

## Sequences

We would like you to write an algorithm that counts the number of occurrences of a word in a dictionary. Although it might seem to have no real-world applications, this algorithm is in fact used in bioinformatics for reporting deletions in a DNA sequence. The input of the algorithm is a JSON file with integer keys and list of strings values, for example:

```
{
  "1": ["windows", "server"],
  "2": ["crystalzoom"],
  "3": ["python", "crystalzoom", "linux"],
  "4": ["crystalzoom"],
  "7": ["java", "crystalzoom", "cpp", "js"],
  "9": ["crystalzoom"],
  "10": ["ruby", "rails"]
}
```

We need a function to count the number of occurrences of the string "crystalzoom" on the list but only when the keys where they appear are consecutive numbers. The first occurrence needs to be replaced by "crystalzoom" plus the number of occurrences found (i.e. "crystalzoom3", "crystalzoom5", etc.) and all further matches need to be removed from the lists of all consecutive keys. If all strings are removed from a key, this key has to be deleted from the dictionary.

Let's look at some examples to help you understand the problem:

```
"9": ["crystalzoom"],
"10": ["ruby", "rails"]
```

In the example above, "crystalzoom" appears only once since "10" is a consecutive number but does not contain the string "crystalzoom". Therefore it has to be replaced with "crystalzoom1":

```
"9": ["crystalzoom1"],
"10": ["ruby", "rails"]
```

And, in this case below:

```
"2": ["crystalzoom"],
"3": ["python", "crystalzoom", "linux"],
```

Here "crystalzoom" is present in two keys and, since "2" and "3" are consecutive numbers, the first occurrence is replaced with "crystalzoom2" and the second one is removed:

```
"2": ["crystalzoom2"],
"3": ["python", "linux"],
```

And, if we have something like this:

```
"2": ["crystalzoom"],
"3": ["python", "crystalzoom", "linux"],
"4": ["crystalzoom"],
```

In this case the total number of "crystalzoom" matches is 3 as "2", "3" and "4" are consecutive. However, after removing "crystalzoom" from key "4" it becomes empty so it needs to be deleted:

```
"2": ["crystalzoom3"],
"3": ["python", "linux"],
```

And in this final case:

```
"7": ["java", "crystalzoom", "cpp", "js"],
"9": ["crystalzoom"],
```

Here "7" and "9" are not consecutive, therefore it counts only as 1 occurrence:

```
"7": ["java", "crystalzoom1", "cpp", "js"],  
"9": ["crystalzoom1"],
```

Going back to the original example, the output of your script should be:

```
{  
  "1": ["windows", "server"],  
  "2": ["crystalzoom3"],  
  "3": ["python", "linux"],  
  "7": ["java", "crystalzoom1", "cpp", "js"],  
  "9": ["crystalzoom1"],  
  "10": ["ruby", "rails"]  
}
```

Deliverables:

- A single code file.
- There is no need to load the JSON file from disk. You can also pre-load it as a Javascript value.
- Your script needs to handle different input in the above-described format.