

LAB GUIDE LIST + TUPLE

I. Conditions and Loops:	1
II. Lists:	1

I. Conditions and Loops:

- Viết chương trình tìm tất cả các số có 3 chữ số sao cho tổng lập phương của các chữ số bằng chính số đó (gọi là các số Armstrong). Ví dụ: Số 153 là số Armstrong vì $153 = 1^3 + 5^3 + 3^3$.
- Một người cần rút một số tiền T từ ngân hàng và muốn tổng số tờ ít nhất. Cho biết có các loại tiền mệnh giá 100, 20, 5 và 1. Viết một chương trình cho phép nhập từ bàn phím số tiền T và in ra số tờ mỗi loại mệnh giá; tổng số tờ nhận được.

II. Lists:

// Nhập danh sách đầu vào bằng console → sử dụng phương thức `str.split()` để chuyển chuỗi nhập vào thành danh sách.

- Viết chương trình python cho phép nhập vào 1 dãy số từ bàn phím, cách nhau bởi dấu phẩy. Hãy tính tổng của các số chẵn, tích của các số lẻ và tính trung bình cộng của dãy số này.
- Viết chương trình python tìm và in ra màn hình tất cả các số hoàn hảo không vượt quá số n cho trước nhập từ bàn phím. Số hoàn hảo là số bằng tổng các ước số của nó không kể ước số là chính số đó. Ví dụ số 6 là số hoàn hảo vì $6=1+2+3$.
- Viết một chương trình Python để đếm số chuỗi thỏa mãn điều kiện: độ dài chuỗi lớn hơn một giá trị đầu vào n, ký tự đầu tiên và cuối cùng là giống nhau từ một danh sách cho các chuỗi. Ví dụ: cho list đầu vào: ['abc', 'xyz', 'aba', '1221', 'ii', 'ii2', '5yhy5'] và số $n = 2 \rightarrow$ Kết quả cho ra 3.
- Viết chương trình loại bỏ các phần tử có giá trị giống nhau trong một list (list này được nhập vào từ bàn phím). Ví dụ: Nhập list đầu vào là [11, 14, 16, 16, 18, 19, 16, 11, 11, 14] thì kết quả thu được là [18, 19].
- Viết một chương trình python sắp xếp 1 list các tuple là số nguyên theo giá trị của phần tử cuối cùng. Ví dụ: Đầu vào là list [(1, 2), (2, 6), (3, 1, 5)]. Đầu ra là: [(2, 6), (3, 1, 5), (1, 2)].