

Aula 6

Tabela de Dispersão

Tratamento de Colisão por Encadeamento

Estruturas de Dados Avançadas

Professor Eurinardo Rodrigues Costa
Universidade Federal do Ceará
Campus Russas

2021.1

Sumário

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

1 Aula Passada

2 Tratamento de Colisão por Encadeamento

- Encadeamento Exterior
- Encadeamento Interior
 - Em duas Zonas
 - Em uma única zona

Aula Passada

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Aula Passada

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

- Tabela de Dispersão (Hash)

Aula Passada

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

- Tabela de Dispersão (Hash)
 - Acesso Direto

Aula Passada

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

- Tabela de Dispersão (Hash)
 - Acesso Direto
 - Funções de Dispersão (Hash)

Aula Passada

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

- Tabela de Dispersão (Hash)
 - Acesso Direto
 - Funções de Dispersão (Hash)
 - Método da Divisão

Aula Passada

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

- Tabela de Dispersão (Hash)
 - Acesso Direto
 - Funções de Dispersão (Hash)
 - Método da Divisão
 - Método da Dobra

Aula Passada

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

- Tabela de Dispersão (Hash)
 - Acesso Direto
 - Funções de Dispersão (Hash)
 - Método da Divisão
 - Método da Dobra
 - Método da Multiplicação

Aula Passada

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

- Tabela de Dispersão (Hash)
 - Acesso Direto
 - Funções de Dispersão (Hash)
 - Método da Divisão
 - Método da Dobra
 - Método da Multiplicação
 - Método da Análise de Dígitos

Aula Passada

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

- Tabela de Dispersão (Hash)
 - Acesso Direto
 - Funções de Dispersão (Hash)
 - Método da Divisão
 - Método da Dobra
 - Método da Multiplicação
 - Método da Análise de Dígitos
 - Problema de Colisão

Tratamento de Colisão por Encadeamento

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

**Encadeamento
Exterior**

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Tratamento de Colisão por Encadeamento

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Encadeamento Exterior

Tratamento de Colisão por Encadeamento

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Encadeamento Exterior

Simple aplicação de listas encadeadas em cada espaço da tabela.

Tratamento de Colisão por Encadeamento

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Encadeamento Exterior

Simple aplicação de listas encadeadas em cada espaço da tabela.

Complexidade no pior caso = $O(n)$.

Tratamento de Colisão por Encadeamento

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Encadeamento Exterior

Simple aplicação de listas encadeadas em cada espaço da tabela.

Complexidade no pior caso = $O(n)$.

Encadeamento Interior

Tratamento de Colisão por Encadeamento

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Encadeamento Exterior

Simple aplicação de listas encadeadas em cada espaço da tabela.

Complexidade no pior caso = $O(n)$.

Encadeamento Interior

As listas encadeadas compartilham a mesma memória da tabela. Temos:

Tratamento de Colisão por Encadeamento

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Encadeamento Exterior

Simple aplicação de listas encadeadas em cada espaço da tabela.

Complexidade no pior caso = $O(n)$.

Encadeamento Interior

As listas encadeadas compartilham a mesma memória da tabela. Temos:

- em duas zonas

Tratamento de Colisão por Encadeamento

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Encadeamento Exterior

Simple aplicação de listas encadeadas em cada espaço da tabela.

Complexidade no pior caso = $O(n)$.

Encadeamento Interior

As listas encadeadas compartilham a mesma memória da tabela. Temos:

- em duas zonas e
- em uma única zona.

Encadeamento Interior

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Encadeamento Interior

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Em duas zonas

Encadeamento Interior

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Em duas zonas

A tabela é dividida em **zona de endereços** e **zona de colisões**.

Encadeamento Interior

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Em duas zonas

A tabela é dividida em **zona de endereços** e **zona de colisões**.

Problema: pode ocorrer **falso overflow!!!**

Encadeamento Interior

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Em duas zonas

A tabela é dividida em **zona de endereços** e **zona de colisões**.

Problema: pode ocorrer **falso overflow!!!**

Em uma única zona

Encadeamento Interior

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Em duas zonas

A tabela é dividida em **zona de endereços** e **zona de colisões**.

Problema: pode ocorrer **falso overflow!!!**

Em uma única zona

- Uma única zona para endereços e colisões.

Encadeamento Interior

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Em duas zonas

A tabela é dividida em **zona de endereços** e **zona de colisões**.

Problema: pode ocorrer **falso overflow!!!**

Em uma única zona

- Uma única zona para endereços e colisões.
- Cada elemento da tabela está em uma lista circular.

Encadeamento Interior

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Em duas zonas

A tabela é dividida em **zona de endereços** e **zona de colisões**.

Problema: pode ocorrer **falso overflow!!!**

Em uma única zona

- Uma única zona para endereços e colisões.
- Cada elemento da tabela está em uma lista circular.
- O início da tabela, temos apenas listas circulares.

Encadeamento Interior

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Em duas zonas

A tabela é dividida em **zona de endereços** e **zona de colisões**.

Problema: pode ocorrer **falso overflow!!!**

Em uma única zona

- Uma única zona para endereços e colisões.
- Cada elemento da tabela está em uma lista circular.
- O início da tabela, temos apenas listas circulares.
- Considere os campos:

Encadeamento Interior

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Em duas zonas

A tabela é dividida em **zona de endereços** e **zona de colisões**.

Problema: pode ocorrer **falso overflow!!!**

Em uma única zona

- Uma única zona para endereços e colisões.
- Cada elemento da tabela está em uma lista circular.
- O início da tabela, temos apenas listas circulares.
- Considere os campos: chave,

Encadeamento Interior

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Em duas zonas

A tabela é dividida em **zona de endereços** e **zona de colisões**.

Problema: pode ocorrer **falso overflow!!!**

Em uma única zona

- Uma única zona para endereços e colisões.
- Cada elemento da tabela está em uma lista circular.
- O início da tabela, temos apenas listas circulares.
- Considere os campos: chave, estado

Encadeamento Interior

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Em duas zonas

A tabela é dividida em **zona de endereços** e **zona de colisões**.

Problema: pode ocorrer **falso overflow!!!**

Em uma única zona

- Uma única zona para endereços e colisões.
- Cada elemento da tabela está em uma lista circular.
- O início da tabela, temos apenas listas circulares.
- Considere os campos: chave, estado (**ocupado, não ocupado**)

Encadeamento Interior

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Em duas zonas

A tabela é dividida em **zona de endereços** e **zona de colisões**.

Problema: pode ocorrer **falso overflow!!!**

Em uma única zona

- Uma única zona para endereços e colisões.
- Cada elemento da tabela está em uma lista circular.
- O início da tabela, temos apenas listas circulares.
- Considere os campos: chave, estado (**ocupado, não ocupado**) e pont

Encadeamento Interior

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Em duas zonas

A tabela é dividida em **zona de endereços** e **zona de colisões**.

Problema: pode ocorrer **falso overflow!!!**

Em uma única zona

- Uma única zona para endereços e colisões.
- Cada elemento da tabela está em uma lista circular.
- O início da tabela, temos apenas listas circulares.
- Considere os campos: chave, estado (**ocupado, não ocupado**) e pont (**ponteiro para o próximo**)

Busca em uma única zona

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Algoritmo 1: Buscar(x , end, a)

Busca em uma única zona

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Algoritmo 2: $\text{Buscar}(x, \text{end}, a)$

Entrada: x : chave para ser encontrada

Busca em uma única zona

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Algoritmo 3: Buscar(x , end, a)

Entrada: x : chave para ser encontrada

Saída:

$$j = \begin{cases} \text{compart. não ocupado na lista } h(x) & , \text{ caso exista} \\ \lambda & , \text{ c.c.} \end{cases}$$

Busca em uma única zona

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Algoritmo 4: Buscar(x , end, a)

Entrada: x : chave para ser encontrada

Saída:

$$j = \begin{cases} \text{compart. não ocupado na lista } h(x) & , \text{ caso exista} \\ \lambda & , \text{ c.c.} \end{cases}$$
$$a = \begin{cases} 1 & , \text{ se } x \text{ for encontrado} \\ 2 & , \text{ caso contrário} \end{cases}$$

Busca em uma única zona

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Algoritmo 5: Buscar(x , end, a)

Entrada: x : chave para ser encontrada

Saída:

$$j = \begin{cases} \text{compart. não ocupado na lista } h(x) & , \text{ caso exista} \\ \lambda & , \text{ c.c.} \end{cases}$$
$$a = \begin{cases} 1 & , \text{ se } x \text{ for encontrado} \\ 2 & , \text{ caso contrário} \end{cases}$$
$$\text{end} = \begin{cases} j & , \text{ se chave não encontrada} \\ \text{endereço de } x & , \text{ caso contrário} \end{cases}$$

Busca em uma única zona

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Algoritmo 6: Buscar(x , end, a)

Entrada: x : chave para ser encontrada

Saída:

$$j = \begin{cases} \text{compart. não ocupado na lista } h(x) & , \text{ caso exista} \\ \lambda & , \text{ c.c.} \end{cases}$$
$$a = \begin{cases} 1 & , \text{ se } x \text{ for encontrado} \\ 2 & , \text{ caso contrário} \end{cases}$$
$$\text{end} = \begin{cases} j & , \text{ se chave não encontrada} \\ \text{endereço de } x & , \text{ caso contrário} \end{cases}$$

1 $a \leftarrow 0$;

Busca em uma única zona

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Algoritmo 7: Buscar(x , end, a)

Entrada: x : chave para ser encontrada

Saída:

$$j = \begin{cases} \text{compart. não ocupado na lista } h(x) & , \text{ caso exista} \\ \lambda & , \text{ c.c.} \end{cases}$$
$$a = \begin{cases} 1 & , \text{ se } x \text{ for encontrado} \\ 2 & , \text{ caso contrário} \end{cases}$$
$$\text{end} = \begin{cases} j & , \text{ se chave não encontrada} \\ \text{endereço de } x & , \text{ caso contrário} \end{cases}$$

1 $a \leftarrow 0$; end $\leftarrow h(x)$;

Busca em uma única zona

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Algoritmo 8: Buscar(x , end, a)

Entrada: x : chave para ser encontrada

Saída:

$$j = \begin{cases} \text{compart. não ocupado na lista } h(x) & , \text{ caso exista} \\ \lambda & , \text{ c.c.} \end{cases}$$
$$a = \begin{cases} 1 & , \text{ se } x \text{ for encontrado} \\ 2 & , \text{ caso contrário} \end{cases}$$
$$\text{end} = \begin{cases} j & , \text{ se chave não encontrada} \\ \text{endereço de } x & , \text{ caso contrário} \end{cases}$$

1 $a \leftarrow 0$; end $\leftarrow h(x)$; $j \leftarrow \lambda$;

Busca em uma única zona

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Algoritmo 9: Buscar(x , end, a)

Entrada: x : chave para ser encontrada

Saída:

$$j = \begin{cases} \text{compart. não ocupado na lista } h(x) & , \text{ caso exista} \\ \lambda & , \text{ c.c.} \end{cases}$$
$$a = \begin{cases} 1 & , \text{ se } x \text{ for encontrado} \\ 2 & , \text{ caso contrário} \end{cases}$$
$$\text{end} = \begin{cases} j & , \text{ se chave não encontrada} \\ \text{endereço de } x & , \text{ caso contrário} \end{cases}$$

1 $a \leftarrow 0$; end $\leftarrow h(x)$; $j \leftarrow \lambda$;

2 **enquanto** $a = 0$ **faça**

Busca em uma única zona

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Algoritmo 10: $\text{Buscar}(x, \text{end}, a)$

Entrada: x : chave para ser encontrada

Saída:

$$j = \begin{cases} \text{compart. não ocupado na lista } h(x) & , \text{ caso exista} \\ \lambda & , \text{ c.c.} \end{cases}$$
$$a = \begin{cases} 1 & , \text{ se } x \text{ for encontrado} \\ 2 & , \text{ caso contrário} \end{cases}$$
$$\text{end} = \begin{cases} j & , \text{ se chave não encontrada} \\ \text{endereço de } x & , \text{ caso contrário} \end{cases}$$

1 $a \leftarrow 0$; $\text{end} \leftarrow h(x)$; $j \leftarrow \lambda$;

2 **enquanto** $a = 0$ **faça**

3 **se** $T[\text{end}].\text{estado} = \text{não ocupado}$ **então**

 |

Busca em uma única zona

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Algoritmo 11: $\text{Buscar}(x, \text{end}, a)$

Entrada: x : chave para ser encontrada**Saída:**
$$j = \begin{cases} \text{compart. não ocupado na lista } h(x) & , \text{ caso exista} \\ \lambda & , \text{ c.c.} \end{cases}$$
$$a = \begin{cases} 1 & , \text{ se } x \text{ for encontrado} \\ 2 & , \text{ caso contrário} \end{cases}$$
$$\text{end} = \begin{cases} j & , \text{ se chave não encontrada} \\ \text{endereço de } x & , \text{ caso contrário} \end{cases}$$

1 $a \leftarrow 0$; $\text{end} \leftarrow h(x)$; $j \leftarrow \lambda$;

2 **enquanto** $a = 0$ **faça**

3 **se** $T[\text{end}].\text{estado} = \text{não ocupado}$ **então**

4 $j \leftarrow \text{end}$

Busca em uma única zona

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Algoritmo 12: Buscar(x , end, a)

Entrada: x : chave para ser encontrada**Saída:**
$$j = \begin{cases} \text{compart. não ocupado na lista } h(x) & , \text{ caso exista} \\ \lambda & , \text{ c.c.} \end{cases}$$
$$a = \begin{cases} 1 & , \text{ se } x \text{ for encontrado} \\ 2 & , \text{ caso contrário} \end{cases}$$
$$\text{end} = \begin{cases} j & , \text{ se chave não encontrada} \\ \text{endereço de } x & , \text{ caso contrário} \end{cases}$$

1 $a \leftarrow 0$; end $\leftarrow h(x)$; $j \leftarrow \lambda$;

2 **enquanto** $a = 0$ **faça**

3 **se** $T[\text{end}].\text{estado} = \text{não ocupado}$ **então**

4 $j \leftarrow \text{end}$

5 **se** $T[\text{end}].\text{chave} = x$ **e** $T[\text{end}].\text{estado} = \text{ocupado}$
 então

 |

Busca em uma única zona

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Algoritmo 13: Buscar(x , end, a)

Entrada: x : chave para ser encontrada**Saída:**
$$j = \begin{cases} \text{compart. não ocupado na lista } h(x) & , \text{ caso exista} \\ \lambda & , \text{ c.c.} \end{cases}$$
$$a = \begin{cases} 1 & , \text{ se } x \text{ for encontrado} \\ 2 & , \text{ caso contrário} \end{cases}$$
$$\text{end} = \begin{cases} j & , \text{ se chave não encontrada} \\ \text{endereço de } x & , \text{ caso contrário} \end{cases}$$

1 $a \leftarrow 0$; end $\leftarrow h(x)$; $j \leftarrow \lambda$;

2 **enquanto** $a = 0$ **faça**

3 **se** $T[\text{end}].\text{estado} = \text{não ocupado}$ **então**

4 $j \leftarrow \text{end}$

5 **se** $T[\text{end}].\text{chave} = x$ **e** $T[\text{end}].\text{estado} = \text{ocupado}$
 então

6 $a \leftarrow 1$

Busca em uma única zona

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Algoritmo 14: Buscar(x , end, a)

Entrada: x : chave para ser encontrada**Saída:**
$$j = \begin{cases} \text{compart. não ocupado na lista } h(x) & , \text{ caso exista} \\ \lambda & , \text{ c.c.} \end{cases}$$
$$a = \begin{cases} 1 & , \text{ se } x \text{ for encontrado} \\ 2 & , \text{ caso contrário} \end{cases}$$
$$\text{end} = \begin{cases} j & , \text{ se chave não encontrada} \\ \text{endereço de } x & , \text{ caso contrário} \end{cases}$$

1 $a \leftarrow 0$; end $\leftarrow h(x)$; $j \leftarrow \lambda$;

2 **enquanto** $a = 0$ **faça**

3 **se** $T[\text{end}].\text{estado} = \text{não ocupado}$ **então**

4 $j \leftarrow \text{end}$

5 **se** $T[\text{end}].\text{chave} = x$ **e** $T[\text{end}].\text{estado} = \text{ocupado}$
6 **então**

6 $a \leftarrow 1$ %chave encontrada

Busca em uma única zona

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Algoritmo 15: Buscar(x , end, a)

Entrada: x : chave para ser encontrada**Saída:**
$$j = \begin{cases} \text{compart. não ocupado na lista } h(x) & , \text{ caso exista} \\ \lambda & , \text{ c.c.} \end{cases}$$
$$a = \begin{cases} 1 & , \text{ se } x \text{ for encontrado} \\ 2 & , \text{ caso contrário} \end{cases}$$
$$\text{end} = \begin{cases} j & , \text{ se chave não encontrada} \\ \text{endereço de } x & , \text{ caso contrário} \end{cases}$$

1 $a \leftarrow 0$; end $\leftarrow h(x)$; $j \leftarrow \lambda$;

2 **enquanto** $a = 0$ **faça**

3 **se** $T[\text{end}].\text{estado} = \text{não ocupado}$ **então**

4 $j \leftarrow \text{end}$

5 **se** $T[\text{end}].\text{chave} = x$ **e** $T[\text{end}].\text{estado} = \text{ocupado}$
 então

6 $a \leftarrow 1$ %chave encontrada

7 **senão**

Busca em uma única zona

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Algoritmo 16: Buscar(x , end, a)

Entrada: x : chave para ser encontrada**Saída:**
$$j = \begin{cases} \text{compart. não ocupado na lista } h(x) & , \text{ caso exista} \\ \lambda & , \text{ c.c.} \end{cases}$$
$$a = \begin{cases} 1 & , \text{ se } x \text{ for encontrado} \\ 2 & , \text{ caso contrário} \end{cases}$$
$$\text{end} = \begin{cases} j & , \text{ se chave não encontrada} \\ \text{endereço de } x & , \text{ caso contrário} \end{cases}$$

1 $a \leftarrow 0$; end $\leftarrow h(x)$; $j \leftarrow \lambda$;

2 **enquanto** $a = 0$ **faça**

3 **se** $T[\text{end}].\text{estado} = \text{não ocupado}$ **então**

4 $j \leftarrow \text{end}$

5 **se** $T[\text{end}].\text{chave} = x$ **e** $T[\text{end}].\text{estado} = \text{ocupado}$
 então

6 $a \leftarrow 1$ %chave encontrada

7 **senão**

8 end $\leftarrow T[\text{end}].\text{pont}$

Busca em uma única zona

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Algoritmo 17: Buscar(x , end, a)

Entrada: x : chave para ser encontrada**Saída:**
$$j = \begin{cases} \text{compart. não ocupado na lista } h(x) & , \text{ caso exista} \\ \lambda & , \text{ c.c.} \end{cases}$$
$$a = \begin{cases} 1 & , \text{ se } x \text{ for encontrado} \\ 2 & , \text{ caso contrário} \end{cases}$$
$$\text{end} = \begin{cases} j & , \text{ se chave não encontrada} \\ \text{endereço de } x & , \text{ caso contrário} \end{cases}$$

```
1   $a \leftarrow 0$ ;  $\text{end} \leftarrow h(x)$ ;  $j \leftarrow \lambda$ ;  
2  enquanto  $a = 0$  faça  
3      se  $T[\text{end}].\text{estado} = \text{não ocupado}$  então  
4           $j \leftarrow \text{end}$   
5      se  $T[\text{end}].\text{chave} = x$  e  $T[\text{end}].\text{estado} = \text{ocupado}$   
        então  
6           $a \leftarrow 1$  %chave encontrada  
7      senão  
8           $\text{end} \leftarrow T[\text{end}].\text{pont}$   
9      se  $\text{end} = h(x)$  então  
          |
```

Busca em uma única zona

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Algoritmo 18: Buscar(x , end, a)

Entrada: x : chave para ser encontrada**Saída:**
$$j = \begin{cases} \text{compart. não ocupado na lista } h(x) & , \text{ caso exista} \\ \lambda & , \text{ c.c.} \end{cases}$$
$$a = \begin{cases} 1 & , \text{ se } x \text{ for encontrado} \\ 2 & , \text{ caso contrário} \end{cases}$$
$$\text{end} = \begin{cases} j & , \text{ se chave não encontrada} \\ \text{endereço de } x & , \text{ caso contrário} \end{cases}$$

```
1   $a \leftarrow 0$ ;  $\text{end} \leftarrow h(x)$ ;  $j \leftarrow \lambda$ ;  
2  enquanto  $a = 0$  faça  
3      se  $T[\text{end}].\text{estado} = \text{não ocupado}$  então  
4           $j \leftarrow \text{end}$   
5      se  $T[\text{end}].\text{chave} = x$  e  $T[\text{end}].\text{estado} = \text{ocupado}$   
        então  
6           $a \leftarrow 1$  %chave encontrada  
7      senão  
8           $\text{end} \leftarrow T[\text{end}].\text{pont}$   
9          se  $\text{end} = h(x)$  então  
10          $a \leftarrow 2$ ;  $\text{end} \leftarrow j$ ;
```

Busca em uma única zona

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Algoritmo 19: Buscar(x , end, a)

Entrada: x : chave para ser encontrada**Saída:**
$$j = \begin{cases} \text{compart. não ocupado na lista } h(x) & , \text{ caso exista} \\ \lambda & , \text{ c.c.} \end{cases}$$
$$a = \begin{cases} 1 & , \text{ se } x \text{ for encontrado} \\ 2 & , \text{ caso contrário} \end{cases}$$
$$\text{end} = \begin{cases} j & , \text{ se chave não encontrada} \\ \text{endereço de } x & , \text{ caso contrário} \end{cases}$$

```
1   $a \leftarrow 0$ ;  $\text{end} \leftarrow h(x)$ ;  $j \leftarrow \lambda$ ;  
2  enquanto  $a = 0$  faça  
3      se  $T[\text{end}].\text{estado} = \text{não ocupado}$  então  
4           $j \leftarrow \text{end}$   
5      se  $T[\text{end}].\text{chave} = x$  e  $T[\text{end}].\text{estado} = \text{ocupado}$   
        então  
6           $a \leftarrow 1$  %chave encontrada  
7      senão  
8           $\text{end} \leftarrow T[\text{end}].\text{pont}$   
9          se  $\text{end} = h(x)$  então  
10              $a \leftarrow 2$ ;  $\text{end} \leftarrow j$ ; %não chave encontrada
```

Remoção em uma única zona

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Algoritmo 20: $\text{Remover}(x, \text{end}, a)$

Remoção em uma única zona

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Algoritmo 21: $\text{Remover}(x, \text{end}, a)$

1 $\text{Buscar}(x, \text{end}, a)$

Remoção em uma única zona

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Algoritmo 22: Remover(x , end, a)

- 1 Buscar(x , end, a)
- 2 **se** $a = 1$ **então**
|

Remoção em uma única zona

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Algoritmo 23: Remover(x , end, a)

- 1 Buscar(x , end, a)
- 2 **se** $a = 1$ **então**
- 3 $T[\text{end}].\text{estado} \leftarrow$ não ocupado

Remoção em uma única zona

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Algoritmo 24: Remover(x , end, a)

```
1  Buscar( $x$ , end,  $a$ )
2  se  $a = 1$  então
3    |    $T[\text{end}].\text{estado} \leftarrow$  não ocupado
4  senão
    |
```

Remoção em uma única zona

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Algoritmo 25: Remover(x , end, a)

- 1 Buscar(x , end, a)
- 2 **se** $a = 1$ **então**
- 3 $T[\text{end}].\text{estado} \leftarrow$ não ocupado
- 4 **senão**
- 5 “Remoção inválida”

Remoção em uma única zona

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Algoritmo 26: Remover(x , end, a)

- 1 Buscar(x , end, a)
 - 2 **se** $a = 1$ **então**
 - 3 $T[\text{end}].\text{estado} \leftarrow$ não ocupado
 - 4 **senão**
 - 5 “Remoção inválida”
-

Inserção em uma única zona

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Algoritmo 27: Inserir(x , end, a)

Inserção em uma única zona

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Algoritmo 28: Inserir(x , end, a)

1 Buscar(x , end, a)

Inserção em uma única zona

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Algoritmo 29: Inserir(x , end, a)

- 1 Buscar(x , end, a)
- 2 **se** $a = 1$ **então**

Inserção em uma única zona

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Algoritmo 30: Inserir(x , end, a)

- 1 Buscar(x , end, a)
- 2 **se** $a = 1$ **então** "Inserção inválida: chave já existe"

Inserção em uma única zona

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Algoritmo 31: Inserir(x , end, a)

- 1 Buscar(x , end, a)
- 2 **se** $a = 1$ **então** “Inserção inválida: chave já existe”;
- 3 **senão**

Inserção em uma única zona

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Algoritmo 32: Inserir(x , end, a)

- 1 Buscar(x , end, a)
- 2 **se** $a = 1$ **então** "Inserção inválida: chave já existe";
- 3 **senão**
- 4 **se** $end \neq \lambda$ **então**

Inserção em uma única zona

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Algoritmo 33: Inserir(x , end, a)

- 1 Buscar(x , end, a)
- 2 **se** $a = 1$ **então** “Inserção inválida: chave já existe”;
- 3 **senão**
- 4 **se** $end \neq \lambda$ **então** $j \leftarrow end$

Inserção em uma única zona

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Algoritmo 34: Inserir(x , end, a)

```
1  Buscar( $x$ , end,  $a$ )
2  se  $a = 1$  então "Inserção inválida: chave já existe";
3  senão
4      se  $end \neq \lambda$  então  $j \leftarrow end$  ;
5      senão
```

Inserção em uma única zona

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Algoritmo 35: Inserir(x , end, a)

```
1  Buscar( $x$ , end,  $a$ )
2  se  $a = 1$  então "Inserção inválida: chave já existe";
3  senão
4      se  $end \neq \lambda$  então  $j \leftarrow end$  ;
5      senão
6           $i \leftarrow 1$ ;
```

Inserção em uma única zona

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Algoritmo 36: Inserir(x , end, a)

```
1  Buscar( $x$ , end,  $a$ )
2  se  $a = 1$  então "Inserção inválida: chave já existe";
3  senão
4      se  $end \neq \lambda$  então  $j \leftarrow end$  ;
5      senão
6           $i \leftarrow 1; j \leftarrow h(x)$ 
```

Inserção em uma única zona

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Algoritmo 37: Inserir(x , end, a)

```
1  Buscar( $x$ , end,  $a$ )
2  se  $a = 1$  então "Inserção inválida: chave já existe";
3  senão
4      se  $end \neq \lambda$  então  $j \leftarrow end$  ;
5      senão
6           $i \leftarrow 1; j \leftarrow h(x)$ 
7          enquanto  $i \leq m$  faça
```

Inserção em uma única zona

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Algoritmo 38: Inserir(x , end, a)

```
1  Buscar( $x$ , end,  $a$ )
2  se  $a = 1$  então "Inserção inválida: chave já existe";
3  senão
4      se  $end \neq \lambda$  então  $j \leftarrow end$  ;
5      senão
6           $i \leftarrow 1; j \leftarrow h(x)$ 
7          enquanto  $i \leq m$  faça
8              se  $t[j].estado = \text{ocupado}$  então
```

Inserção em uma única zona

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Algoritmo 39: Inserir(x , end, a)

```
1  Buscar( $x$ , end,  $a$ )
2  se  $a = 1$  então "Inserção inválida: chave já existe";
3  senão
4      se  $end \neq \lambda$  então  $j \leftarrow end$  ;
5      senão
6           $i \leftarrow 1; j \leftarrow h(x)$ 
7          enquanto  $i \leq m$  faça
8              se  $t[j].estado = \text{ocupado}$  então
9                   $j \leftarrow (j + 1) \bmod m$ 
```


Inserção em uma única zona

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Algoritmo 40: Inserir(x , end, a)

```
1  Buscar( $x$ , end,  $a$ )
2  se  $a = 1$  então "Inserção inválida: chave já existe";
3  senão
4      se  $end \neq \lambda$  então  $j \leftarrow end$  ;
5      senão
6           $i \leftarrow 1; j \leftarrow h(x)$ 
7          enquanto  $i \leq m$  faça
8              se  $t[j].estado = \text{ocupado}$  então
9                   $j \leftarrow (j + 1) \bmod m$ 
10                  $i \leftarrow i + 1$ 
```

Inserção em uma única zona

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Algoritmo 41: Inserir(x , end, a)

```
1  Buscar( $x$ , end,  $a$ )
2  se  $a = 1$  então "Inserção inválida: chave já existe";
3  senão
4      se  $end \neq \lambda$  então  $j \leftarrow end$  ;
5      senão
6           $i \leftarrow 1; j \leftarrow h(x)$ 
7          enquanto  $i \leq m$  faça
8              se  $t[j].estado = \text{ocupado}$  então
9                   $j \leftarrow (j + 1) \bmod m$ 
10                  $i \leftarrow i + 1$ 
11             senão
```

Inserção em uma única zona

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Algoritmo 42: Inserir(x , end, a)

```
1  Buscar( $x$ , end,  $a$ )
2  se  $a = 1$  então "Inserção inválida: chave já existe";
3  senão
4      se  $end \neq \lambda$  então  $j \leftarrow end$  ;
5      senão
6           $i \leftarrow 1; j \leftarrow h(x)$ 
7          enquanto  $i \leq m$  faça
8              se  $t[j].estado = \text{ocupado}$  então
9                   $j \leftarrow (j + 1) \bmod m$ 
10                  $i \leftarrow i + 1$ 
11             senão
12                  $i \leftarrow m + 2$ 
```

Inserção em uma única zona

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Algoritmo 43: Inserir(x , end, a)

```
1  Buscar( $x$ , end,  $a$ )
2  se  $a = 1$  então "Inserção inválida: chave já existe";
3  senão
4      se  $end \neq \lambda$  então  $j \leftarrow end$  ;
5      senão
6           $i \leftarrow 1; j \leftarrow h(x)$ 
7          enquanto  $i \leq m$  faça
8              se  $t[j].estado = \text{ocupado}$  então
9                   $j \leftarrow (j + 1) \bmod m$ 
10                  $i \leftarrow i + 1$ 
11             senão
12                  $i \leftarrow m + 2$ 
```

Inserção em uma única zona

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Algoritmo 44: Inserir(x , end, a)

```
1  Buscar( $x$ , end,  $a$ )
2  se  $a = 1$  então "Inserção inválida: chave já existe";
3  senão
4      se  $end \neq \lambda$  então  $j \leftarrow end$  ;
5      senão
6           $i \leftarrow 1; j \leftarrow h(x)$ 
7          enquanto  $i \leq m$  faça
8              se  $t[j].estado = \text{ocupado}$  então
9                   $j \leftarrow (j + 1) \bmod m$ 
10                  $i \leftarrow i + 1$ 
11             senão
12                  $i \leftarrow m + 2$ 
13     se  $i = m + 1$  então "overflow";
```

Inserção em uma única zona

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Algoritmo 45: Inserir(x , end, a)

```
1  Buscar( $x$ , end,  $a$ )
2  se  $a = 1$  então "Inserção inválida: chave já existe";
3  senão
4      se  $end \neq \lambda$  então  $j \leftarrow end$  ;
5      senão
6           $i \leftarrow 1; j \leftarrow h(x)$ 
7          enquanto  $i \leq m$  faça
8              se  $t[j].estado = \text{ocupado}$  então
9                   $j \leftarrow (j + 1) \bmod m$ 
10                  $i \leftarrow i + 1$ 
11             senão
12                  $i \leftarrow m + 2$ 
13     se  $i = m + 1$  então "overflow"; pare
```

Inserção em uma única zona

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Algoritmo 46: Inserir(x , end, a)

```
1  Buscar( $x$ , end,  $a$ )
2  se  $a = 1$  então "Inserção inválida: chave já existe";
3  senão
4      se  $end \neq \lambda$  então  $j \leftarrow end$  ;
5      senão
6           $i \leftarrow 1; j \leftarrow h(x)$ 
7          enquanto  $i \leq m$  faça
8              se  $t[j].estado = \text{ocupado}$  então
9                   $j \leftarrow (j + 1) \bmod m$ 
10                  $i \leftarrow i + 1$ 
11             senão
12                  $i \leftarrow m + 2$ 
13         se  $i = m + 1$  então "overflow"; pare ;
14          $temp \leftarrow T[j].pont$  %fusão de listas
15          $T[j].pont \leftarrow T[h(x)].pont$ 
16          $T[h(x)].pont \leftarrow temp$ 
```

Inserção em uma única zona

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona

Algoritmo 47: Inserir(x , end, a)

```
1  Buscar( $x$ , end,  $a$ )
2  se  $a = 1$  então "Inserção inválida: chave já existe";
3  senão
4      se  $end \neq \lambda$  então  $j \leftarrow end$  ;
5      senão
6           $i \leftarrow 1; j \leftarrow h(x)$ 
7          enquanto  $i \leq m$  faça
8              se  $t[j].estado = \text{ocupado}$  então
9                   $j \leftarrow (j + 1) \bmod m$ 
10                  $i \leftarrow i + 1$ 
11             senão
12                  $i \leftarrow m + 2$ 
13         se  $i = m + 1$  então "overflow"; pare ;
14          $temp \leftarrow T[j].pont$  %fusão de listas
15          $T[j].pont \leftarrow T[h(x)].pont$ 
16          $T[h(x)].pont \leftarrow temp$ 
17      $T[j].chave \leftarrow x$  %inserção da chave
18      $T[j].estado \leftarrow \text{ocupado}$ 
```

Bibliografia

EDA - Aula 6

Prof.
Eurinardo

Aula Passada

Tratamento de
Colisão por
Encadea-
mento

Encadeamento
Exterior

Encadeamento
Interior

Em duas Zonas

Em uma única zona



SZWARCFITER , Jayme; MARKENZON, Lilian.
Estruturas de Dados e Seus Algoritmos.3a edição.
LTC, ano 2010. (ISBN 9788521617501).

Obrigado!