

# Google Code Jam '19 Round 1A Problem C - Alien Rhyme

---

**Time limit:** 20.0s    **Memory limit:** 1G

---

During some extraterrestrial exploration, you found evidence of alien poetry! Your team of linguists has determined that each word in the alien language has an accent on exactly one position (letter) in the word; the part of the word starting from the accented letter is called the accent-suffix. Two words are said to rhyme if both of their accent-suffixes are equal. For example, the words `PROL` and `TARPOL` rhyme if the accented letter in both is the `O` or the `L`, but they do not rhyme if the accented letters are the `R`s, or the `R` in `PROL` and the `P` in `TARPOL`, or the `O` in `PROL` and the `L` in `TARPOL`.

You have recovered a list of  $N$  words that may be part of an alien poem. Unfortunately, you do not know which is the accented letter for each word. You believe that you can discard zero or more of these words, assign accented letters to the remaining words, and then arrange those words into pairs such that each word rhymes only with the other word in its pair, and with none of the words in other pairs.

You want to know the largest number of words that can be arranged into pairs in this way.

## Input Specification

---

The first line of the input gives the number of test cases,  $T$ .  $T$  test cases follow. Each test case starts with a line with a single integer  $N$ . Then,  $N$  lines follow, each of which contains a string  $W_i$  of uppercase English letters, representing a distinct word. Notice that the same word can have different accentuations in different test cases.

## Output Specification

---

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 size of the largest subset of words meeting the criteria described above.

## Limits

---

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

Time limit: 20 seconds per test set.

Memory limit: 1 GB.

$$1 \leq \text{length of } W_i \leq 50, \text{ for all } i.$$

$W_i$  consists of uppercase English letters, for all  $i$ .

$W_i \neq W_j$ , for all  $i \neq j$  (Words are not repeated within a test case).

## Test Set 1

$$2 \leq N \leq 6.$$

## Test Set 2

$2 \leq N \leq 1000$ .

## Sample Input

```
4
2
TARPOL
PROL
3
TARPOR
PROL
TARPRO
6
CODEJAM
JAM
HAM
NALAM
HUM
NOLOM
4
PI
HI
WI
FI
```

## Sample Output

```
Case #1: 2
Case #2: 0
Case #3: 6
Case #4: 2
```

In Sample Case #1, the two words can rhyme with an appropriate accent assignment, as described above, so the largest subset is the entire input.

In Sample Case #2, no two words can rhyme regardless of how we assign accents, because any two suffixes will differ at least in the last letter. Therefore, the largest subset is the empty one, of size 0.

In Sample Case #3, we can use the entire set of words if we accentuate `CODEJAM` and `JAM` at the `J` s, `HAM` and `NALAM` at their last `A` s and `HUM` and `NOLOM` at the `M` s.

In Sample Case #4, any two words can be made to rhyme, but always by making the accented letter the **I**. Therefore, if we add two pairs to the subset, words from different pairs will rhyme. We can, thus, only form a subset of size 2, by choosing any 2 of the input words.