UNIVERSIDADE FEDERAL DA FRONTEIRA SUL **CURSO DE CIÊNCIA DA COMPUTAÇÃO**

GEX609 - Pesquisa e Ordenação de Dados

Trabalho T2

Este trabalho tem como objetivo a prática de operações de ordenação simples, eficiente e

lineares sobre estruturas de dados complexas.

I Instruções

O trabalho deve conter, para cada questão, um arquivo com o código exigido, comentários

explicando a lógica de implementação e uma demonstração de funcionamento do código (função

main). Os casos de teste serão publicados no SIGAA junto com a tarefa.

II Entrega

A entrega da atividade será on-line através do SIGAA. O estudante deve enviar um único arquivo

ZIP contendo as suas implementações e demais itens solicitados.

Data limite para entrega: 17/11/2024, 23h59m.

III Avaliação

A avaliação será realizada de acordo com a entrega do estudante observados os seguintes critérios:

1. código compilando e funcionando sem alterações na lógica;

2. nível de comentários explicando a lógica utilizada;

3. presença da demonstração do código;

4. construção das listas encadeadas simples;

5. utilização dos laços de repetição da forma correta (for, while, do-while, etc).

Em caso de plágio (incluindo utilização de ferramentas de geração de código), a entrega será

desconsiderada e os estudantes envolvidos receberão nota 0. Procedimentos administrativos

poderão ser iniciados após consulta à coordenação de curso. Os professores do CCR se reservam

ao direito de solicitar que estudantes apresentem o código produzido para o trabalho após a

entrega.



UNIVERSIDADE FEDERAL DA FRONTEIRA SUL CURSO DE CIÊNCIA DA COMPUTAÇÃO

GEX609 - Pesquisa e Ordenação de Dados



IV Atividades

- 1. Implemente uma tabela hash para armazenamento de números inteiros positivos.
 - a. Esta tabela deve utilizar a função k mod M para cálculo dos índices.
 - b. A tabela deve tratar as colisões através de encadeamento separado, com uma lista encadeada (ou seja) cada elemento da tabela hash deve ser o início da lista encadeada.
 - c. O tamanho da tabela hash M deve ser setado para 23.
 - d. Utilize o struct abaixo para armazenar os dados da lista encadeada:

```
struct tlist {
   int valor;
   struct tlist* next;
};
```

- 2. Os itens que serão armazenados na tabela hash deverão ser lidos de um arquivo TXT:
 - a. O arquivo deve ser obrigatoriamente nomeado como casos de teste.txt.
 - Manter o caminho do arquivo fixo no código, não deve ser solicitada nenhuma leitura para os usuários.
 - c. Considere que os elementos no arquivo estão separados por ; e não possuem espaços no início ou no final da linha, e nem entre os itens.
 - i. Exemplo:

- 3. Imprima, ao final, a tabela hash resultante.
 - a. Exemplo

```
0 - 23 -> NULL
1 - 47 -> 24 -> NULL
```

- - -