

## LẬP TRÌNH VỚI PYTHON INT13162

### CÂU 1: TÁCH CHUỖI

Nhập chuỗi S có độ dài không quá 100. Một từ được định nghĩa là một dãy ký tự không có khoảng trống. Hãy tách chuỗi S thành các từ, tìm ra từ có độ dài lớn nhất và nhỏ nhất của S, kèm theo đó là độ dài tương ứng.

#### Input:

- Chỉ có một dòng ghi xâu S (độ dài không quá 100).

#### Output:

- Từ có độ dài lớn nhất và nhỏ nhất, kèm theo là độ dài tương ứng của chúng. Kết quả của mỗi từ được viết trên một dòng. Nếu có nhiều từ có độ dài bằng nhau, hãy sắp xếp kết quả theo chiều tăng dần của bảng chữ cái.

#### Ví dụ:

Input	Output
Hom nay thi cuoi ky mon Lap trinh Python	Python 6 ky
Hom nay troi mua	troi 4 Hom 3 mua 3 nay 3

## CÂU 2: HỆ SỐ JACCARD

Hệ số Jaccard là một đại lượng dùng để đo mức độ tương tự giữa hai chuỗi bất kỳ. Cho hai chuỗi A và B, hệ số Jaccard giữa A và B, được ký hiệu là  $J(A, B)$  và được xác định theo công thức sau:

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A| + |B| - |A \cap B|}$$

Cho hai chuỗi A và B, Tính độ đo tương đồng Jaccard giữa hai chuỗi.

Lưu ý:

- Nếu A và B rỗng, thì  $J(A, B)=1$
- $J(A, B)$  nằm trong đoạn  $[0, 1]$
- Lưu ý các kí tự trong chuỗi để tính tương đồng không phân biệt chữ hoa thường, không bao gồm các ký tự như khoảng trắng, dấu chấm (.), dấu phẩy (,) và dấu hỏi chấm (?).

### Input:

Hai chuỗi A và B, mỗi chuỗi được viết trên một dòng.

### Output:

Hệ số Jaccard giữa A và B. Kết quả được in ra với 2 chữ số sau dấu phẩy.

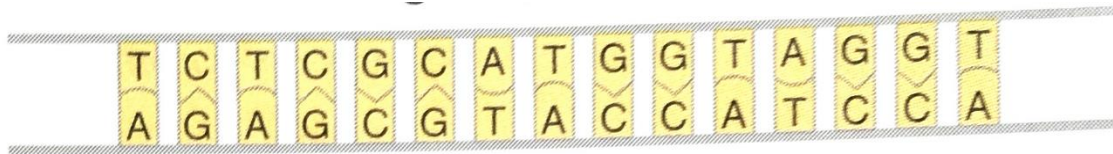
### Ví dụ:

Input	Output
hom nay thi lap trinh python	0.91
Lap trinh Python.	
Hom qua troi mua.	0.50
Hom qua mua, ngay mai co nang khong?	

### CÂU 3: COVID-19 OMICRON SUBSTITUTION VARIANT

Biến thể Omicron của virus SARS-CoV-2 được phát hiện lần đầu tiên ở Nam Phi vào ngày 24 tháng 11 năm 2021 với khả năng hoạt hóa rất mạnh. Biết rằng, cứ 3 ngày trong quần thể virus xảy ra một đột biến điểm (Nucleotid/ base).

Đây là thuật ngữ của di truyền học phân tử, trong tiếng nước ngoài gọi là Point Mutation (tiếng Anh) dùng để chỉ một loại đột biến gen chỉ xảy ra ở một “điểm” trên phân tử di truyền DNA. “Điểm” này tương ứng với một cặp Nucleotide duy nhất.

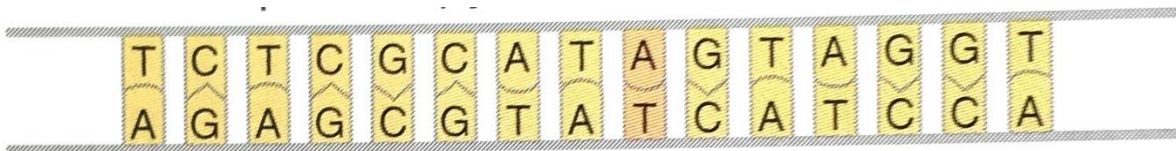


*Hình 1: Bộ gen ban đầu của Virus*

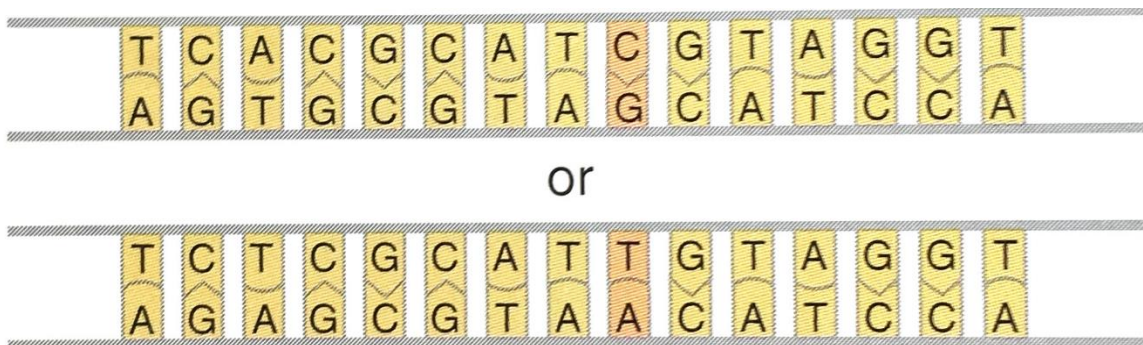
Có ba loại đột biến điểm – Point Mutation là:

- Đột biến thay thế - Substitution
- Đột biến xóa – Deletion
- Đột biến thêm – Insertion

#### **Đột biến thay thế - Substitution:**



Thay thế một cặp số AT thành GC hoặc thay đổi thứ tự AT thành TA và GC thành CG:



Biết Bộ gen (hay hệ gen) của virus có chiều dài (kích thước) làm tròn là 30.000 Nucleotides / Bases; các đột biến hoàn toàn ngẫu nhiên và với xác suất là như nhau. Giả sử tất cả các đột biến điểm mới của biến thể Omicron đều là **đột biến thay thế Substitution** và thay thế duy nhất 1 cặp.

Gợi ý:

Một đột biến có thể xảy ra tại 1 cặp bất kỳ trên 30.000 cặp Nucleotides nên xác suất 2 đột biến giống nhau là  $1/30.000 \sim 0$  coi như các đột biến sinh ra là khác nhau.

- Sau 3 ngày biến thể Omicron ban đầu Om0 trên toàn bộ quần thể sẽ tạo ra một đột biến điểm là Om01 lúc này ta có 1 **đột biến điểm kiểu Substitution**.
- Sau 3 ngày tiếp theo quần thể Omicron Om0 sẽ tạo ra thêm 1 biến thể mới là Om02 và quần thể Om01 sẽ tạo ra thêm 1 đột biến điểm là Om012 vậy lúc này ta sẽ có 3 đột biến điểm là Om01, Om02 và Om012.
- .....

Viết chương trình tính số lượng **đột biến điểm kiểu thay thế Substitution** mà biến thể Omicron đã có sau ngày 24/11/2021.

**Input:**

Dòng đầu tiên số nguyên N ( $1 \leq N \leq 10$ ) là số tập test.

N dòng tiếp theo, mỗi dòng gồm một số nguyên n có giá trị không vượt quá 365 ( $0 < n < 366$ ) là số ngày tính từ 24/11/2021 thời điểm biến thể Omicron bắt đầu xuất hiện.

**Output**

In ra một số nguyên là số lượng đột biến điểm của biến thể Omicron.

Trường hợp nếu số nguyên N vượt quá 10 hoặc nhỏ hơn 0 in ra **INVALID INPUT**

Bộ test nào có giá trị lớn hơn 365 in ra dòng đó là **INVALID INPUT**

**Ví dụ:**

<b>Input:</b>	<b>Output:</b>
5	3
7	INVALID INPUT

380	63
18	137438953471
113	1023
30	

#### **CÂU 4: PHẦN DƯ LỚN NHẤT**

Cho  $K$  danh sách, danh sách thứ  $i$  thuộc  $K$  có  $N_i$  phần tử. Cho hàm  $f(X) = X^2$ . Chọn từ mỗi danh sách một phần tử sao cho giá trị của biểu thức sau là lớn nhất:

$$S = (f(X_1) + f(X_2) + \dots + f(X_k)) \% M$$

Trong đó  $X_i$  được chọn từ danh sách thứ  $i$ .  $\%$  là phép chia lấy phần dư.

**Lưu ý:** Cần lấy chính xác một phần tử từ mỗi danh sách, không nhất thiết phải lấy phần tử lớn nhất.

**Điều kiện:**

- $1 \leq K \leq 7$
- $1 \leq M \leq 1000$
- $1 \leq N_i \leq 7$

**Input:**

- Dòng đầu là  $K$  danh sách và số  $M$ , cách nhau bởi khoảng trắng
- $K$  dòng tiếp theo là các danh sách.

**Output:**

- Giá trị  $S_{max}$
- Nếu các giá trị  $K, M, N_i$  không thỏa mãn điều kiện, in ra **INVALID INPUT**

**Ví dụ:**

Input	Output
3 1000	206
2 5 4	

3 7 8 9	
5 5 7 8 9 10	

**Giải thích:**

Lấy phần tử 5 từ danh sách thứ nhất, phần tử 9 từ danh sách thứ hai và phần tử 10 từ danh sách thứ ba sẽ thu được giá trị  $S_{max}$ :

$$S_{max} = (5^2 + 9^2 + 10^2) \% 1000 = 206$$

### **CÂU 5: TỔNG TÍCH LŨY**

Cho một danh sách số gồm N phần tử. Viết chương trình tính tổng tích lũy của một danh sách, nghĩa là, sẽ tạo ra một danh sách mới, trong đó phần tử thứ  $i$  là tổng của  $i+1$  phần tử đầu tiên từ danh sách ban đầu.

Ví dụ: Một danh sách có 3 phần tử: [1, 2, 3] => danh sách tổng tích lũy mới là: [1, 3, 6]

In ra màn hình tích và tổng của các phần tử trong danh sách mới đó.

**Input:**

- Dòng đầu tiên là số phần tử của mảng N.
- Dòng thứ hai là N phần tử của mảng, mỗi phần tử cách nhau bởi khoảng trắng.

**Output:**

- Tổng và tích của danh sách tổng tích lũy, viết trên cùng một dòng.

**Ví dụ:**

Input	Output
3 1 2 3	10 18

### CÂU 6: DANH SÁCH LỒNG NHAU

Tên và điểm thi môn Lập trình Python của mỗi sinh viên được lưu vào một danh sách, lưu trữ chúng trong một danh sách lồng nhau. Với một lớp có N ( $N \leq 70$ ) sinh viên, tìm và in ra tên của K ( $K \leq 5$ ) sinh viên có điểm thi cao nhất. Nếu có nhiều sinh viên bằng điểm nhau, hãy sắp xếp tên theo chiều tăng dần của bảng chữ cái.

#### Input:

- Dòng đầu tiên là giá trị N và K, cách nhau bởi một khoảng trắng.
- N dòng tiếp theo là tên và điểm thi của sinh viên

#### Output:

- Tên của K sinh viên có điểm cao nhất, được viết trên một dòng duy nhất mỗi tên cách nhau bởi một khoảng trắng.
- Nếu giá trị N và K không thỏa mãn điều kiện, in ra thông báo: **INVALID INPUT**

#### Ví dụ:

Input	Ouput
10 3	
Hung 6	
Long 7	
Giang 8	
Linh 5	
Tuan 8	Ngan Hoa Khanh
Hoa 9	
Mai 5	
Ngoc 4	
Khanh 9	

Ngan 10	
---------	--

### CÂU 7:

Cho một từ điển gồm K phần tử. Viết chương trình tính tổng và tích của các phần tử có giá trị (value) là số trong từ điển đã cho.

#### Input:

- Dòng đầu tiên là giá trị K ( $K \leq 10$ )
- K dòng tiếp theo là các phần tử của từ điển, mỗi phần tử viết trên một dòng, lần lượt là key và value tương ứng.

#### Output:

- Tổng và tích được viết trên cùng một dòng, cách nhau bởi khoảng trắng.
- Nếu K không thỏa mãn điều kiện thì in ra thông báo **INVALID INPUT**

#### Ví dụ:

Input	Output
4 a 9 b 5 c abc d 1	15 45



CÂU 8: ANNUAL TAX

PHIẾU THU NHẬP THÁNG 10/2021

HỌ VÀ TÊN	Nguyễn Văn A	EMAIL	Nva@abc.com.vn
Chức danh	DEV	Bộ phận	IT

NGÀY CÔNG CHUẨN	22	CÁC KHOẢN KHẤU TRỪ (2)	000.000
Ngày công chính thức	22	BHXX (8%*LCB)	000.000
Ngày công thử việc	0	BHYT (1,5%*LCB)	00.000
THU NHẬP THỰC TẾ (1)	00.000.000	BHTN (1%*LCB)	00.000
Lương cơ bản chính thức (LCB)	0.000.000	Đoàn phí (1%LCB)	00.000
Thưởng đánh giá tháng chính thức (LT)	00.000.000	Thuế TNCN	00.000
Lương cơ bản thử việc	0		
Thưởng đánh giá tháng thử việc	0		
Trợ cấp 1 (Không chịu thuế TNCN)			
Trợ cấp 2 (Không chịu thuế TNCN)			
THỰC LĨNH (3) = (1) - (2)	00.000.000		

Trên đây là Phiếu thu nhập của một người trong một tháng bao gồm:

- Thu nhập thực tế (1)
- Các khoản khấu trừ (2)
- Thực lĩnh (3)

Thu nhập thực tế (1) = Lương cơ bản (LCB) + Lương thưởng đánh giá tháng (LT)

Các khoản khấu trừ (2) = BHXX + BHYT + BHTN + ĐP + TNCN

Thực lĩnh (3) = Thu nhập thực tế (1) - Các khoản khấu trừ (2)

BHXX (Bảo hiểm xã hội) = 8%\*LCB

BHYT (Bảo hiểm y tế) = 1,5%\*LCB

BHTN (Bảo hiểm thất nghiệp) = 1%\*LCB

$$\text{ĐP (Đoàn phí)} = 1\% * \text{LCB}$$

$$\text{Thuế TNCN (Thuế Thu nhập cá nhân)} = \text{Thu nhập chịu thuế (TNTT)} * \text{Thuế suất}$$

Thu nhập chịu thuế được tính bằng:

$$\text{TNTT} = \text{Thu nhập thực tế (1)} - \text{Các khoản giảm trừ} - \text{Tiền đóng bảo hiểm}$$

$$\text{Tiền đóng bảo hiểm (BH)} = \text{BHXH} + \text{BHYT} + \text{BHTN}$$

$$\text{Các khoản giảm trừ} = \text{Giảm trừ bản thân (11.000.000 VNĐ)} + \text{Người phụ thuộc (4.400.000 VNĐ / 1 người)}$$

Dựa vào điều kiện của Thu nhập chịu thuế (TNTT) như bảng dưới đây ta sẽ có cách tính Thuế TNCN:

Bậc	Thu nhập tính thuế /tháng	Thuế suất	Tính số thuế phải nộp	
			Cách 1	Cách 2
1	Đến 05 trđ	5%	0 trđ + 5% TNTT	5% TNTT
2	Trên 05 trđ đến 10 trđ	10%	0,25 trđ + 10% TNTT trên 5 trđ	10% TNTT - 0,25 trđ
3	Trên 10 trđ đến 18 trđ	15%	0,75 trđ + 15% TNTT trên 10 trđ	15% TNTT - 0,75 trđ
4	Trên 18 trđ đến 32 trđ	20%	1,95 trđ + 20% TNTT trên 18 trđ	20% TNTT - 1,65 trđ
5	Trên 32 trđ đến 52 trđ	25%	4,75 trđ + 25% TNTT trên 32 trđ	25% TNTT - 3,25 trđ
6	Trên 52 trđ đến 80 trđ	30%	9,75 trđ + 30% TNTT trên 52 trđ	30 % TNTT - 5,85 trđ
7	Trên 80 trđ	35%	18,15 trđ + 35% TNTT trên 80 trđ	35% TNTT - 9,85 trđ

$$\text{Ví dụ: TNTT} = 6.000.000 \text{ VNĐ thì thuế TNCN} = 250.000 + 10\% * (\text{TNTT} - 5.000.000) = 350.000 \text{ VNĐ}$$

Viết chương trình in ra số tiền Bảo hiểm (BH) và Thuế TNCN của một người trong một năm (12 tháng) biết Thu nhập thực tế (1) và LCB của người đó hàng tháng và người này không có người phụ thuộc.

**Input:**

Dòng đầu tiên số nguyên N ( $1 \leq N \leq 10$ ) là số tập test.

N dòng tiếp theo, mỗi dòng gồm 2 số m, n lớn  $> 0$ . Trong đó m là Thu nhập thực tế (1) và n là LCB

**Output:**

In ra 2 số là tiền tiền đóng bảo hiểm (BH) và thuế TNCN (lấy đến phần nguyên) trong một năm của người Lao động trên trong 1 năm (12 tháng)

*Trường hợp số nguyên N vượt quá 10 hoặc nhỏ hơn 0 in ra **INVALID INPUT***

*Bộ test nào có giá trị nhỏ hơn 1000 in ra dòng đó là **INVALID INPUT***

**Ví dụ**

**Input:**

4

15700000 6300000

13300000 5700000

100000000 7700000

1000 123

**Output:**

7938000 2423100

7182000 1020900

9702000 252204300

INVALID INPUT

### CÂU 9: KHOẢNG CÁCH HAVERSINE

Khoảng cách Haversine là một đại lượng để xác định khoảng cách địa lý giữa hai địa điểm bất kỳ. Giả sử hai địa điểm  $P_1$  và  $P_2$  được quy đổi về kinh độ và vĩ độ tương ứng là  $long_1, lat_1, long_2, lat_2$ . Khoảng cách Haversine  $d$  được xác định theo công thức sau:

$$a = \sin^2\left(\frac{\Delta lat}{2}\right) + \cos(lat_1) \cdot \cos(lat_2) \cdot \sin^2\left(\frac{\Delta long}{2}\right)$$

$$c = 2 \cdot \arcsin(\sqrt{a})$$

$$d = R \cdot c$$

Trong đó:

$$\Delta lat = lat_2 - lat_1$$

$$\Delta long = long_2 - long_1$$

$R = 6731$  (km) (Bán kính Trái Đất)

Hãy viết hàm tính khoảng cách Haversine giữa hai địa điểm bất kỳ.

Input:

- Kinh độ và vĩ độ tương ứng của hai địa điểm, mỗi địa điểm được viết trên một dòng, mỗi giá trị cách nhau bởi khoảng trắng và là các số thực (float).

Output:

- Khoảng cách Haversine giữa hai địa điểm, kết quả được in ra với hai số sau dấu phẩy.

Ví dụ:

Input	Output
105.96 10.21	4686.47
107.17 16.79	

### CÂU 10:

Khai báo lớp Tam giác với ba thuộc tính là 3 cạnh của một tam giác. Viết các phương thức phù hợp để kiểm tra xem 3 cạnh đó có tạo thành một tam giác vuông hay không. Nếu có, tính chu vi và diện tích của tam giác đó.

#### Input:

- 3 cạnh của một tam giác, được viết trên một dòng, cách nhau bởi một khoảng trắng.

#### Output:

- Nếu 3 cạnh của tam giác không tạo thành một tam giác vuông thì in ra **INVALID**
- Nếu 3 cạnh tạo thành một tam giác vuông, in ra chu vi và diện tích của tam giác.

#### Ví dụ:

Input	Output
3 4 5	12 6
5 9 12	INVALID