

Projeto Final – Algoritmos e Estruturas de Dados

(em grupos de 2 alunos)

Tema: Radix-Trees (Tries)

Objetivo:

Implementar a estrutura de dados Radix-Tree (Tries). Essa estrutura deve ser usada para implementar da lista a seguir (a seu critério):

- 1 aplicação **não tão fácil**, ou
- 2 aplicações **Diretas** e 1 aplicação **Fácil**, ou
- 3 aplicações **fáceis**

Escolha entre:

Aplicações Diretas:

A partir de um dicionário (um conjunto de palavras lido de um arquivo texto),

1. para uma entrada qualquer (lida do teclado), sugerir palavras que possuam aquele prefixo (autocompletar)
2. implementar busca de contatos em um celular (listar nomes contidos no dicionário)
3. Buscar em uma base de dados os proprietários de veículos a partir da placa (caso não seja possível informar todos os dígitos, listar possíveis proprietários)
4. Qualquer aplicação similar – busca de palavras a partir de prefixo

Aplicações Fáceis:

1. Implementar uma função do tipo 'contains' (substring, string), que verifica se a substring é parte da string *não é necessariamente o prefixo – pode ocorrer em qualquer posição da string
2. A partir do IP, determinar qual é o país de origem.
3. Buscar a localização de um determinado CEP

Não tão Fáceis:

1. Indexar um texto. Escrever um programa que lê um arquivo texto de entrada e coloca, em ordem alfabética, as palavras e as linhas que cada palavra aparece (ignorando maiúscula/minúscula e pontuação. Ignore palavras com menos de 2 letras) (https://en.wikipedia.org/wiki/Aho%E2%80%93Corasick_algorithm)
2. Similaridade entre 2 documentos utilizando trigrams*

Entregar

1 arquivo compactado contendo:

1. código, devidamente comentado e padronizado, contendo a implementação da RadixTree (em uma **classe OO** -a escolha da linguagem é livre) (vale 35% da nota do projeto)
2. código contendo as aplicações – arquivo à parte da classe (vale 15% da nota do projeto)
3. um relatório de execução das aplicações (informando como deve ser invocada a aplicação, e mostrando alguns testes de conformidade que foram executados -entradas, saídas) (vale 10% da nota do projeto)
4. um pequeno texto descrevendo o que é RADIX TREE, e as operações relacionadas (vale 20% da nota do projeto)
5. uma explicação de como funcionam os algoritmos Aho-Corasick e Similaridade usando trigams (vale 20% da nota)*mesmo para aqueles não implementarem

Disponibilizar os arquivos-fonte de código, arquivos de dicionário utilizados e instruções de utilização em algum repositório ou pasta compartilhada publicamente para que os demais possam acessar (informar o link).

OBS IMPORTANTE: A falha na entrega do código implica que os demais itens não serão avaliados.