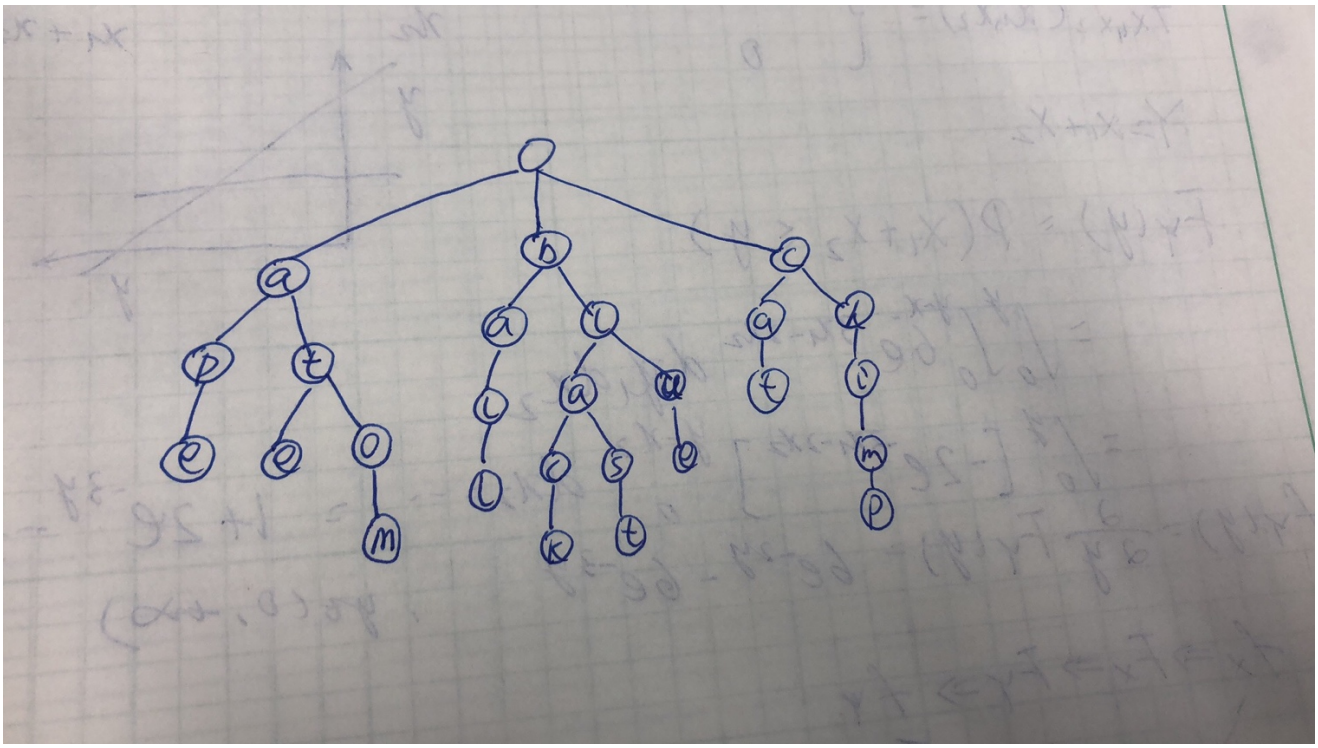
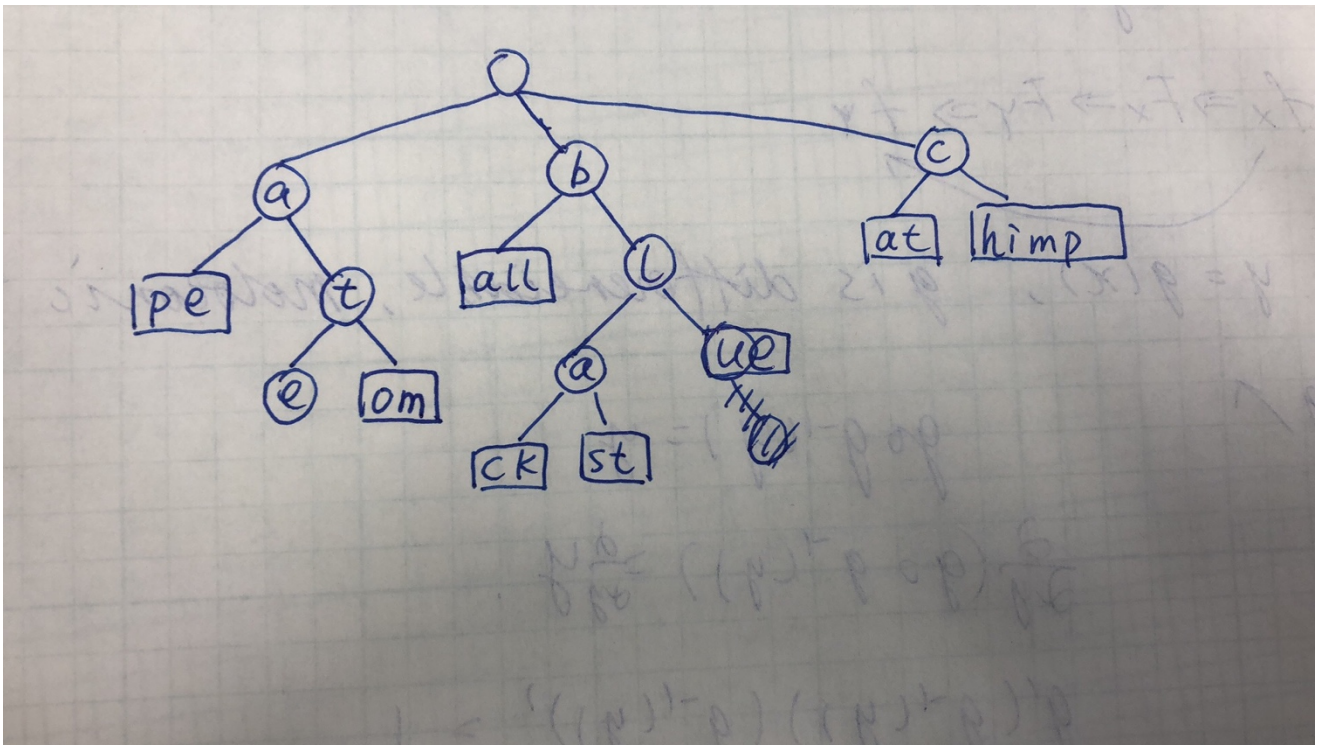
CS251 Homework5

Liyao (Mars) Gao

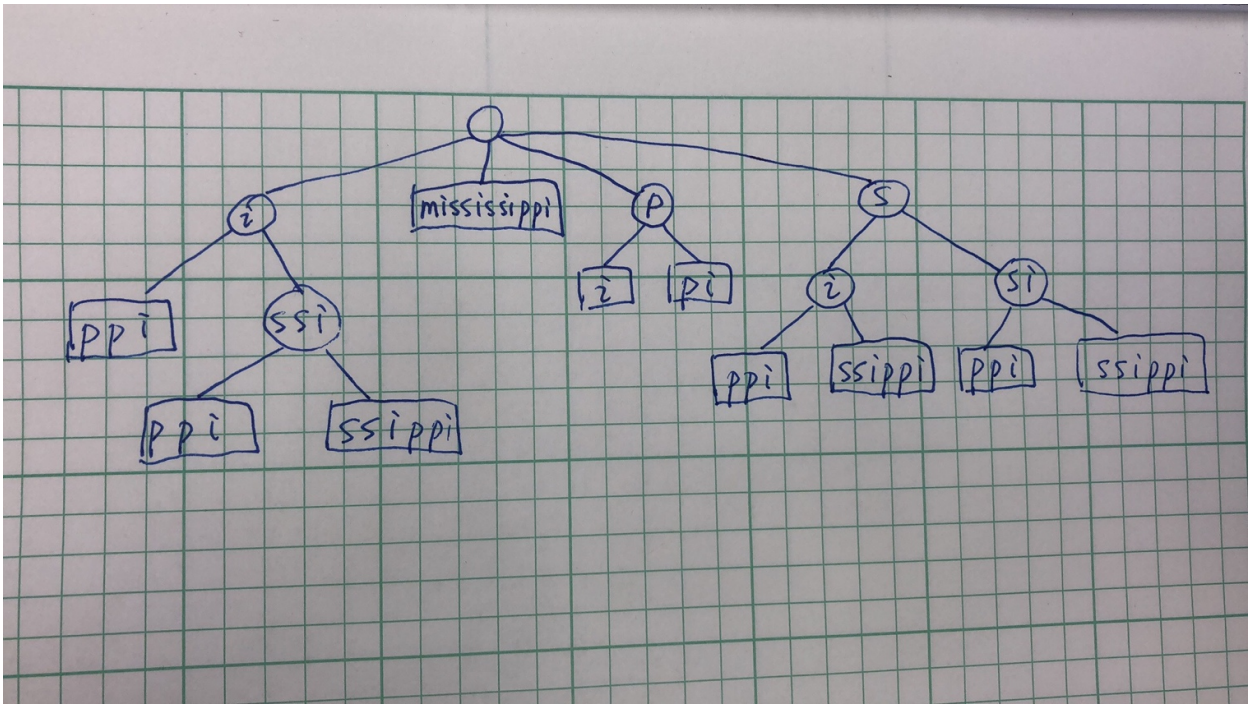
1.



2.



3.



4.

Revised Preorder Traversal Searching.

Explanation: Basically, the algorithm is accessing tree nodes like preorder traversal. It outputs the first appearance by strings matching. Then accessing the child nodes. This guarantees the shortest matching will output for the first.

Time Complexity: $O(NK)$.

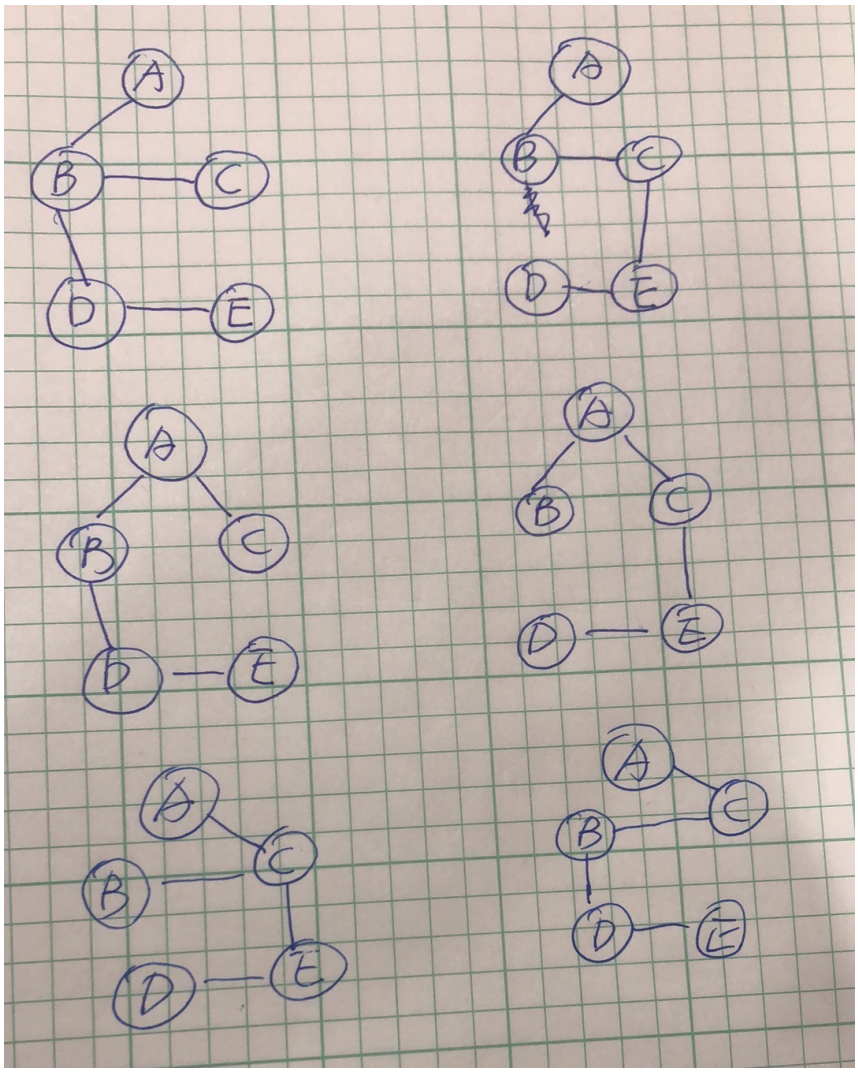
Space: $O(NL)$.

5.

b)

Explanation: Time complexity for constructing a standard trie is O(NL) in this case. The searching time complexity for all kinds of tries is O(KM) in this case.
Therefore, the time complexity will be O(NL+KM), using the parameters: ‘N’, ‘L’, ‘K’, ‘M’.

6.



7.
(a)
a) d)
Explanation: Using Prim's algorithm, starting from A, we have (A,D), (A,B). Then, since (A,C) and (D,F) have same weight. So A and D could both be possible outcome of this algorithm.

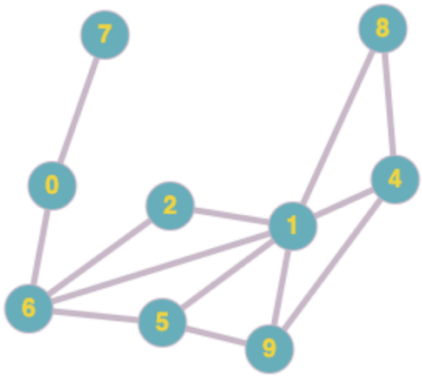
(b)
(E,G), (C,F), (F,G), (A,D), (A,B)

Explanation: Using Kruskal's Algorithm, we aim to find the smallest weight edge for each time.
The lengths of edges in this case is ordered by 2,4,5,7,10.

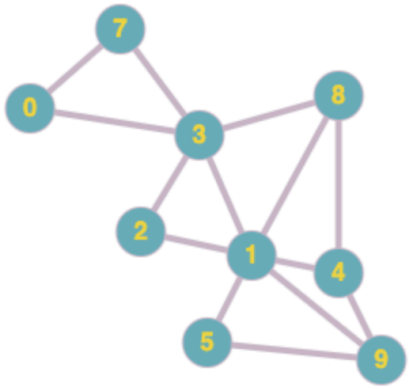
8.

c) e)

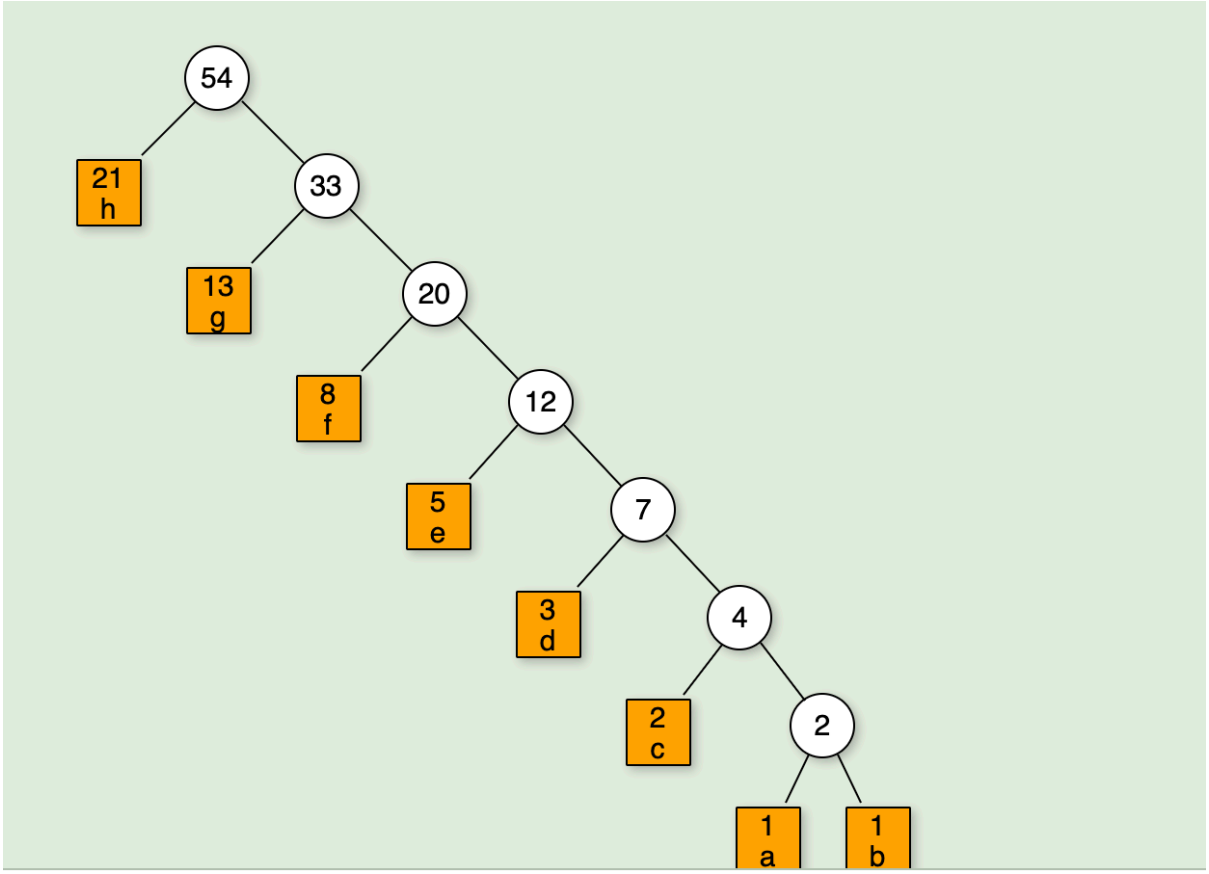
Explanation: After we remove vertex 3. If we remove 0 additionally, this will no longer be biconnected.



After we remove 6, if we remove 3, the graph will no longer be connected.



9.



h: 0, g: 10, f: 110, e: 1110, d: 11110, c: 111110, a: 1111110, b: 1111111.

decaf: 11110 1110 111110 1111110 110
d e c a f

10.

$n-1$.

Explanation: Consider the first two nodes, 1, and 1. 1+1=a_3. The smallest two elements are the formed node, and the third node.

Consider the n-th case. The formed tree node will always be the smallest two elements. Since $\sum a_i = a_{n+1} - 1$. We have the smallest two nodes are a_n, and the formed node.

Therefore, the depth will always add by one after every time of operation of Huffman tree.

The depth will be $n-1$.