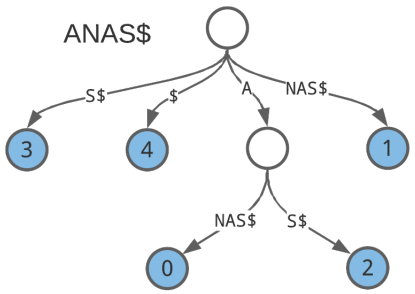
9C Construct the Suffix Tree of a String

Suffix Tree Construction Problem
Construct the suffix tree of a string.

Input: A string *Text*.
Output: SUFFIXTREE(*Text*).



Formatting

Input: A string *Text* with an appended dollar-sign ("$").
Output: A space-separated list of edge labels from the constructed suffix tree (in any order).

Constraints

- The length of *Text* will be between 1 and 10^3.

Test Cases

Case 1

Description: The sample dataset is not actually run on your code.

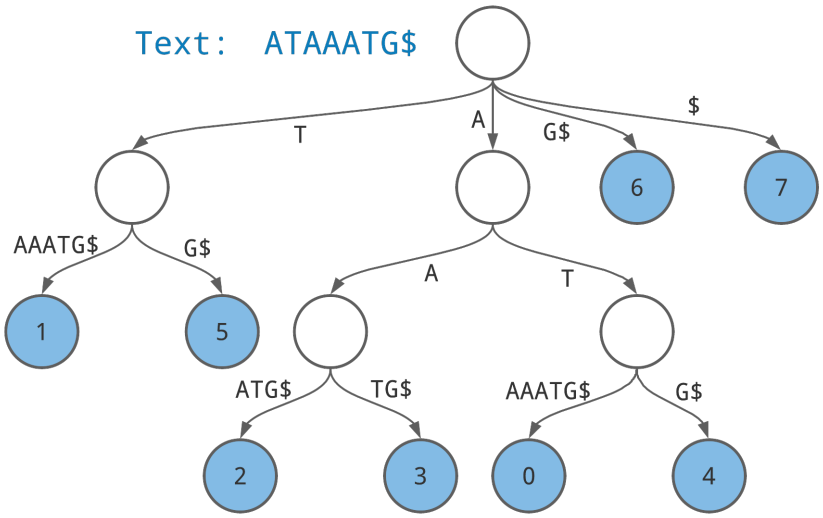
Input:

```
ATAAATG$
```

Output:

```
$ $ A A AAATG$ AAATG$ ATG G$ G$ G$ T T TG$
```

Figure:



Above is the suffix tree for the string `ATAAATG$` (notice the $ appended to the end of our input string `ATAAATG`). Each path from the root to each of the leaves (shown in blue) represents the suffix of `ATAAATG$` corresponding to the index in the leaf.

Case 2

Description: There are repeats in *Text*.

Input:
```
AATCAATC$
```

Output:
```
$ $ $ $ A AATC$ AATC$ AATC$ ATC C TC TC
```

Case 3

Description: There are no repeats in *Text*.

Input:
```
ATCG$
```

Output:
```
$ ATCG$ CG$ G$ TCG$
```

Case 4

Description: Large regions of *Text* being a single character or short tandem repeat (STR).

Input:
```
AAACA$
```

Output:
```
$ $ A AACA$ ACA$ C$ CA$
```

Case 5

Description: A larger dataset of the same size as that provided by the randomized autograder. Check input/output folders for this dataset.