

Qualquer Tolo Consegue

Contest Local, Universidade de Ulm  Alemanha**Timelimit: 5**

Com certeza você conhece alguém que acha que é muito esperto. Você, então, decide o deixar triste com o seguinte problema:

- "Você sabe me dizer qual é a sintaxe de um conjunto?", você o pergunta.
- "Claro!", ele responde, "um conjunto abriga a possibilidade de uma lista vazia de elementos dentro de duas chaves. Cada elemento pode estar dentro de um outro conjunto ou ser uma letra do alfabeto escolhido. Elementos da lista devem ser separados por uma vírgula".
- "Então, se eu lhe der uma palavra, você consegue me dizer se ela é a representação sintaticamente correta de um conjunto?"
- "Claro, qualquer tolo consegue!" é a resposta dele.

Agora você o pegou! Você o apresenta a seguinte gramática, definindo formalmente a sintaxe para um conjunto (que foi descrita informalmente por ele):

```
Conjunto ::= "{" ElementoDaLista "}"
ElementoDaLista ::= <empty> | Lista
Lista ::= Elemento | Elemento "," Lista
Elemento ::= Átomo | Conjunto
Átomo ::= "{" | "}" | ","
```

<vazio> significa uma palavra vazia, i.e., a lista em um conjunto pode ser vazia. Logo ele perceberá que esta tarefa é muito mais difícil do que ele pensou, já que o alfabeto consiste de caracteres que também são usados para a sintaxe do conjunto. Então, ele diz que não é possível decidir, de forma eficiente, se uma palavra consistindo de "{", "}" e "," é a representação sintaticamente correta de um conjunto ou não. Para prová-lo errado, você precisa escrever um programa eficiente que decidirá este problema.

Entrada

A primeira linha da entrada contém um número representando o número de linhas que seguem.

Cada linha consiste de uma palavra, para qual o seu programa deve decidir se é a representação sintaticamente correta de um conjunto. Você pode assumir que cada palavra contém entre 1 e 200 caracteres de um conjunto { "{", "}", "," }.

Saída

A saída para cada caso de teste deve dizerse uma palavra ("word") consiste em um conjunto ("set") ou não ("no set"). Você deve aderir ao formato mostrado na saída de exemplo.

Exemplo de Entrada	Exemplo de Saída
4	Word #1: Set
{ }	Word #2: Set
{ { } }	Word #3: Set
{ { } }, { , }	Word #4: No Set
{ , , }	

