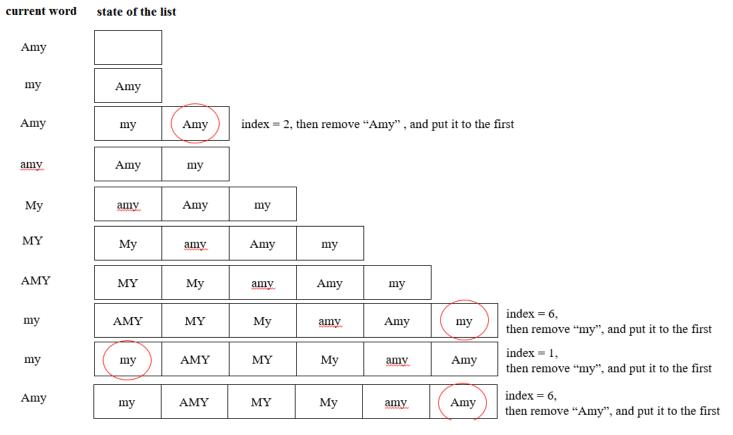# [Reference] Paragraph Compression

Here is a method to **compress** a paragraph. Given **a paragraph only consists of alphabets** (used to represent a word), and **other symbols** (used to separate two words), e.g. comma, dot, space, etc. **There will be no number in a paragraph**. Rules to generate such a compressed paragraph and maintain a list to generate a compressed paragraph:

(1) Once it reads a word that never occurs, **put that word in the front of the list** (It could be either a string array or a linked list. Depend on your choice), and output the original word.

(2) If the incoming word has already been in the list, then **replace and output that word with its index number** (index starts from 1) in the list, **remove it from the list, and put it to the front of the list again**.

(3) Always output the symbolic in the original position.

(4) Words "Abc" and "abc" represent different words. "X-ray" represents words "X" and "ray", which are separate by symbol "-".

(5) The maximum line of the paragraph is **10**. The maximum of length of a word is **15**. The number of characters in a line won't exceed **80**.

The following picture shows the steps to compress a paragraph "**Amy my Amy amy My MY AMY my my'Amy**". Notice that the list is always empty before reading a paragraph.

After compression, the paragraph becomes "**Amy my 2 amy My MY AMY 6 1'6**".

Examples of compression

```
============ Input Paragraph ============
Amy my Amy amy My MY AMY my my'Amy
============ Output Paragraph ============
Amy my 2 amy My MY AMY 6 1'6
```

```
============ Input Paragraph ============
I love you.
You love him,
and he loves him!
 He loves her. ><
However...she loves you and I.
============ Output Paragraph ============
I love you.
You 3 him,
and he loves 4!
 He 3 her. ><
However...she 4 11 9 12.
```