình thêm một phần tử vào cuối của một bộ cho trước.

Ví dụ: ("a", 2, "b", "c")

Thêm: 2.5

Bộ sau khi thêm: ("a", 2, "b", "c", 2.5)

**Bài 8.3** Viết chương trình xóa phần tử ở cuối khỏi một bộ cho trước.

Ví dụ: ("a", 2, "b", "c")

Bộ sau khi xóa: ("a", 2, "b")

**Bài 8.4** Viết chương trình thêm phần tử vào vị trí bất kì trong bộ cho trước. Giá trị và vị trí do người dùng chỉ định. Vị trí được tính từ 0.

Ví dụ: ("a", 2, "b", "c")

Thêm 8 vào vị trí 2.

Bộ sau khi thêm: ("a", 2, 8, "b", "c")

**Bài 8.5** Viết chương trình xóa một phần tử ở vị trí bất kì trong bộ cho trước. Vị trí do người dùng chỉ định và được tính từ 0.

Ví dụ: ("a", 2, "b", "c")

Xóa tại vị trí 1.

Bộ sau khi xóa: ("a", "b", "c")

**Bài 8.6** Viết chương trình xóa các phần tử trùng nhau trong bộ cho trước (nếu có phần tử trùng thì chỉ giữ lại 1 phần tử đầu tiên gặp được).

Ví dụ: (“a”, 2, “b”, 2, “a”, “c”, “a”)

Bộ sau khi xóa: (“a”, 2, “b”, “c”)

**Bài 8.7** Viết chương trình tính tổng các số có trong bộ (số nguyên, số thực).

Ví dụ: (“a”, 3, “b”, 2.5, “a”, 8.5, “a”)

Tổng là: 14

### **3. Tài liệu tham khảo**

[1] [https://www.w3schools.com/python/python\\_intro.asp](https://www.w3schools.com/python/python_intro.asp)

[2] <https://www.tutorialspoint.com/python/index.htm>

[3] <https://www.learnpython.org/>

[4] <https://www.guru99.com/python-tutorials.html>