

Algoritmos e Estrutura de Dados II
Prof. Hélder Pereira Borges
Atividade 04 – Tabelas Hash
OBSERVAR AS REGRAS DESCRITAS NA UNIDADE 01

***** Atividade COLABORATIVA *****

Esta atividade deverá ser desenvolvida em duplas. Cada questão deverá ser desenvolvida como um projeto separado, porém, quando for enviado no classroom, será enviado apenas um arquivo compactado por dupla, assim, após finalizadas as implementações e testes, o projeto da questão 1, *ATV04_Q1_FulanoSauro.zip*, será compactado junto com o projeto da questão 2, *ATV04_Q2_FulanoSauro.zip*, no arquivo *ATV04_FulanoSauro.zip* que será efetivamente enviado.

1. Implemente um tipo de dado abstrato que represente uma Tabela Hash. **NÃO PODEM** ser utilizados recursos nativos da linguagem para simplificar o desenvolvimento, ou seja, todas as funcionalidades devem ser **desenvolvidas pelos alunos**. Funções mínimas e obrigatórias: **InserirDeArquivo** (lê um arquivo texto e insere o conteúdo de cada linha como um elemento da tabela), **Inserir** (insere um único elemento na tabela), **Remover**, **Buscar**, **Imprimir** (deve exibir todos os dados da estrutura, um por linha, nos buckets vazios exibir Null). Não esquecer de tratar as colisões e quando necessário, redimensionar a estrutura.

Para fins de testes, deve haver um método **main()** para executar todas as funcionalidades obrigatórias.

2. Utilizando a estrutura desenvolvida na questão 1, cada equipe será responsável por desenvolver uma implementação para um algoritmo de criptografia, conforme a tabela abaixo. Serão utilizados dois arquivos de entrada, com várias linhas, sendo cada uma individualmente processada. No arquivo **Entrada1.txt**, deve ser gerado o *hash* conforme o algoritmo sob sua responsabilidade, já no arquivo **Entrada2.txt**, cada linha corresponderá a um *hash* que deverá se “traduzido” para o normal.

Serão disponibilizados alguns links sobre os algoritmos de criptografia, obviamente, quanto mais cada equipe pesquisar, melhor será, havendo muito mais subsídio para o desenvolvimento da implementação .

Restrições da Atividade:

1. Não utilizar a biblioteca **security** para implementação dos algoritmos.
2. Nos testes de validação, serão utilizados arquivos de entrada com no mínimo **50 KB** de tamanho (1K de linhas). É aconselhável, depois que validar sua solução, testar com arquivos no tamanho especificado.

Duplas	Algoritmos
Daniel Moura e João Gabriel	MD5
Kaillane Corrêa e Marcio Sá	SHA-1
Paulo Henrique e Joniel Costa	SHA-2-224
Edilson Marques e Ramon Moreira	SHA-3-224
Benjamin Marcos e Gabriel Vinicius	RIPEMD-128
Brendo Nicholas e Francisco Rafael	MD5
Marcos Vinicius Cardoso e Valter Pereira	SHA-1
Marcos Vinicius Santos e Maria Luiza	SHA-2-224
Chebl José e Vitória Ávila	SHA-3-224
José Peterson e Jose Francisco	RIPEMD-128