

# MC202 (Estruturas de Dados) - 1s2021

## Tarefa 12: Tabelas de Espalhamento

Prof<sup>o</sup> Ricardo Dahab

Assistente: Elisa Dell'Arriva

Assistente: Jônatas Trabuco Belotti

Instituto de Computação - UNICAMP

### Sobre a Tarefa

O objetivo desta tarefa é que o(a) aluno(a) se familiarize com conceitos e propriedades relacionados à estrutura de dados Tabela de Espalhamento (**Hash**). Ao lidar com tabelas de espalhamento, existem mais de uma forma de remover um elemento da tabela e métodos distintos podem resultar em tabelas finais distintas. O exercício consiste em popular uma Tabela de Espalhamento e realizar remoções através de dois métodos diferentes.

### O Problema

O problema a ser resolvido é o de implementar as operações de inserção e de dois métodos de remoção em uma estrutura hash de endereçamento aberto com sondagem linear. A descrição desta estrutura bem como suas operações podem ser encontradas nas páginas 142-214 do pdf da Aula 29/06 (<https://classroom.google.com/u/1/c/MjYONzg4MTM3MDcw/m/MzY4MzczODcwMjc4/details>). Os dois métodos de remoção que devem ser implementados são os apresentados nos slides como opções 1 e 2:

1. Opção 1: *Rehash* dos elementos seguintes do bloco.
2. Opção 2: Troca por um valor *dummy*.

Sendo assim, vocês deverão:

- Ler a entrada e construir dois **hashs**: um que não utilize variáveis **dummy** e outro no qual é permitido o uso; em cada um deles será usado um método de remoção diferente.
- Implementar a função de conversão das palavras (*strings*) da entrada para sua representação numérica de acordo com as páginas 101-103 do pdf da Aula 29/06 (<https://classroom.google.com/u/1/c/MjYONzg4MTM3MDcw/m/MzY4MzczODcwMjc4/details>). A variável para armazenar a representação numérica precisa ser do tipo `long long int`.
- Implementar a função hashing dada a seguir:

$$h_{a,b}(k) = ((ak + b) \bmod p) \bmod M \quad (1)$$

em que  $M$  é um primo (longe de uma potência de dois) e é o tamanho do *hash*;  $p$  é um primo maior que  $M$ ;  $a \in \{1, \dots, p\}$  e  $b \in \{0, \dots, p\}$ . Aqui,  $M$ ,  $p$ ,  $a$  e  $b$  fazem parte da entrada.

## Formato da entrada e saída

A entrada é dada por 5 linhas, no seguinte formato:

```
m p a b
n
palavra_1 palavra_2 palavra_3 ... palavra_n-1 palavra_n
k
palavra_1 palavra_2 palavra_3 ... palavra_k-1 palavra_k
```

Em que:

- *m*, *p*, *a* e *b* são o tamanho do *hash* e os parâmetros *p*, *a* e *b*, respectivamente;
- *n* é o número de elementos que devem ser inseridos em ambos os *hashs*;
- *palavra\_1 palavra\_2 ... palavra\_n-1 palavra\_n* as *n* palavras a serem inseridas;
- *k* é o número de elementos que devem ser removidos de ambos os *hashs*;
- *palavra\_1 palavra\_2 ... palavra\_k-1 palavra\_k* as *k* palavras a serem removidas de ambos os *hashs*.

Cada palavra tem no máximo 20 caracteres e o *hash* tem no máximo 50 elementos.

Ao final de todas as operações, seu algoritmo deve imprimir uma tabela com os dois *hashs* no seguinte formato:

```
-----
rehash          | dummy
-----
<rep_numérica>  | <rep_numérica>
VAZIO           | DUMMY
VAZIO           | <rep_numérica>
VAZIO           | DUMMY
<rep_numérica>  | VAZIO
```

Em que:

- Cada linha do conteúdo da tabela tem 45 caracteres, sendo o primeiro caractere um espaço em branco, 20 caracteres para o elemento da Tabela de Espelhamento do tipo *rehash*, 1 espaço em branco, o caractere "|", 1 espaço em branco, 20 caracteres para o elemento da Tabela de Espelhamento do tipo *dummy* e por fim, mais um espaço em branco;
- *<rep\_numérica>* são as representações numéricas das palavras restantes no *hash*;
- *DUMMY* indica que a posição está preenchida com valor *dummy*;
- *VAZIO* indica que a posição está vazia.
- O último caractere da saída deve obrigatoriamente ser um `\n`.

### Exemplo 1.

Entrada:

```
6 7 2 3
4
ola hello mundo world
2
hello world
```

Saída:

rehash	dummy
7302241	7302241
VAZIO	DUMMY
470121604207	470121604207
VAZIO	DUMMY
VAZIO	VAZIO
VAZIO	VAZIO

## Exemplo 2.

Entrada:

```
15 953 753 739
15
Esta e a ultima tarefa de progra macao da disci plina Estru turas de Dados
3
tarefa progra macao
```

Saída:

rehash	dummy
1165194337	DUMMY
482855186017	25697
293691748211	482855186017
97	298289754741
VAZIO	500186636659
VAZIO	25701
VAZIO	293691748211
431265899369	431265899369
25701	129108669984097
25701	1165194337
298289754741	97
500186636659	DUMMY
101	101
25697	25701
129108669984097	DUMMY

## Exemplo 3.

Entrada:

```
7 179 115 54
7
Ultimo caso de teste da ultima tarefa
7
de teste da caso Ultimo tarefa ultima
```

Saída:

```
-----  
rehash          | dummy  
-----  
VAZIO           | DUMMY  
VAZIO           | DUMMY  
VAZIO           | DUMMY  
VAZIO           | DUMMY  
VAZIO           | DUMMY  
VAZIO           | DUMMY  
VAZIO           | DUMMY
```

## Importante

- Todas as operações de inserção e remoção nos *hashs* devem ser implementadas de acordo com o material de Aula disponível no Classroom (<https://classroom.google.com/c/MjYONzg4MTM3MDcw/m/MzY4MzczODcwMjc4/details>);
- Para esta tarefa, o uso de alocação dinâmica de memória não é obrigatório, contudo, se você utilizar esse recurso, toda memória alocada dinamicamente deve ser liberada. Para fazer essa verificação, recomendamos o uso da ferramenta *valgrind*. Caso haja memória não liberada, serão descontados dois pontos da nota final desta tarefa.
- É proibido o uso de variáveis globais. O não cumprimento desta regra implicará em nota 0 (zero) na atividade.
- É proibido o uso de variáveis estáticas (aquelas declaradas com a diretiva *static*). O não cumprimento desta regra implicará em nota 0 (zero) na atividade.

## Alguns avisos e lembretes

- A página da disciplina no SuSy é <https://susy.ic.unicamp.br:9999/mc202abc>.
- Para submeter, utilizem somente os dígitos numéricos do RA e a senha da DAC.
- Espere, no mínimo, 60 segundos entre uma submissão e outra no SuSy, para evitar problemas na entrega.
- O número máximo de submissões é 20.
- Esta tarefa tem peso 2 (dois).
- O prazo de entrega desta tarefa é até 19/07/2021, às 23h59. Serão aceitas submissões atrasadas até o dia 22/07/2021 e serão aplicados os descontos de nota estabelecidos conforme o número de dias em atraso.