DSC 40B - Discussion 01

Problem 1.

Draw a trie representing the following collection of strings:

{"friend", "frog", "frob", "fun", "glob", "glarb", "glow", "guard"}

Problem 2.

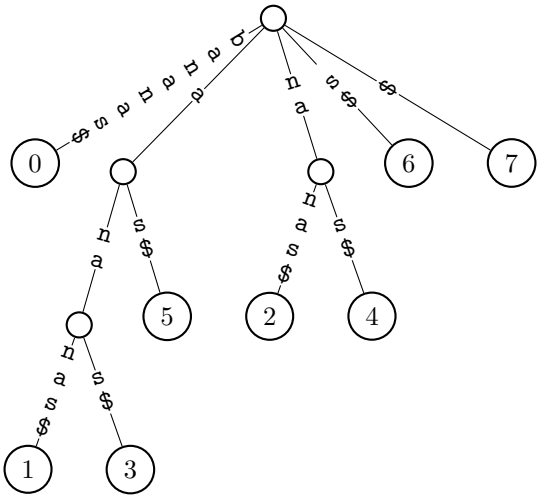
Draw the suffix tree for the following strings:

a) "mississippi"

b) "aaaa"

Problem 3.

Each edge in a suffix tree for s is associated with one or more characters. For instance, the edges in the suffix tree for "bananas" represent the strings "bananas$", "nas$", "na", "a", etc.



a) Suppose the suffix tree is implemented by storing the string associated with each edge explicitly in memory. In other words, the string "nas$" will be stored somewhere in memory, "na" will be stored, etc. How much memory is used in the worst case to stored these strings in terms of $|s|$?

Hint: in the worst case, the string s has no repeated characters.

> **Solution:** If s has no repeated characters, then the root node has $|s|+1$ branches: one for each suffix of s. Each edge goes directly to a leaf node, and the edge stores the suffix string. The largest suffix is of length $|s|$, the second is of length $|s|-1$, the next has length $|s|-2$, and so on down to the empty string. The total number of characters stored is therefore $\Theta(|s|^2)$.

b) Describe a way of storing the suffix tree so that only $\Theta(|s|)$ memory is used for storing strings. In other words, how can we know what string is associated with each edge without actually storing that string?

Solution: Each string is a contiguous substring of s, so instead of storing each substring we can store the string s once, and for each edge store the start index and stop index. For instance, for the string `s = "bananas"`, if `"ana"` appears on an edge we should simply store `3` and `6`, since `s[3:6] == "ana"`.

This will require $\Theta(1)$ memory per edge. Since there are $\Theta(|s|)$ nodes, there are $\Theta(|s|)$ edges in the tree, for a total of $\Theta(|s|)$ memory.