Validating Credit Card Numbers

You and Fredrick are good friends. Yesterday, Fredrick received N credit cards from **ABCD Bank**. He wants to verify whether his credit card numbers are valid or not. You happen to be great at regex so he is asking for your help!

A valid credit card from **ABCD Bank** has the following characteristics:

► It must start with a 4, 5 or 6.
► It must contain exactly 16 digits.
► It must only consist of digits (0-9).
► It may have digits in groups of 4, separated by *one* hyphen **"-"**.
► It must `NOT` use any other separator like ' ' , '_', etc.
► It must `NOT` have 4 or more consecutive repeated digits.

Examples:

Valid Credit Card Numbers

```
4253625879615786
4424424424442444
5122-2368-7954-3214
```

Invalid Credit Card Numbers

```
42536258796157867      #17 digits in card number → Invalid
4424444424442444       #Consecutive digits are repeating 4 or more times → Invalid
5122-2368-7954 - 3214  #Separators other than '-' are used → Invalid
44244x4424442444       #Contains non digit characters → Invalid
0525362587961578       #Doesn't start with 4, 5 or 6 → Invalid
```

Input Format

The first line of input contains an integer N.
The next N lines contain credit card numbers.

Constraints

$0 < N < 100$

Output Format

Print 'Valid' if the credit card number is valid. Otherwise, print 'Invalid'. Do not print the quotes.

Sample Input

```
6
4123456789123456
5123-4567-8912-3456
61234-567-8912-3456
4123356789123456
5133-3367-8912-3456
5123 - 3567 - 8912 - 3456
```

Sample Output

```
Valid
Valid
Invalid
Valid
Invalid
Invalid
```

Explanation

4123456789123456 : **Valid**
5123-4567-8912-3456 : **Valid**
61234-567-8912-3456 : **Invalid**, because the card number is not divided into equal groups of 4.
4123356789123456 : **Valid**
51$33$-$33$67-8912-3456 : **Invalid**, consecutive digits 3333 is repeating 4 times.
5123 — 4567 — 8912 — 3456 : **Invalid**, because space ' ' and `-` are used as separators.