6.897
L10.2
Mar. 31, 2003

Searching in a suffix tree

- $O(|P|)$ time to find an occurrence of P — actually find all occurrences of P as subtree
- Augment tree with subtree size $\Rightarrow$ count #occurrences of P.
  leftmost descendent $\Rightarrow$ enumerate k in $O(k)$.
- Longest repeated substring: deepest nonleaf — $O(n)$

Building suffix tree in $O(n)$ time ($|\Sigma| = O(1)$) [Weiner]

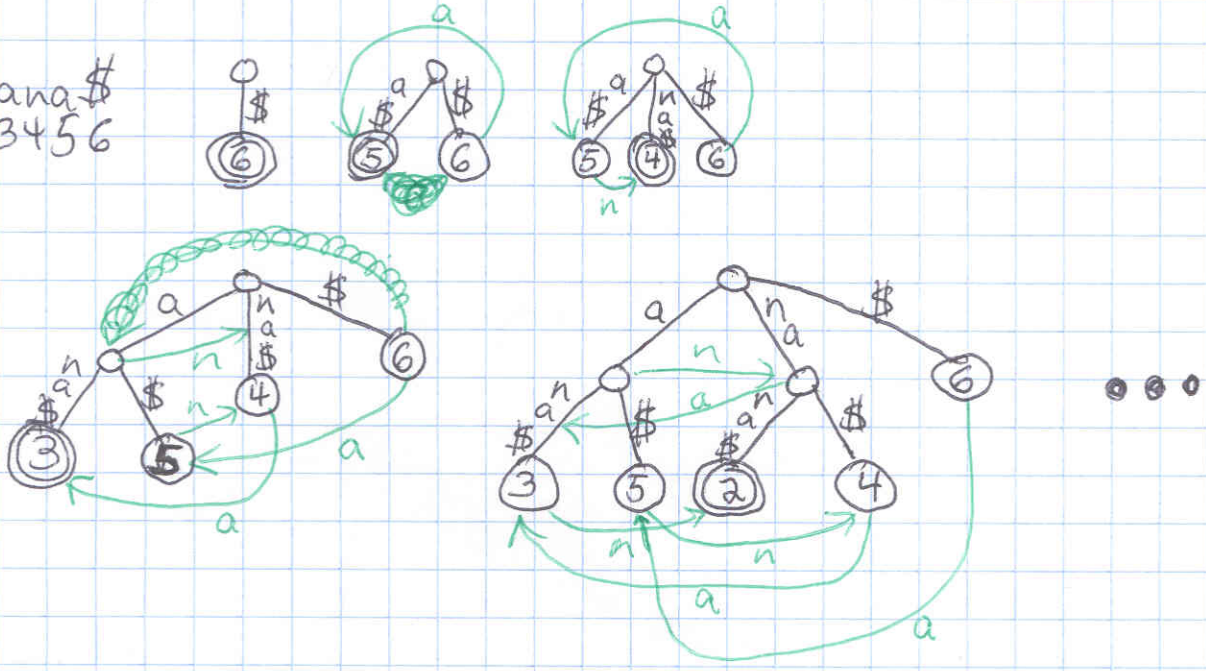
not covered in L10

- At high level: Insert $T[i:]$ for $i = |T|-1, \ldots, 1, 0$.
- $T[i:]$ branches off at longest prefix of $T[i:]$ that is a substring of $T[i+1:]$
- Idea: store at each node x a prepend table mapping $a \in \Sigma$ to pointer to "ax" (if exists)
  maybe node or middle of edge
- To find where $T[i:]$ branches off, try to prepend $T[i]$ to $T[i+1:]$, $T[i+1:|T|-\alpha]$, ... until success
  just added ← parent ... grandparent

[
- If we end up with a node, add $T[i:]$ as a child leaf
- If in middle of an edge, split the edge, add $T[i:]$ as child of new node.
]

[
- Construct prepend table of $T[i:]$ by trying to extend parent's & new node from split
- Update prepend tables of $T[i+1:]$ & all but last visited ancestor.
]

banana$
0123456



→ Farach (FOCS 1997): $O(n)$ time even for $\Sigma = \{1, \ldots, k\}$ (!)