

In this challenge, you will determine whether a string is *funny* or not. To determine whether a string is funny, create a copy of the string in reverse e.g. *abc* \rightarrow *cba*. Iterating through each string, compare the absolute difference in the `ascii` values of the characters at positions 0 and 1, 1 and 2 and so on to the end. If the list of absolute differences is the same for both strings, they are funny.

Determine whether a give string is funny. If it is, return `Funny`, otherwise return `Not Funny`.

Example

`s = 'lmnop'`

The ordinal values of the charcters are `[108, 109, 110, 111, 112]`. `sreverse = 'ponml'` and the ordinals are `[112, 111, 110, 109, 108]`. The absolute differences of the adjacent elements for both strings are `[1, 1, 1, 1]`, so the answer is `Funny`.

Function Description

Complete the `funnyString` function in the editor below.

`funnyString` has the following parameter(s):

- `string s`: a string to test

Returns

- `string`: either `Funny` or `Not Funny`

Input Format

The first line contains an integer `q`, the number of queries.
The next `q` lines each contain a string, `s`.

Constraints

- $1 \leq q \leq 10$
- $2 \leq \text{length of } s \leq 10000$

Sample Input

STDIN	Function
-----	-----
2	q = 2
acxz	s = 'acxz'
bcxz	s = 'bcxz'

Sample Output

Funny
Not Funny

Explanation

Let r be the reverse of s .

Test Case 0:

$s = \mathbf{acxz}, r = \mathbf{zxca}$

Corresponding ASCII values of characters of the strings:

$s = [97, 99, 120, 122]$ and $r = [122, 120, 99, 97]$

For both the strings the adjacent difference list is $[2, 21, 2]$.

Test Case 1:

$s = \mathbf{bcxz}, r = \mathbf{zxcb}$

Corresponding ASCII values of characters of the strings:

$s = [98, 99, 120, 122]$ and $r = [122, 120, 99, 98]$

The difference list for string s is $[1, 21, 2]$ and for string r is $[2, 21, 1]$.