

Д-р М стартира собствен курс по програмиране, където показва различни бързи алгоритми. Той много държи да знае имената на своите курсисти и е решен да запомни всичките. За да оптимизира времето да им каже целите имена той прилага следната практика: всеки курсист има ново, съкратено име в курса, с което М ще се обръща към него. Имайки нареден списък с курсисти, името на даден курсист е най-краткия префикс, който не е префикс на друг курсист, вече срещнат в списъка. Вашата задача е да генерирате списъка с новите имена на курсистите.

Ако цялото име на курсист вече е срещнато, то новото му име е *старо\_име число* (разделени с интервал), показващо кой подред е със същото име.

## Input Format

Първият ред на входа е размерът на списъка с курсисти N.

Следват N реда с първоначалните имена на курсистите.

## Constraints

$$1 \leq N \leq 10^5$$

Всяко име е до 10 символа.

Всяко име е съставено от символи *a-z*.

## Output Format

N реда, където на всеки ред е съкратеното име на съответния курсист.

## Sample Input 0

```
3
alexander
alina
georgi
```

## Sample Output 0

```
a
ali
g
```

## Explanation 0

- 1. Тривиален случай - най-краткия префикс е 'a'.
- 2. И двата префикса - 'a', 'al' са префикси на вече срещнат курсист, следващият най-кратък е 'ali'.

## Sample Input 1

```
4
antoan
anton
antonina
anton
```

### Sample Output 1

```
a
anton
antoni
anton 2
```

### Explanation 1

1. Тривиален случай
2. Всичките префикси на 'anton' - 'a', 'an', 'ant', 'anto' са префикси на предишен курсист - съкратеното име е 'anton'
3. Аналогично на 2
4. Цялото име 'anton' вече сме го срещнали, съкратеното име е 'anton 2'

### Sample Input 2

```
3
ivan
i
i
```

### Sample Output 2

```
i
i
i 2
```

### Explanation 2

По горните изисквания, не е изключено двама студенти да имат едно и също съкратено име. Например при този вход, двама се казват i. Цялото име на втория курсист е префикс на предишен, но е първият с това име, затова няма число, което да следва.