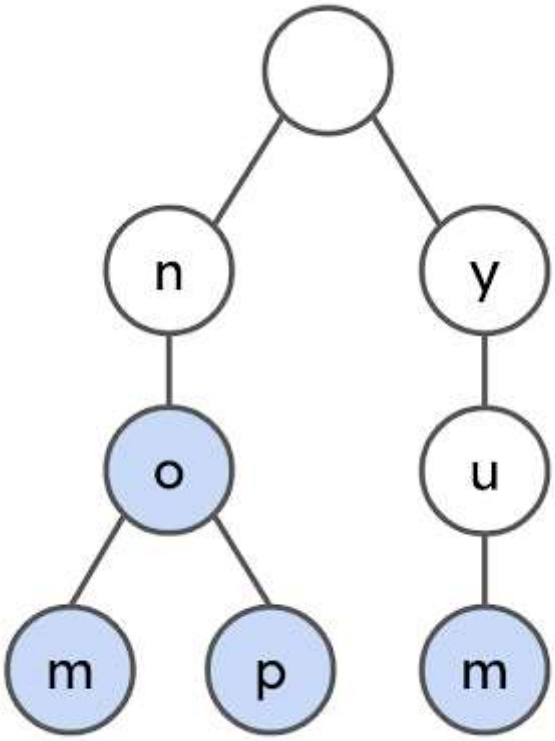
Which of the following Strings are contained within this trie? * 1 point



- [ ] n
- [ ] no
- [ ] nom
- [ ] nop
- [ ] y
- [ ] yu
- [ ] yum

Assume we don't treat the String lengths to be constant, but instead know that all Strings are of length <= K. Which of the following statements are true? * 1 point

- [ ] The height (number of edges along the longest root to leaf path) of the trie is <= K.
- [ ] The runtimes of get and add are O(1).
- [ ] The runtimes of get and add are O(K).
- [ ] A single add operation will create at most K new Nodes.

Which of the following statements accurately describe tradeoffs between tracking the children of a node in a DataIndexedCharMap, hash table, and BST? * 1 point

- [ ] Hash tables and BSTs utilize memory more efficiently than DataIndexedCharMaps.
- [ ] The runtime for finding the child using a DataIndexedCharMap is O(1), using a BST is O(logR), and using a hash table is O(R). R is the size of the alphabet.
- [ ] DataIndexedCharMaps are less wasteful as the alphabet size grows larger.

You are given a list of patterns of a's and b's followed by a wildcard (*) representing any number of a's and b's. Given a query string of only a's and b's, you want to find the longest pattern that matches. Select the steps below that will allow us to efficiently find the answer. * 1 point

For example, the pattern "ab*" matches strings "ab", "aba", "abba" and any other string that starts with "ab" followed by a's and b's. You are given a list of patterns:
a*
b*
aba*
ababba*

Now, given a new query string like "ababbba", you want to find the longest pattern that matches. In this case, the answer would be the pattern "aba*".

- [ ] Add all patterns (excluding the trailing *) to a trie.
- [ ] Add the query string to the trie.
- [ ] Find the longest prefix of the query string contained within the trie and return it.
- [ ] Find all keys in the trie with a prefix of the query string, and return the longest one.

Submit Clear form