1 Suffix Trees

Most Common Letter: Given a text $T$, determine the most commonly occurring letter in $T$.

Challenge Questions

Find Anagrams: Given a list of $n$ strings of length $L$ over some alphabet $\mathcal{E}$, determine how many strings are anagrams of some other string in the list.

Is Valid Suffix Tree?: Given a suffix tree $T$, determine if it is a valid suffix tree for some text.

2 Building a Suffix Array

Build the suffix array for the text $T = \text{"neverever"}$.

3 Huffman Encoding

1. Build the Huffman tree for $S = \text{"pusheen"}$

2. Given $n$ characters with frequencies $1, 1, 2, 4, 8, ..., 2^{n-2}$, show that the Huffman tree of these frequencies would be of height $n - 1$.

3. Below is an encoding trie $T$ for the string “xerxes”. Argue that this is not a Huffman tree.
4. More generally, let $T$ be an encoding trie with two characters $a$ and $b$ such that $f(a) < f(b)$ and $d_T(a) < d_T(b)$ (where $d_T(c)$ denotes the depth of the leaf that stores $c$). Argue that $T$ is not a Huffman tree.

5. Let $T$ be an encoding trie with two characters $a$ and $b$ such that $f(a) \leq f(b)$ and $d_T(a) \leq d_T(b)$. Let $T'$ be the trie obtained by switching $a$ and $b$. Argue that the encoding with $T'$ is no longer than the one of $T$.

Remark: With this insight, doing Case 2 of Huffman-optimality is very easy. Time permitting, we may do this as well.