OH TODAY:  1:15pm - 2:15pm

Emails

Violation of course policies

Suffix Tries

Text $T$ of length $m$.

$T \rightarrow T\$ $\rightarrow$ abababc$\$

$S = \{\$, a\$, ba\$, aba\$, baba\$, abcba\$, baba\$, abababa\$, abababa\$\$

Diagram:

[Diagram showingSuffix Trie structure with nodes and edges labeled with characters and transitions.]
Every path from root to a leaf corresponds to a suffix of T and for every suffix of T, there is a path from the root to a leaf in the trie.

If no $ then the above property does not hold.

With $, no suffix is a prefix of another suffix.
Q. Is $s$ a substring of $T$?

Starting from the root, follow the edges corresponding to each character in $s$. If we "fall off" the trie tree, the answer is no.

```
else
    ans = yes
```

Note: Every substring is a prefix of some suffix.

Q. Is $s$ a suffix of $T$?

Do the same thing as in the previous question, checking the node you end up at has an edge with label '$s$' leaving it.
3) How many times appears as substring in $T$?

Solution: Follow edges corr. to char $a \in \Sigma$.

Let $u$ be the node we end up at.

Count # leaves in the tree rooted at $u$. This is our answer.

4) Longest repeated substring of $T$? That is, the longest substring of $T$ that appears $\geq$ twice.

Solution: Deepest node in the tree that has $\geq 2$ children.

# nodes in a Suffix trie?
\( |T| = m \)

**total # characters in all suffixes of \( T \)?**

\( O(m^2) \).

**Space:** \( O(m^2) \).

Consider texts that take \( O(m) \) space.

\( T: \ a\ a\ a\ \ldots\ a\ \\ \underbrace{m}_{\text{\# of } a} \)

\( O(m) \).

Consider \( T: a^m b^n \)

\( a^m \overbrace{b^n}^{\text{\# of } b} \).
\[ \Theta(n^2), \quad m \approx 2^n \]

\[ \Rightarrow \Theta(m^2) \]

- Combine all edges that come out of nodes that have only one outgoing edge.

\[ T: \quad a, b a b, a b c \]

\[ \text{Ans: } 0, 2, 4 \]
# nodes in the tree: $O(m)$

In our suffix tree each node has $\geq$
2 children. Thus \( \# \text{ int. nodes} \leq \# \text{ leaves} \).

We know \( \# \text{ leaves} = n \).

\( \Rightarrow \) Total nodes: \( \Theta(n) \).

Build a Suffix tree:

1. Create suffixes.
2. Build a Std. Trie.
3. Compress edges as necessary.

Time: \( \Theta(n^2) \) \( \checkmark \)  Space: \( \Theta(n^2) \)

Time: \( \Theta(m \log m) \), Space: \( \Theta(m^2) \)

D D D D D

\( 2(m^2) \) time.

Algo 2: \( \overline{abbaab} \)
Time: $O(m)$  
Space: $O(m)$
\( O(m^2) \)

\[ T : a^n b^n \quad \text{e.g.} \quad a^n b^n \]

Ukkonen: \( O(m) \) time, \( O(m) \) space.

Online Gusfield.

1. We have a pattern \( P \). We want to find the indices where \( P \) occurs in \( T \).

**Solu:**
1. Build a suffix tree using \( T \).
    Start from root.
2. If follow the edges corresponding to chains in \( P \).
3) Let u be the node when we end up.

4) Return the indices corresponding to each leaf in the subtree rooted at u.

Running time: $O(m + n)$

Subtree has to be “full”

- Every node has $\geq 2$ children
- Total # nodes = $O(k)$

2) Longest Common Substring of X and Y.

$X: \text{tap}$, $Y: \text{app}$

$O(m+n)$
T: $\{tap, \#, app\}$

- Build a suffix tree using T.

- Go to each node and annotate each node as $X$ or $Y$ or $XY$.

- Go to the deepest node that is marked $XY$. 