9G Construct the Suffix Array of a String

Suffix Array Construction Problem

Construct the suffix array of a string.

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

```
7  $
1  ANANAS$
3  ANAS$
5  AS$
0  BANANAS$
2  NANAS$
4  NAS$
6  S$
```

Formatting

Input: A string *Text*.
Output: A space-separated list of integers corresponding to SUFFIXARRAY(*Text*).

Constraints

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

Test Cases

Case 1

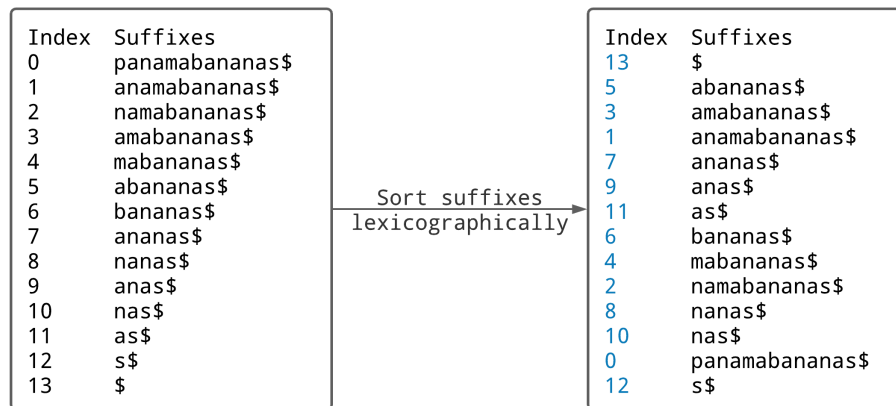Description: The sample dataset is not actually run on your code.

Input:
```
panamabananas
```

Output:
```
13 5 3 1 7 9 11 6 4 2 8 10 0 12
```

Figure:

```
Index  Suffixes                    Index  Suffixes
0      panamabananas$              13     $
1      anamabananas$               5      abananas$
2      namabananas$                3      amabananas$
3      amabananas$                 1      anamabananas$
4      mabananas$                  7      ananas$
5      abananas$      Sort suffixes 9     anas$
6      bananas$      lexicographically 11 as$
7      ananas$                     6      bananas$
8      nanas$                      4      mabananas$
9      anas$                       2      namabananas$
10     nas$                        8      nanas$
11     as$                         10     nas$
12     s$                          0      panamabananas$
13     $                           12     s$
```

Shown above is a general (and inefficient) construction of the suffix array of the input string `panamabananas`. We first generate all suffixes of *Text* before sorting the suffixes lexicographically and outputting the indices representing the sorted suffixes as the complete suffix array of *Text*.

Case 2

Description: There are repeats in *Text*.

Input:
AATCAATC

Output:
8 4 0 5 1 6 2 7 3

Case 3

Description: There are no repeats in *Text*.

Input:
ATCG

Output:
4 0 2 3 1

Case 4

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

Input:
AAACA

Output:
5 4 0 1 2 3

Case 5

Description: Many different characters in one pattern.

Input:
ABCFED

Output:
6 0 1 2 5 4 3

Case 6

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