

# Test Data Engineer

## Exercises:

1. Implement the function `thousands_with_commas` that gets an integer and returns a string with the number with commas every 3 digits.

```
def thousands_with_commas(i):  
    #Escribí tu código aquí.  
    return str(i)  
  
if __name__ == '__main__':  
    assert thousands_with_commas(1234) == '1,234'  
    assert thousands_with_commas(123456789) == '123,456,789'  
    assert thousands_with_commas(12) == '12'
```

2. Implement the "annograms" function that uses the `WORD.LST` file to return anagrams of the word given in the "word" parameter.

```
def annograms(word):  
    #Escribí tu código aquí.  
    words = [w.rstrip() for w in open('WORD.LST')]  
    raise NotImplementedError  
  
if __name__ == "__main__":  
    print(annograms("train"))  
    print('--')  
    print(annograms('drive'))  
    print('--')  
    print(annograms('python'))
```

`WORD.LST`: in this [link](#).

3. Bleatrix Trotter the sheep has devised a strategy that helps her fall asleep faster. First, she picks a number  $N$ . Then she starts naming  $N$ ,  $2 \times N$ ,  $3 \times N$ , and so on. Whenever she names a number, she thinks about all of the digits in that number.

She keeps track of which digits (0, 1, 2, 3, 4, 5, 6, 7, 8, and 9) she has seen at least once so far as part of any number she has named. Once she has seen each of the ten digits at least once, she will fall asleep.

Bleatrix must start with  $N$  and must always name  $(i + 1) \times N$  directly after  $i \times N$ .

For example, suppose that Bleatrix picks  $N = 1692$ . She would count as follows:

- $N = 1692$ . Now she has seen the digits 1, 2, 6, and 9.
- $2N = 3384$ . Now she has seen the digits 1, 2, 3, 4, 6, 8, and 9.
- $3N = 5076$ . Now she has seen all ten digits, and falls asleep.

What is the last number that she will name before falling asleep? If she will count forever, print INSOMNIA instead.

## INPUT

The first line of the input gives the number of test cases,  $T$ .  $T$  test cases follow. Each consists of one line with a single integer  $N$ , the number Bleatrix has chosen.

## OUTPUT

For each test case, output one line containing Case # $x$ :  $y$ , where  $x$  is the test case number (starting from 1) and  $y$  is the last number that Bleatrix will name before falling asleep, according to the rules described in the statement.

### LIMITS

$$1 \leq T \leq 100.$$

### DATASET

$$0 \leq N \leq 200.$$

### SAMPLE

Input

5

0

1

2

11

1692

Output

Case #1: INSOMNIA

Case #2: 10

Case #3: 90

Case #4: 110

Case #5: 5076

- In Case #1, since  $2 \times 0 = 0$ ,  $3 \times 0 = 0$ , and so on, Bleatrix will never see any digit other than 0, and so she will count forever and never fall asleep. Poor sheep!
- In Case #2, Bleatrix will name 1, 2, 3, 4, 5, 6, 7, 8, 9, 10. The 0 will be the last digit needed, and so she will fall asleep after 10.
- In Case #3, Bleatrix will name 2, 4, 6... and so on. She will not see the digit 9 in any number until 90, at which point she will fall asleep. By that point, she will have already seen the digits 0, 1, 2, 3, 4, 5, 6, 7, and 8, which will have appeared for the first time in the numbers 10, 20, 30, 40, 50, 60, 70, 80, and 90, respectively.
- In Case #4, Bleatrix will name 11, 22, 33, 44, 55, 66, 77, 88, 99, 110 and then fall asleep.
- Case #5 is the one described in the problem statement.

Sample input: in this [link](#).

4. Given the following dataset: [applicationdata](#).  
Generate a report with a tool of your choice (eg. pandas, jupyter notebooks, etc), the report must comply the following requirements :
  - a. The summary\_date must be in ISO Date format.
  - b. The metrics must be grouped by campaign.
  - c. The metrics that must be:
    - i. Number of impressions.
    - ii. Number of clicks.
    - iii. Number of installs.
    - iv. Amount of spend.
    - v. CTR (number of clicks/number of impressions)
    - vi. CPI (amount of spend/number of installs)

*Plus: if you wish, include an exploratory analysis of the dataset.*

## Requirements:

1. The resolution of the exercises should be uploaded to GitHub and send the repository link to [developers@winclap.com](mailto:developers@winclap.com).