Algorithms in the Real World (15-853), Fall 06
Assignment #1 **Due: September 18th**

You are not permitted to look at solutions of previous year assignments, or to search the web for related literature.

Problem 1: Sorting the Ends of the Line (50pt)

Given a string $S = s_1 s_2 s_3 \dots s_n$, consider all the suffixes $S_1, S_2, \dots, S_i, \dots, S_n$ of this string, where $S_i = s_i s_{i+1} \dots s_n$ is the i^{th} suffix of S. Define $S_i < S_j$ if the string S_i lexicographically precedes S_j. Can you sort the n suffixes in $O(n)$ time? If you cannot give a linear time algorithm, give one with the least running time you can construct.

We would prefer you not use suffix trees for this problem. If you do use suffix trees, you can get at most 40 points, and you must properly describe how to do it in linear time and space even when the number of different characters is not considered constant.

The input is an array `S` of length n containing the string S. The output is an array `A` of length n, with each position `A[i]` containing an integer between 1 and n, such that

$$\texttt{A[i]} < \texttt{A[j]} \Leftrightarrow S_i < S_j.$$

For example, given $S =$ `algs in the real$`, the suffixes are

```
algs_in_the$
lgs_in_the$
gs_in_the$
s_in_the$
_in_the$
in_the$
n_the$
_the$
the$
he$
e$
$
```

Since the sorted list would be

```
algs_in_the$
e$
gs_in_the$
he$
in_the$
lgs_in_the$
n_the$
s_in_the$
the$
_in_the$
_the$
$
```

the output will be given by `A = [1 11 3 10 6 2 7 4 9 5 8 12]`.