Touchscreen Keyboard

Nowadays, people do not use hardware keyboards but touchscreens. Usually, they touch on the wrong letters with their chunky fingers, because screen space is precious and the letters therefore too small. Usually, a spell checker runs after typing a word and suggests other words to select the correct spelling from. Your job is to order that list so that more likely words are on top. The typical touchscreen keyboard looks like this:

```
qwertyuiop
asdfghjkl
zxcvbnm
```

You should use the distance between the letters to type a word: the distance is the sum of the horizontal and vertical distance between the typed and proposed letter. Assume you typed a `w`, the distance to `e` is 1, while the distance to `z` is 3. The typed word and the list of words from the spell checker all have the same length. The distance between two words is the sum of the letter distances. So the distance between `ifpv` and `icpc` is 3.

Input

The first line of the input contains an integer t. t test cases to follow, each of them separated by a blank line.

Each test case begins with a line consisting of a string s and an integer ℓ, where s is the word that was typed using the touchscreen keyboard, and ℓ specifies the number of entries in the spell checker. ℓ lines follow, each containing one word of the spell checker list.

Output

For each test case, print a line containing "Case #i:" where i is its number, starting at 1. Print ℓ more lines. The j-th line should consist of s_j d_j, where s_j is the j-th word in sorted order by increasing distance in the spell checker list, and d_j is its distance. If two words have the same distance, sort them alphabetically.

Constraints

- $1 \leq t \leq 20$
- $1 \leq \ell \leq 10$
- $1 \leq |s|, |s_j| \leq 10^5$ for all $1 \leq j \leq \ell$
- All words of one test case have the same length.
- All words consist of only the lowercase letters, i.e., `a-z`.
- Each word appears at most once in the spell checker list per test case.

Sample Input 1

```
2
ifpv 3
iopc
icpc
gcpc

edc 5
wsx
edc
rfv
plm
qed
```

Sample Output 1

```
Case #1:
icpc 3
gcpc 7
iopc 7
Case #2:
edc 0
rfv 3
wsx 3
qed 4
plm 17
```