

Estruturas de Dados Avançadas (EDA) – INF1010
Departamento de Informática – PUC-Rio - 2023.2
Tarefa 3 – Tabelas de Dispersão (*Hash Tables*)

Descrição:

Implementar uma tabela de dispersão criando o TDA hashtable.

Implementar algoritmos de **inclusão**, **busca** e **exclusão** em vetor de M posições (onde M é o número primo após 1024, ou 2^{10}), usando hashing com tratamento de colisão para armazenar placas de automóveis no formato “CCCNCNN” onde C é um caractere maiúsculo e N um numeral. Os algoritmos devem prever a entrada de dados via arquivo texto contendo as placas de automóveis geradas aleatoriamente (em cada linha do arquivo texto deverá vir uma placa seguida pelos caracteres <cr> <lf>).

A rotina de tratamento de colisão deve armazenar a nova placa no próprio vetor. Não devem ser usadas outras estruturas de dados. A função hash deve ser definida individualmente e utilizar encadeamento interior (versão 1 ou variação- com zona de colisões) ou endereçamento aberto.

Para diferentes valores de entrada: 128, 256 e 512 placas, calcule o **total de colisões geradas** e os **tempos de inclusão e de busca a todos os elementos** do vetor (a busca deve ser feita após o término da inclusão de todas as placas) (complexidade prática).

Todos os aspectos considerados deverão ser avaliados. A avaliação vai ser feita em um mesmo equipamento para todos os trabalhos. A maior nota caberá ao grupo que entregar o trabalho completo, com menor número de colisões e menor tempo de execução dos algoritmos. Para a avaliação é fornecido um arquivo de placas geradas aleatoriamente para servir de comparação entre os trabalhos (número de colisões e tempo de execução).

Orientações de entrega:

Faça o upload no site do EAD dos seguintes:

- O TDA hashtable (interface .h e arquivos .c)
- Um relatório em formato .pdf contendo:
 - Identificação. Nome e matrícula do(s) aluno(s), sendo no máximo 2 alunos por trabalho;
 - Introdução. Breve descrição sobre o conteúdo do trabalho;
 - Estrutura do programa. Descrição da interface e as funções implementadas;
 - Solução. Descrição da solução passo a passo de cada item do programa, incluindo capturas do código fonte e a saída do programa. A descrição da solução deve incluir ainda:
 - Funções hash e método de tratamento de colisões utilizados
 - O total de colisões geradas e os tempos de inclusão e de busca a todos os elementos do vetor com 128, 256 e 512 placas (complexidade prática).
 - Observações e conclusões. Descrição de aspectos que precisem ser destacados, tais como dificuldades e facilidades encontradas.

Data de Entrega: Até 23:59 do dia 22 de Outubro/2023

Atenção:

- Serão avaliados os trabalhos que atendem a todos os requisitos especificados anteriormente, código organizado e relatório completo.
- Trabalhos entregues com atraso sofrerão perda de 10% de sua nota por cada dia após o prazo de entrega