

Self-Practice Week 10 - Strings (part 2)

String Applications

The goal of this assignment is to use string-related data structures and algorithms to efficiently tackle a variety of problems.

Exercise 1 – Phone directory lookup

Given a list of employees' names and phone numbers, design and implement a data structure and algorithms to efficiently support an employee's phone number lookup. As you enter an employee's name, a list of suggestions should appear matching the characters entered so far.

Hint: use a trie data structure to store employees' names.

Exercise 2 – Tag cloud

Given a large text document D , design and implement an efficient data structure and algorithm to support the generation of a tag cloud, representing the top k most frequently occurring words in D (e.g., the higher the frequency of a word, the higher its weight in the tag cloud). For an example of a large document, use file "5-mobydick.txt". What are the 10 most frequently used words in that document?

Hint: use a trie and a min heap.

Exercise 3 – Prefix-free codes

In data compression, a set of binary strings is said to be "prefix-free" if no string is a prefix of another. For example, $\{01, 10, 0010, 1111\}$ is prefix free, but $\{01, 10, 0010, 10100\}$ is not because 10 is a prefix of 10100. Design an efficient algorithm to determine if a set of binary strings is prefix-free.

Hint: insert the binary strings into a trie.