

# MC202GH - Estrutura de Dados - Turmas G e H

## Laboratório 9 - *Espalhamento (hashing)*

**Docente:** Marcelo da Silva Reis

**Monitor PED:** Matheus Abrantes Cerqueira

**Monitores PAD:** Andreas Cisi Ramos  
Wallace Gustavo Santos Lima

30 de novembro de 2022

**Data de entrega:** 09/12/2022

Entrega no codePost<sup>1</sup>

### Informações gerais

Neste laboratório revisitaremos o Laboratório (Lab) 7, visando atualizar o gerenciamento de lista de espera de restaurantes utilizando uma tabela de espalhamento (hashing) como estrutura auxiliar para guardar os índices das pessoas nos heaps. De fato, este trabalho é uma refatoração do Lab 7, com mesma especificação de funcionamento e mesmos testes no codePost.

#### Observações importantes:

1. Siga à risca a especificação do Lab 7 para as operações, formato de entrada e de saída. Os protótipos também são os mesmos.
2. Os testes no codePost deste Lab 9 serão exatamente os mesmos do Lab 7 (ou seja, o código deste laboratório, do ponto de vista de funcionamento, é para ser idêntico ao do Lab 7, porém com uma maior eficiência computacional).
3. Se você já entregou o Lab 7 utilizando percorrimento linear ou árvore de busca binária para gerenciar os índices das pessoas, pode manter a submissão já feita (que será aceita sem descontos) no Lab 7, em seguida substituindo o gerenciamento dos índices por um hashing e submetendo o código atualizado no codePost do Lab 9.

---

<sup>1</sup><https://codepost.io/signup/join?code=ZW239C3IID>

4. Alternativamente, caso você ainda esteja trabalhando no Lab 7, você pode utilizar hashing já no código do Lab 7; nesse caso, o mesmo mesmíssimo código deve ser submetido tanto no codePost do Lab 7 quanto no do Lab 9.

## Questão 1 (10 pontos) - Controle de índice das pessoas nos heaps por hashing

No Laboratório 7, controlamos a ordem de pessoas em uma lista de espera através de uma fila de prioridades. Como precisávamos saber em  $O(\log n)$  tanto quem era o primeiro quanto o último da fila, utilizamos para esse fim, respectivamente, um max-heap e um min-heap.

Entretanto, como uma pessoa poderia ter sua prioridade na fila alterada, assim como era preciso manter os dois heaps “sincronizados”, implementamos também uma estrutura auxiliar, que mapeia um sobrenome nos índices desse sobrenome no max-heap e no min-heap. Dois modos para fazer esse controle, e que foram aceitos no Lab 7, são busca linear ( $O(n)$ ) e árvore de busca binária ( $O(\log n)$  se balanceada).

Todavia, uma tabela de espalhamento é uma estrutura mais adequada para gerenciar o mapeamento de sobrenomes para índices, pois ela garante operações de inserção, busca e remoção em tempo aproximadamente  $O(1)$ . Dessa forma, revisitaremos o Laboratório 7 e faremos o gerenciamento dos índices utilizando uma tabela de espalhamento. Vocês podem utilizar qualquer um dos tipos de tabela vistos na Aula 26 (e.g., encadeamento separado, endereçamento aberto com sondagem linear) e qualquer uma das funções de espalhamento vistas na mesma aula (e.g., método da divisão, método da multiplicação).

*Dica:* considere criar um `struct` que armazene o sobrenome como chave e os índices dos dois heaps como outros membros desse `struct`.