

## Cơ số - BASE

Trong hệ 10, biểu thức “ $6 \times 9 = 42$ ” là không đúng, nhưng biểu thức này sẽ đúng trong hệ cơ số 13.

Với 3 số nguyên  $p$ ,  $q$  và  $r$ , hãy xác định số nguyên  $B$  ( $2 \leq B \leq 16$ ) để biểu thức  $p \times q = r$  là đúng trong cơ số  $B$ . Nếu có nhiều số  $B$  thỏa yêu cầu thì đưa ra giá trị  $B$  nhỏ nhất. Nếu không tồn tại  $B$ , thì đưa ra số 0.

**Dữ liệu:** gồm 3 số nguyên  $p$ ,  $q$  và  $r$ , các số cách nhau ít nhất một dấu cách ( $1 \leq p, q, r \leq 10^6$ ).

**Kết quả:** cơ số  $B$  tìm được.

**Dữ liệu mẫu:**

Dữ liệu	Kết quả	Giải thích
6 9 42	13	$(6, 9, 42)_{13} \sim (6, 9, 54)_{10}$

## Mã hóa - ENCODING

Mật mã để mở cửa phòng thực hành máy tính là một dãy gồm các chữ số từ 1 đến 9 – gọi là mã số, sinh viên muốn vào thực hành cần phải nhập đúng mã số này. Để đảm bảo an ninh, mã số sẽ được mã hóa thành thông điệp chỉ chứa các ký tự a, b và c theo quy tắc sau:

Chữ số	1	2	3	4	5	6	7	8	9
Mã	a	b	cc	bbc	cbc	abc	bac	aac	cac

Ví dụ mã số 132 sẽ được chuyển thành thông điệp accb.

Trước buổi thực hành, giảng viên sẽ cung cấp thông điệp và sinh viên phải nhanh chóng tìm ra mã số tương ứng để có thể vào phòng thực hành.

**Dữ liệu:** thông điệp là một xâu chứa không quá 100 ký tự (a,b, c).

**Kết quả:** Đưa ra mã số tương ứng với thông điệp từ dữ liệu.

Dữ liệu được đảm bảo để luôn có thể tìm được mật mã.

**Dữ liệu mẫu:**

Dữ liệu	Kết quả
accb	132

## Máy in thẻ sinh viên – CARD\_PRINTER

Phòng F1 của UIT mới vừa được trang bị máy in thẻ sinh viên. Máy hoạt động theo nguyên tắc sau:

- Máy in từng thẻ SV một, máy in sẽ in từng lớp màu riêng biệt đè lên nhau.
- Một lượt quét, máy sẽ in 1 màu duy nhất từ một vị trí nào bất kỳ và kéo dài đến vị trí kết thúc của dải màu cần in. Sau đó đầu in sẽ trở về phần đầu của thẻ.

Máy in có các màu sau: g, r, b, w. Độ dài tối đa của thẻ là 200 điểm ảnh.

**Dữ liệu:** chuỗi các màu cần in lên thẻ (g, r, b, w).

**Kết quả:** Đưa ra số lượt quét tối thiểu.

**Dữ liệu mẫu:**

Dữ liệu	Kết quả	Giải thích
wwbbbb	2	máy sẽ in ww, sau đó sẽ in bbbb
wwbbwwbb	3	máy sẽ in wwwwwwww, in bb ở vị trí 2, in tiếp bb ở vị trí 6



## Trò chơi Oẳn tù xì – RPS (rock - paper - scissors)

Oẳn tù xì là một trò chơi thiếu nhi khá phổ biến trên toàn thế giới thường dùng để giải quyết tranh cãi. Có nhiều tài liệu cho rằng trò chơi này bắt nguồn từ châu Á và nhanh chóng lan rộng ra toàn thế giới với ưu điểm luật chơi đơn giản và không cần công cụ hỗ trợ. Người chơi sẽ chọn ngẫu nhiên 1 trong 3 vật: búa, kéo và bao. Sau đó cả hai người chơi công bố lựa chọn của mình cùng một lúc bằng cách dùng bàn tay ra dấu. Kết quả được quyết định theo nguyên tắc: Búa thắng kéo, kéo thắng bao, và bao thắng búa. Nếu cả 02 người chơi có cùng lựa chọn, kết quả được tính là hòa.

Phòng F1 muốn tổ chức game show truyền hình “vua tù xì UIT” trong đó có yêu cầu người chơi thực hiện oẳn tù xì và bộ phận công nghệ thông tin sẽ hỗ trợ xác định thắng thua. Phòng F1 đã được trang bị một hệ thống thị giác máy tính để nhận diện lựa chọn từ bàn tay của người chơi. Nhiệm vụ bây giờ của bạn, một lập trình viên của ban tổ chức, là nhận dữ liệu từ hệ thống và chỉ ra người thắng cuộc.

**Dữ liệu:** gồm 2 dòng, mỗi dòng ghi 1 trong 3 từ BUA, KEO, BAO. Dòng thứ nhất là lựa chọn của người chơi A, dòng thứ 2 là lựa chọn của người chơi B

**Kết quả:** hiện thông báo người thắng cuộc hoặc kết quả hòa theo 1 trong 3 tình huống: “A WON”, “B WON” hoặc “DRAW”

**Dữ liệu mẫu:**

Dữ liệu	Kết quả
BUA BAO	B WON
KEO KEO	DRAW
BAO BUA	A WON

## Đếm số ký tự trùng nhau – COMMON LETTERS

Cho 02 xâu ký tự, hãy đếm số lượng ký tự xuất hiện ở cả 02 chuỗi.

**Dữ liệu:** gồm 2 dòng, mỗi dòng ghi 1 xâu ký tự không chứa khoảng trắng. Độ dài mỗi xâu không quá 100000 ký tự

**Kết quả:** số ký tự xuất hiện ở cả 2 chuỗi

**Dữ liệu mẫu:**

Dữ liệu	Kết quả	Giải thích
Aabbcc Adcbeb	4	Chung: 1 chữ A, 2 chữ b, và 1 chữ c
aaa aa	2	Chung: 2 chữ a

## Đồ thị có hướng – DIRECT GRAPH

Cài đặt một số thao tác cơ bản trên một đồ thị có hướng, mỗi đỉnh của đồ thị được đặt tên là một xâu ký tự trong bảng chữ cái.

Các thao tác bao gồm: thêm đỉnh, thêm cạnh, kiểm tra cạnh và đếm số cạnh đi ra từ một đỉnh  
Bắt đầu với đồ thị rỗng.

**Dữ liệu:** gồm nhiều dòng, mỗi dòng mô tả 1 thao tác phải thực hiện được mô tả theo 1 trong các dạng sau:

- Bắt đầu với số 1, sau đó là xâu ký tự V. Dòng này mô tả thao tác thêm đỉnh có tên là V vào đồ thị
- Bắt đầu với số 2, sau đó là 2 xâu ký tự U, V cách nhau bởi khoảng trắng. Dòng này mô tả thao tác thêm cạnh nối từ U đến V vào đồ thị
- Bắt đầu với số 3, sau đó là 2 xâu ký tự U, V cách nhau bởi khoảng trắng. Dòng này mô tả thao tác kiểm tra có cạnh nối từ đỉnh U tới đỉnh V hay không?
- Bắt đầu với số 4, sau đó là xâu ký tự V. Dòng này mô tả thao tác đếm số cạnh đi ra từ đỉnh V
- Dòng chỉ với số 0 thể hiện kết thúc dữ liệu vào.

**Kết quả:** Mỗi dòng dữ liệu cần được đưa ra 1 dòng kết quả theo qui ước sau:

- Với thao tác thêm đỉnh V, nếu V đã có trong đồ thị thì kết quả là xâu “DUP”, nếu chưa có thì thêm đỉnh và kết quả là xâu “ADD”
- Với thao tác thêm cạnh (U, V): nếu U hoặc V hoặc cả 2 chưa có trong đồ thị thì thêm các đỉnh đó và thêm cạnh (U, V) vào đồ thị; kết quả là xâu “ADD2”. Trong trường hợp ngược lại, nếu cạnh (U, V) đã có thì kết quả là xâu “DUP2”, nếu cạnh chưa có thì thêm cạnh và kết quả là xâu “ADD3”
- Với thao tác kiểm tra cạnh (U, V): kết quả là xâu “TRUE” nếu tồn tại cạnh nối từ đỉnh U tới đỉnh V và là “FALSE” trong trường hợp ngược lại.
- Với thao tác đếm số cạnh đi ra từ đỉnh V: kết quả là số lượng cạnh đi ra từ đỉnh V. Nếu không có cạnh nào đi ra từ đỉnh V xuất ra số 0

**Dữ liệu mẫu:**

Dữ liệu	Kết quả
2 Ma Quy	ADD2
1 Quai	ADD
2 Quai Quy	ADD3
2 Yeu Quai	ADD2

1 Ma	DUP
2 Quai Quy	DUP2
3 Quy Ma	FALSE
3 Ma Quy	TRUE
4 Ma	1
4 Quy	0
2 Quai Ma	ADD3
4 Quai	2
0	
3 J Z	FALSE
1 P	ADD
4 T	0
4 I	0
4 Y	0
3 S L	FALSE
1 N	ADD
4 O	0
3 S O	FALSE
1 R	ADD
3 T V	FALSE
3 K M	FALSE
2 A R	ADD2
2 B G	ADD2
4 B	1
4 Q	0
2 M O	ADD2
4 E	0
4 J	0
1 U	ADD
0	