Suffix Arrays and Longest Common Prefixes

...

Motivation

Given a string, how many repeated substrings does it have?

https://open.kattis.com/problems/substrings

Solution?

- Brute force.
- Add every substring to a set.
- Output size of set.

Complexity?

- Suppose input string length is N
- There are O(N^2) substrings
- Compare 2 strings take O(N)
- O(N^3)
- Cubic in the size of the input

Suffix Arrays:
What is it?

What is a suffix?

Consider the string “asdf”

What are the suffixes?

“f”, “df”, “sdf”, “asdf”

Is this a suffix? “as”

No

What is lexicographical sorting?

“aaa”, “aba”, “aa”, how do you sort these lexicographically?

“aa”, “aaa”, “aba”

What is a suffix array?

A suffix array of a string S is a lexicographically sorted array of all suffixes of S.

Example: input string “abacabacx”

i	sa	suffix
0	0	abacabacx
1	4	abacx
2	2	acabacx
3	6	acx
4	1	bacabacx
5	5	bacx
6	3	cabacx
7	7	cx
8	8	x

LCP:
What is it?

What is a LCP?

The longest common prefix array stores the length of the longest common prefixes between two adjacent elements in a suffix array.

Example: input string “abacabacx”

i	lcp	sa	suffix
0	0	0	abacabacx
1	4	4	abacx
2	1	2	acabacx
3	2	6	acx
4	0	1	bacabacx
5	3	5	bacx
6	0	3	cabacx
7	1	7	cx
8	0	8	x

Motivation
How to solve Repeated Substring with LCP?

Example: input string “aabaab”

i	lcp	sa	suffix
0	?	3	aab
1	?	0	aabaab
2	?	4	ab
3	?	1	abaab
4	?	5	b
5	?	2	baab

Example: input string “aabaab”

i	lcp	sa	suffix
0	0	3	aab
1	?	0	aabaab
2	?	4	ab
3	?	1	abaab
4	?	5	b
5	?	2	baab

Example: input string “aabaab”

i	lcp	sa	suffix
0	0	3	aab
1	3	0	aabaab
2	?	4	ab
3	?	1	abaab
4	?	5	b
5	?	2	baab

Example: input string “aabaab”

i	lcp	sa	suffix
0	0	3	aab
1	3	0	aabaab
2	1	4	ab
3	?	1	abaab
4	?	5	b
5	?	2	baab

Example: input string “aabaab”

i	lcp	sa	suffix
0	0	3	aab
1	3	0	aabaab
2	1	4	ab
3	2	1	abaab
4	?	5	b
5	?	2	baab

Example: input string “aabaab”

i	lcp	sa	suffix
0	0	3	aab
1	3	0	aabaab
2	1	4	ab
3	2	1	abaab
4	0	5	b
5	?	2	baab

Example: input string “aabaab”

i	lcp	sa	suffix
0	0	3	aab
1	3	0	aabaab
2	1	4	ab
3	2	1	abaab
4	0	5	b
5	1	2	baab

Suffix Arrays:
How to implement?

Naive Implementation

- Generate every suffix
- Sort them

Naive Implementation Complexity?

- Sorting takes O(Nlog(N)) compares
- Each string compare takes O(N) time
- Overall O(N^2 log(N))

Better Implementation

SA Implementation Problem: Burrows-Wheeler

Burrows-Wheeler

https://open.kattis.com/problems/burrowswheeler

Solution?

- Generate every shifted string and sort them.
- O(N^2 log(N))
- Bad

Suffix Array Solution?

- Compute suffix array SA
- For i from 0 to N-1, print char at SA[i]