# 2017   Hey, Youre Not Marion Jones!

The Olympic committee has hired American Code Masters (ACM) to verify the IDs issued to all the athletes, reserves, judges, staff, and the press. Each badge has a barcode written on it in base-5, encoding the ID number. The system of ID numbers uses a check-digit scheme to detect errors and reduce forgeries. You are to write a program to help ACM detect invalid ID numbers.

The devices that security uses to read the barcodes produce strings of the letters V, W, X, Y, Z. Each letter represents a base-5 digit: V represents 4, W represents 3, X represents 2, Y represents 1, and Z represents 0. So, WXZ = 320 (base-5), which is 85 (base-10). The base-5 number is first converted to a base-10 number. Any number with more than 8 (base-10) digits is considered invalid. Numbers with less than 8 digits are padded on the left with zeroes. IDs are allocated based on the most significant digit (in base-10):

| | |
|---|---|
| 0, 1 | athletes; |
| 2, 3 | reserves, |
| 4, 5 | judges; |
| 6, 7 | staff; and |
| 8, 9 | press. |

Consider the ID number $d_7\, d_6 \ldots d_1\, d_0$ expressed in base-10, where $d_i$ ($0 \le i \le 7$) is a single digit of the ID number. For this ID to be valid the following checksum value must be 0:

$$F(0, d_0) \times F(1, d_1) \times F(2, d_2) \times \ldots \times F(6, d_6) \times F(7, d_7)$$

We will define the function $F(i, j)$ and the operator '$\times$' next. The function $F$ is defined as:

$$F(i, j) = \begin{cases} j & \text{if } i = 0 \\ G(j) & \text{if } i = 1 \\ F(i - 1, G(j)) & \text{otherwise} \end{cases}$$

The definition of the function $F$ depends on a permutation of the decimal digits we call $G$:

| $j$: | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 |
|---|---|---|---|---|---|---|---|---|---|---|
| $G(j)$: | 1 | 5 | 7 | 6 | 2 | 8 | 3 | 0 | 9 | 4 |

That is, $G(0) = 1$, $G(1) = 5$, etc.

The function $i \times j$ is based on dihedral groups and has the nice property that transposing digits in the ID creates a checksum error. It is defined as follows:

$$i \times j = \begin{cases} (i + j) \bmod 5 & \text{if } i < 5, j < 5 \\ (i + (j - 5)) \bmod 5 + 5 & \text{if } i < 5, j \ge 5 \\ ((i - 5) - j) \bmod 5 + 5 & \text{if } i \ge 5, j < 5 \\ (i - j) \bmod 5 & \text{otherwise} \end{cases}$$

Note that $-4 \bmod 5 = 1$.

The operator '$\times$' is left-associative, so for example $i \times j \times k = (i \times j) \times k$.

## Input

The first line of the input contains an integer $n \ge 1$. Each of the next $n$ lines contains a single scanned barcode representing a potential ID number. Each scanned barcode will consist only of the characters 'V', 'W', 'X', 'Y', and 'Z' and will be at least 1 and at most 12 letters long.

## Output

You should output the ID number as read, its corresponding base-10 number, and a message telling either that the ID is invalid or, if it is valid, the type of participant that the ID belongs to. Use the wording and format in the sample output below.

## Sample Input

```
6
WYYXWVZXX
YWYWYYXWVZYY
YWYWYYXWVZYX
YYZWYYXWVZYX
YXXWYYXWVZXW
XYXWYYXWXYY
```

## Sample Output

```
WYYXWVZXX/01274262 is valid athlete id number
YWYWYYXWVZYY/81352381 is valid press id number
YWYWYYXWVZYX/81352382 is invalid id number
YYZWYYXWVZYX/59868007 is valid judge id number
YXXWYYXWVZXW/73539888 is valid staff id number
XYXWYYXWXYY/22520431 is valid reserve id number
```