

MATA54 - Estruturas de Dados e Algoritmos II

Hashing - Endereçamento Aberto

Flávio Assis

Versão gerada a partir de slides do Prof. George Lima

IC - Instituto de Computação

Salvador, agosto de 2021

Hashing com Endereçamento Aberto

- ▶ **Motivação:** aproveitamento de espaço
- ▶ Uma **função hash** é usada para **encadeamento**
- ▶ Ponteiros explícitos **não são** necessários

Primeira Abordagem: Sondagem Linear (Linear Probing)

- ▶ Caso haja colisão, insere-se o registro na primeira posição vazia, considerando o arquivo como uma estrutura circular
- ▶ Ex.: 27, 18, 29, 43, 77, 16, 40, 49 e 5; $m = 11$; $h(k) = k \bmod 11$

End	Reg.
0:	
1:	
2:	
3:	
4:	
5:	
6:	
7:	
8:	
9:	
10:	

Primeira Abordagem: Sondagem Linear (Linear Probing)

- ▶ Caso haja colisão, insere-se o registro na primeira posição vazia, considerando o arquivo como uma estrutura circular
- ▶ Ex.: 27, 18, 29, 43, 77, 16, 40, 49 e 5; $m = 11$; $h(k) = k \bmod 11$

End	Reg.	End	Reg.
0:		0:	
1:		1:	
2:		2:	
3:		3:	
4:		4:	
5:		5:	27
6:		6:	
7:		7:	
8:		8:	
9:		9:	
10:		10:	

Primeira Abordagem: Sondagem Linear (Linear Probing)

- ▶ Caso haja colisão, insere-se o registro na primeira posição vazia, considerando o arquivo como uma estrutura circular
- ▶ Ex.: 27, 18, 29, 43, 77, 16, 40, 49 e 5; $m = 11$; $h(k) = k \bmod 11$

End	Reg.	End	Reg.	End	Reg.
0:		0:		0:	
1:		1:		1:	
2:		2:		2:	
3:		3:		3:	
4:		4:		4:	
5:		5:	27	5:	27
6:		6:		6:	
7:		7:		7:	18
8:		8:		8:	
9:		9:		9:	
10:		10:		10:	

Primeira Abordagem: Sondagem Linear (Linear Probing)

- ▶ Caso haja colisão, insere-se o registro na primeira posição vazia, considerando o arquivo como uma estrutura circular
- ▶ Ex.: 27, 18, 29, 43, 77, 16, 40, 49 e 5; $m = 11$; $h(k) = k \bmod 11$

End	Reg.	End	Reg.	End	Reg.	End	Reg.
0:		0:		0:		0:	
1:		1:		1:		1:	
2:		2:		2:		2:	
3:		3:		3:		3:	
4:		4:		4:		4:	
5:		5:	27	5:	27	5:	27
6:		6:		6:		6:	
7:		7:		7:	18	7:	18
8:		8:		8:		8:	29
9:		9:		9:		9:	
10:		10:		10:		10:	

Primeira Abordagem: Sondagem Linear (Linear Probing)

- Caso haja colisão, insere-se o registro na primeira posição vazia, considerando o arquivo como uma estrutura circular
- Ex.: 27, 18, 29, 43, 77, 16, 40, 49 e 5; $m = 11$; $h(k) = k \bmod 11$

End	Reg.
0:	
1:	
2:	
3:	
4:	
5:	
6:	
7:	
8:	
9:	
10:	

End	Reg.
0:	
1:	
2:	
3:	
4:	
5:	27
6:	
7:	
8:	
9:	
10:	

End	Reg.
0:	
1:	
2:	
3:	
4:	
5:	27
6:	
7:	18
8:	
9:	
10:	

End	Reg.
0:	
1:	
2:	
3:	
4:	
5:	27
6:	
7:	18
8:	29
9:	
10:	

End	Reg.
0:	
1:	
2:	
3:	
4:	
5:	27
6:	
7:	18
8:	29
9:	
10:	43

Primeira Abordagem: Sondagem Linear (Linear Probing)

- ▶ Caso haja colisão, insere-se o registro na primeira posição vazia, considerando o arquivo como uma estrutura circular
- ▶ Ex.: 27, 18, 29, 43, 77, 16, 40, 49 e 5; $m = 11$; $h(k) = k \bmod 11$

End	Reg.
0:	77
1:	
2:	
3:	
4:	
5:	27
6:	
7:	18
8:	29
9:	
10:	43

Primeira Abordagem: Sondagem Linear (Linear Probing)

- ▶ Caso haja colisão, insere-se o registro na primeira posição vazia, considerando o arquivo como uma estrutura circular
- ▶ Ex.: 27, 18, 29, 43, 77, 16, 40, 49 e 5; $m = 11$; $h(k) = k \bmod 11$

End	Reg.	End	Reg.
0:	77	0:	77
1:		1:	
2:		2:	
3:		3:	
4:		4:	
5:	27	5:	27
6:		6:	16
7:	18	7:	18
8:	29	8:	29
9:		9:	
10:	43	10:	43

Primeira Abordagem: Sondagem Linear (Linear Probing)

- Caso haja colisão, insere-se o registro na primeira posição vazia, considerando o arquivo como uma estrutura circular
- Ex.: 27, 18, 29, 43, 77, 16, 40, 49 e 5; $m = 11$; $h(k) = k \bmod 11$

End	Reg.	End	Reg.	End	Reg.
0:	77	0:	77	0:	77
1:		1:		1:	
2:		2:		2:	
3:		3:		3:	
4:		4:		4:	
5:	27	5:	27	5:	27
6:		6:	16	6:	16
7:	18	7:	18	7:	18
8:	29	8:	29	8:	29
9:		9:		9:	40
10:	43	10:	43	10:	43

Primeira Abordagem: Sondagem Linear (Linear Probing)

- Caso haja colisão, insere-se o registro na primeira posição vazia, considerando o arquivo como uma estrutura circular
- Ex.: 27, 18, 29, 43, 77, 16, 40, 49 e 5; $m = 11$; $h(k) = k \bmod 11$

End	Reg.	End	Reg.	End	Reg.	End	Reg.
0:	77	0:	77	0:	77	0:	77
1:		1:		1:		1:	49
2:		2:		2:		2:	
3:		3:		3:		3:	
4:		4:		4:		4:	
5:	27	5:	27	5:	27	5:	27
6:		6:	16	6:	16	6:	16
7:	18	7:	18	7:	18	7:	18
8:	29	8:	29	8:	29	8:	29
9:		9:		9:	40	9:	40
10:	43	10:	43	10:	43	10:	43

Primeira Abordagem: Sondagem Linear (Linear Probing)

- Caso haja colisão, insere-se o registro na primeira posição vazia, considerando o arquivo como uma estrutura circular
- Ex.: 27, 18, 29, 43, 77, 16, 40, 49 e 5; $m = 11$; $h(k) = k \bmod 11$

End	Reg.
0:	77
1:	
2:	
3:	
4:	
5:	27
6:	
7:	18
8:	29
9:	
10:	43

End	Reg.
0:	77
1:	
2:	
3:	
4:	
5:	27
6:	16
7:	18
8:	29
9:	
10:	43

End	Reg.
0:	77
1:	
2:	
3:	
4:	
5:	27
6:	16
7:	18
8:	29
9:	40
10:	43

End	Reg.
0:	77
1:	49
2:	
3:	
4:	
5:	27
6:	16
7:	18
8:	29
9:	40
10:	43

End	Reg.
0:	77
1:	49
2:	5
3:	
4:	
5:	27
6:	16
7:	18
8:	29
9:	40
10:	43

Sondagem Linear

- ▶ Média de acessos ao arquivo para este conjunto de chaves:

$$\frac{1+1+2+1+1+2+3+6+9}{9} = 2,9$$

- ▶ Qual seria a média de acessos para o caso de encadeamento explícito?
- ▶ Sondagem linear tem um desempenho ruim - gera **clusterização primária**

Sondagem Linear: Remoção de Registros

- ▶ Como remover registros? Marcando registros como **logicamente** apagados
- ▶ Remoção do 43:

End	Reg.
0:	77
1:	49
2:	5
3:	
4:	
5:	27
6:	16
7:	18
8:	29
9:	40
10:	*

O marcador * indica que a posição está livre, mas já foi ocupada. A busca por uma chave não deve parar nesta posição.

Como tentar melhorar?

Hashing Duplo

- ▶ Espalhar as chaves na ocorrência de colisão
- ▶ Uma **função hash** é usada para **encadeamento**

$$h(k, i) = (h_1(k) + i h_2(k)) \bmod m \quad h_2(k) = \text{depende do método}$$

i é o número de sondagens já realizadas

Ex:

- ▶ Sondagem linear: $h_1(k) = k \bmod m$ e $h_2(k) = 1$
- ▶ Hashing duplo:

$$h_1(k) = k \bmod m \quad \text{e} \quad h_2(k) = \begin{cases} 1 & \text{se } k < m \\ \lfloor \frac{k}{m} \rfloor & \text{se } k \geq m \end{cases}$$

Busca sequencial caso haja colisão

- Ex.: 27, 18, 29, 43, 77, 16, 40, 49 e 5; $m = 11$

$$h_1(k) = k \bmod m \quad \text{e} \quad h_2(k) = 1$$

End	Reg.
0:	77
1:	49
2:	5
3:	
4:	
5:	27
6:	16
7:	18
8:	29
9:	40
10:	43

Hashing Duplo

Busca com saltos caso haja colisão (espalhamento das chaves)

► Ex.: 27, 18, 29, 43, 77, 16, 40, 49 e 5; $m = 11$

$$h_1(k) = k \bmod m \quad \text{e} \quad h_2(k) = \begin{cases} 1 & \text{se } k < m \\ \lfloor \frac{k}{m} \rfloor & \text{se } k \geq m \end{cases}$$

End	Reg.
0:	
1:	
2:	
3:	
4:	
5:	
6:	
7:	
8:	
9:	
10:	

Hashing Duplo

Busca com saltos caso haja colisão (espalhamento das chaves)

► Ex.: 27, 18, 29, 43, 77, 16, 40, 49 e 5; $m = 11$

$$h_1(k) = k \bmod m \quad \text{e} \quad h_2(k) = \begin{cases} 1 & \text{se } k < m \\ \lfloor \frac{k}{m} \rfloor & \text{se } k \geq m \end{cases}$$

End	Reg.	End	Reg.
0:		0:	
1:		1:	
2:		2:	
3:		3:	
4:		4:	
5:		5:	27
6:		6:	
7:		7:	
8:		8:	
9:		9:	
10:		10:	

Hashing Duplo

Busca com saltos caso haja colisão (espalhamento das chaves)

► Ex.: 27, 18, 29, 43, 77, 16, 40, 49 e 5; $m = 11$

$$h_1(k) = k \bmod m \quad \text{e} \quad h_2(k) = \begin{cases} 1 & \text{se } k < m \\ \lfloor \frac{k}{m} \rfloor & \text{se } k \geq m \end{cases}$$

End	Reg.	End	Reg.	End	Reg.
0:		0:		0:	
1:		1:		1:	
2:		2:		2:	
3:		3:		3:	
4:		4:		4:	
5:		5:	27	5:	27
6:		6:		6:	
7:		7:		7:	18
8:		8:		8:	
9:		9:		9:	
10:		10:		10:	

Hashing Duplo

Busca com saltos caso haja colisão (espalhamento das chaves)

► Ex.: 27, 18, 29, 43, 77, 16, 40, 49 e 5; $m = 11$

$$h_1(k) = k \bmod m \quad \text{e} \quad h_2(k) = \begin{cases} 1 & \text{se } k < m \\ \lfloor \frac{k}{m} \rfloor & \text{se } k \geq m \end{cases}$$

End	Reg.	End	Reg.	End	Reg.	End	Reg.
0:		0:		0:		0:	
1:		1:		1:		1:	
2:		2:		2:		2:	
3:		3:		3:		3:	
4:		4:		4:		4:	
5:		5:	27	5:	27	5:	27
6:		6:		6:		6:	
7:		7:		7:	18	7:	18
8:		8:		8:		8:	
9:		9:		9:		9:	29
10:		10:		10:		10:	

Hashing Duplo

Busca com saltos caso haja colisão (espalhamento das chaves)

► Ex.: 27, 18, 29, 43, 77, 16, 40, 49 e 5; $m = 11$

$$h_1(k) = k \bmod m \quad \text{e} \quad h_2(k) = \begin{cases} 1 & \text{se } k < m \\ \lfloor \frac{k}{m} \rfloor & \text{se } k \geq m \end{cases}$$

End	Reg.
0:	
1:	
2:	
3:	
4:	
5:	
6:	
7:	
8:	
9:	
10:	

End	Reg.
0:	
1:	
2:	
3:	
4:	
5:	27
6:	
7:	
8:	
9:	
10:	

End	Reg.
0:	
1:	
2:	
3:	
4:	
5:	27
6:	
7:	18
8:	
9:	
10:	

End	Reg.
0:	
1:	
2:	
3:	
4:	
5:	27
6:	
7:	18
8:	
9:	29
10:	

End	Reg.
0:	
1:	
2:	
3:	
4:	
5:	27
6:	
7:	18
8:	
9:	29
10:	43

Hashing Duplo

Busca com saltos caso haja colisão (espalhamento das chaves)

► Ex.: 27, 18, 29, 43, 77, 16, 40, 49 e 5; $m = 11$

$$h_1(k) = k \bmod m \quad \text{e} \quad h_2(k) = \begin{cases} 1 & \text{se } k < m \\ \lfloor \frac{k}{m} \rfloor & \text{se } k \geq m \end{cases}$$

End	Reg.
0:	77
1:	
2:	
3:	
4:	
5:	27
6:	
7:	18
8:	
9:	29
10:	43

Hashing Duplo

Busca com saltos caso haja colisão (espalhamento das chaves)

► Ex.: 27, 18, 29, 43, 77, 16, 40, 49 e 5; $m = 11$

$$h_1(k) = k \bmod m \quad \text{e} \quad h_2(k) = \begin{cases} 1 & \text{se } k < m \\ \lfloor \frac{k}{m} \rfloor & \text{se } k \geq m \end{cases}$$

End	Reg.	End	Reg.
0:	77	0:	77
1:		1:	
2:		2:	
3:		3:	
4:		4:	
5:	27	5:	27
6:		6:	16
7:	18	7:	18
8:		8:	
9:	29	9:	29
10:	43	10:	43

Hashing Duplo

Busca com saltos caso haja colisão (espalhamento das chaves)

► Ex.: 27, 18, 29, 43, 77, 16, 40, 49 e 5; $m = 11$

$$h_1(k) = k \bmod m \quad \text{e} \quad h_2(k) = \begin{cases} 1 & \text{se } k < m \\ \lfloor \frac{k}{m} \rfloor & \text{se } k \geq m \end{cases}$$

End	Reg.	End	Reg.	End	Reg.
0:	77	0:	77	0:	77
1:		1:		1:	
2:		2:		2:	40
3:		3:		3:	
4:		4:		4:	
5:	27	5:	27	5:	27
6:		6:	16	6:	16
7:	18	7:	18	7:	18
8:		8:		8:	
9:	29	9:	29	9:	29
10:	43	10:	43	10:	43

Hashing Duplo

Busca com saltos caso haja colisão (espalhamento das chaves)

► Ex.: 27, 18, 29, 43, 77, 16, 40, 49 e 5; $m = 11$

$$h_1(k) = k \bmod m \quad \text{e} \quad h_2(k) = \begin{cases} 1 & \text{se } k < m \\ \lfloor \frac{k}{m} \rfloor & \text{se } k \geq m \end{cases}$$

End	Reg.
0:	77
1:	
2:	
3:	
4:	
5:	27
6:	
7:	18
8:	
9:	29
10:	43

End	Reg.
0:	77
1:	
2:	
3:	
4:	
5:	27
6:	16
7:	18
8:	
9:	29
10:	43

End	Reg.
0:	77
1:	
2:	40
3:	
4:	
5:	27
6:	16
7:	18
8:	
9:	29
10:	43

End	Reg.
0:	77
1:	
2:	40
3:	49
4:	
5:	27
6:	16
7:	18
8:	
9:	29
10:	43

Hashing Duplo

Busca com saltos caso haja colisão (espalhamento das chaves)

► Ex.: 27, 18, 29, 43, 77, 16, 40, 49 e 5; $m = 11$

$$h_1(k) = k \bmod m \quad \text{e} \quad h_2(k) = \begin{cases} 1 & \text{se } k < m \\ \lfloor \frac{k}{m} \rfloor & \text{se } k \geq m \end{cases}$$

End	Reg.
0:	77
1:	
2:	
3:	
4:	
5:	27
6:	
7:	18
8:	
9:	29
10:	43

End	Reg.
0:	77
1:	
2:	
3:	
4:	
5:	27
6:	16
7:	18
8:	
9:	29
10:	43

End	Reg.
0:	77
1:	
2:	40
3:	
4:	
5:	27
6:	16
7:	18
8:	
9:	29
10:	43

End	Reg.
0:	77
1:	
2:	40
3:	49
4:	
5:	27
6:	16
7:	18
8:	
9:	29
10:	43

End	Reg.
0:	77
1:	
2:	40
3:	49
4:	
5:	27
6:	16
7:	18
8:	5
9:	29
10:	43

- Média de acessos ao arquivo para este conjunto de chaves:

$$\frac{1+1+2+1+1+2+3+6+4}{9} = 2,3$$

Hashing Duplo: Remoção de Registro

- ▶ A remoção é análoga à de sondagem linear: marcação de registro
- ▶ Remoção do 39:

End	Reg.
0:	77
1:	
2:	13
3:	49
4:	
5:	27
6:	39
7:	18
8:	5
9:	29
10:	43

Hashing Duplo: Remoção de Registro

- ▶ A remoção é análoga à de sondagem linear: marcação de registro
- ▶ Remoção do 39:

End	Reg.	End	Reg.
0:	77	0:	77
1:		1:	
2:	13	2:	13
3:	49	3:	49
4:		4:	
5:	27	5:	27
6:	39	6:	*
7:	18	7:	18
8:	5	8:	5
9:	29	9:	29
10:	43	10:	43

Exercício

Qual seria o estado final de um arquivo usando **sondagem linear** com $m = 11$ e $h(k) = k \bmod m$, após a inserção da seguinte sequência de chaves:

16, 19, 27, 41, 40, 5, 29, 13

End	Reg.
0:	
1:	
2:	
3:	
4:	
5:	
6:	
7:	
8:	
9:	
10:	

Exercício

Qual seria o estado final de um arquivo usando **sondagem linear** com $m = 11$ e $h(k) = k \bmod m$, após a inserção da seguinte sequência de chaves:

16, 19, 27, 41, 40, 5, 29, 13

End	Reg.	End	Reg.
0:		0:	
1:		1:	
2:		2:	
3:		3:	
4:		4:	
5:		5:	16
6:		6:	
7:		7:	
8:		8:	
9:		9:	
10:		10:	

Exercício

Qual seria o estado final de um arquivo usando **sondagem linear** com $m = 11$ e $h(k) = k \bmod m$, após a inserção da seguinte sequência de chaves:

16, 19, 27, 41, 40, 5, 29, 13

End	Reg.	End	Reg.	End	Reg.
0:		0:		0:	
1:		1:		1:	
2:		2:		2:	
3:		3:		3:	
4:		4:		4:	
5:		5:	16	5:	16
6:		6:		6:	
7:		7:		7:	
8:		8:		8:	19
9:		9:		9:	
10:		10:		10:	

Exercício

Qual seria o estado final de um arquivo usando **sondagem linear** com $m = 11$ e $h(k) = k \bmod m$, após a inserção da seguinte sequência de chaves:

16, 19, 27, 41, 40, 5, 29, 13

End	Reg.	End	Reg.	End	Reg.	End	Reg.
0:		0:		0:		0:	
1:		1:		1:		1:	
2:		2:		2:		2:	
3:		3:		3:		3:	
4:		4:		4:		4:	
5:		5:	16	5:	16	5:	16
6:		6:		6:		6:	27
7:		7:		7:		7:	
8:		8:		8:	19	8:	19
9:		9:		9:		9:	
10:		10:		10:		10:	

Exercício

Qual seria o estado final de um arquivo usando **sondagem linear** com $m = 11$ e $h(k) = k \bmod m$, após a inserção da seguinte sequência de chaves:

16, 19, 27, 41, 40, 5, 29, 13

End	Reg.	End	Reg.	End	Reg.	End	Reg.	End	Reg.
0:		0:		0:		0:		0:	
1:		1:		1:		1:		1:	
2:		2:		2:		2:		2:	
3:		3:		3:		3:		3:	
4:		4:		4:		4:		4:	
5:		5:	16	5:	16	5:	16	5:	16
6:		6:		6:		6:	27	6:	27
7:		7:		7:		7:		7:	
8:		8:		8:	19	8:	19	8:	19
9:		9:		9:		9:		9:	41
10:		10:		10:		10:		10:	

Exercício

Qual seria o estado final de um arquivo usando **sondagem linear** com $m = 11$ e $h(k) = k \bmod m$, após a inserção da seguinte sequência de chaves:

16, 19, 27, 41, 40, 5, 29, 13

End	Reg.
0:	
1:	
2:	
3:	
4:	
5:	16
6:	27
7:	40
8:	19
9:	41
10:	

Exercício

Qual seria o estado final de um arquivo usando **sondagem linear** com $m = 11$ e $h(k) = k \bmod m$, após a inserção da seguinte sequência de chaves:

16, 19, 27, 41, 40, 5, 29, 13

End	Reg.	End	Reg.
0:		0:	
1:		1:	
2:		2:	
3:		3:	
4:		4:	
5:	16	5:	16
6:	27	6:	27
7:	40	7:	40
8:	19	8:	19
9:	41	9:	41
10:		10:	5

Exercício

Qual seria o estado final de um arquivo usando **sondagem linear** com $m = 11$ e $h(k) = k \bmod m$, após a inserção da seguinte sequência de chaves:

16, 19, 27, 41, 40, 5, 29, 13

End	Reg.	End	Reg.	End	Reg.
0:		0:		0:	29
1:		1:		1:	
2:		2:		2:	
3:		3:		3:	
4:		4:		4:	
5:	16	5:	16	5:	16
6:	27	6:	27	6:	27
7:	40	7:	40	7:	40
8:	19	8:	19	8:	19
9:	41	9:	41	9:	41
10:		10:	5	10:	5

Exercício

Qual seria o estado final de um arquivo usando **sondagem linear** com $m = 11$ e $h(k) = k \bmod m$, após a inserção da seguinte sequência de chaves:

16, 19, 27, 41, 40, 5, 29, 13

End	Reg.	End	Reg.	End	Reg.	End	Reg.
0:		0:		0:	29	0:	29
1:		1:		1:		1:	
2:		2:		2:		2:	13
3:		3:		3:		3:	
4:		4:		4:		4:	
5:	16	5:	16	5:	16	5:	16
6:	27	6:	27	6:	27	6:	27
7:	40	7:	40	7:	40	7:	40
8:	19	8:	19	8:	19	8:	19
9:	41	9:	41	9:	41	9:	41
10:		10:	5	10:	5	10:	5

Exercício

Qual seria o estado final de um arquivo usando **sondagem linear** com $m = 11$ e $h(k) = k \bmod m$, após a inserção da seguinte sequência de chaves:

16, 19, 27, 41, 40, 5, 29, 13

End	Reg.	End	Reg.	End	Reg.	End	Reg.
0:		0:		0:	29	0:	29
1:		1:		1:		1:	
2:		2:		2:		2:	13
3:		3:		3:		3:	
4:		4:		4:		4:	
5:	16	5:	16	5:	16	5:	16
6:	27	6:	27	6:	27	6:	27
7:	40	7:	40	7:	40	7:	40
8:	19	8:	19	8:	19	8:	19
9:	41	9:	41	9:	41	9:	41
10:		10:	5	10:	5	10:	5

Média de acessos: $\frac{1+1+2+2+1+6+5+1}{8} = 2,38$

Exercício

Qual seria o estado final de um arquivo usando **hashing duplo** com **$m = 11$** e as funções $h_1(k)$ e $h_2(k)$ definidas anteriormente, após a inserção da seguinte sequência de chaves:

16, 19, 27, 41, 40, 5, 29, 13

End	Reg.
0:	
1:	
2:	
3:	
4:	
5:	
6:	
7:	
8:	
9:	
10:	

Exercício

Qual seria o estado final de um arquivo usando **hashing duplo** com $m = 11$ e as funções $h_1(k)$ e $h_2(k)$ definidas anteriormente, após a inserção da seguinte sequência de chaves:

16, 19, 27, 41, 40, 5, 29, 13

End	Reg.	End	Reg.
0:		0:	
1:		1:	
2:		2:	
3:		3:	
4:		4:	
5:		5:	16
6:		6:	
7:		7:	
8:		8:	
9:		9:	
10:		10:	

Exercício

Qual seria o estado final de um arquivo usando **hashing duplo** com $m = 11$ e as funções $h_1(k)$ e $h_2(k)$ definidas anteriormente, após a inserção da seguinte sequência de chaves:

16, 19, 27, 41, 40, 5, 29, 13

End	Reg.	End	Reg.	End	Reg.
0:		0:		0:	
1:		1:		1:	
2:		2:		2:	
3:		3:		3:	
4:		4:		4:	
5:		5:	16	5:	16
6:		6:		6:	
7:		7:		7:	
8:		8:		8:	19
9:		9:		9:	
10:		10:		10:	

Exercício

Qual seria o estado final de um arquivo usando **hashing duplo** com $m = 11$ e as funções $h_1(k)$ e $h_2(k)$ definidas anteriormente, após a inserção da seguinte sequência de chaves:

16, 19, 27, 41, 40, 5, 29, 13

End	Reg.	End	Reg.	End	Reg.	End	Reg.
0:		0:		0:		0:	
1:		1:		1:		1:	
2:		2:		2:		2:	
3:		3:		3:		3:	
4:		4:		4:		4:	
5:		5:	16	5:	16	5:	16
6:		6:		6:		6:	
7:		7:		7:		7:	27
8:		8:		8:	19	8:	19
9:		9:		9:		9:	
10:		10:		10:		10:	

Exercício

Qual seria o estado final de um arquivo usando **hashing duplo** com $m = 11$ e as funções $h_1(k)$ e $h_2(k)$ definidas anteriormente, após a inserção da seguinte sequência de chaves:

16, 19, 27, 41, 40, 5, 29, 13

End	Reg.	End	Reg.	End	Reg.	End	Reg.	End	Reg.
0:		0:		0:		0:		0:	41
1:		1:		1:		1:		1:	
2:		2:		2:		2:		2:	
3:		3:		3:		3:		3:	
4:		4:		4:		4:		4:	
5:		5:	16	5:	16	5:	16	5:	16
6:		6:		6:		6:		6:	
7:		7:		7:		7:	27	7:	27
8:		8:		8:	19	8:	19	8:	19
9:		9:		9:		9:		9:	
10:		10:		10:		10:		10:	

Exercício

Qual seria o estado final de um arquivo usando **hashing duplo** com $m = 11$ e as funções $h_1(k)$ e $h_2(k)$ definidas anteriormente, após a inserção da seguinte sequência de chaves:

16, 19, 27, 41, 40, 5, 29, 13

End	Reg.
0:	41
1:	
2:	
3:	
4:	
5:	16
6:	
7:	27
8:	19
9:	
10:	40

Exercício

Qual seria o estado final de um arquivo usando **hashing duplo** com $m = 11$ e as funções $h_1(k)$ e $h_2(k)$ definidas anteriormente, após a inserção da seguinte sequência de chaves:

16, 19, 27, 41, 40, 5, 29, 13

End	Reg.	End	Reg.
0:	41	0:	41
1:		1:	
2:		2:	
3:		3:	
4:		4:	
5:	16	5:	16
6:		6:	5
7:	27	7:	27
8:	19	8:	19
9:		9:	
10:	40	10:	40

Exercício

Qual seria o estado final de um arquivo usando **hashing duplo** com $m = 11$ e as funções $h_1(k)$ e $h_2(k)$ definidas anteriormente, após a inserção da seguinte sequência de chaves:

16, 19, 27, 41, 40, 5, 29, 13

End	Reg.	End	Reg.	End	Reg.
0:	41	0:	41	0:	41
1:		1:		1:	
2:		2:		2:	
3:		3:		3:	
4:		4:		4:	
5:	16	5:	16	5:	16
6:		6:	5	6:	5
7:	27	7:	27	7:	27
8:	19	8:	19	8:	19
9:		9:		9:	29
10:	40	10:	40	10:	40

Exercício

Qual seria o estado final de um arquivo usando **hashing duplo** com $m = 11$ e as funções $h_1(k)$ e $h_2(k)$ definidas anteriormente, após a inserção da seguinte sequência de chaves:

16, 19, 27, 41, 40, 5, 29, 13

End	Reg.
0:	41
1:	
2:	
3:	
4:	
5:	16
6:	
7:	27
8:	19
9:	
10:	40

End	Reg.
0:	41
1:	
2:	
3:	
4:	
5:	16
6:	5
7:	27
8:	19
9:	
10:	40

End	Reg.
0:	41
1:	
2:	
3:	
4:	
5:	16
6:	5
7:	27
8:	19
9:	29
10:	40

End	Reg.
0:	41
1:	
2:	13
3:	
4:	
5:	16
6:	5
7:	27
8:	19
9:	29
10:	40

Exercício

Qual seria o estado final de um arquivo usando **hashing duplo** com $m = 11$ e as funções $h_1(k)$ e $h_2(k)$ definidas anteriormente, após a inserção da seguinte sequência de chaves:

16, 19, 27, 41, 40, 5, 29, 13

End	Reg.	End	Reg.	End	Reg.	End	Reg.
0:	41	0:	41	0:	41	0:	41
1:		1:		1:		1:	
2:		2:		2:		2:	13
3:		3:		3:		3:	
4:		4:		4:		4:	
5:	16	5:	16	5:	16	5:	16
6:		6:	5	6:	5	6:	5
7:	27	7:	27	7:	27	7:	27
8:	19	8:	19	8:	19	8:	19
9:		9:		9:	29	9:	29
10:	40	10:	40	10:	40	10:	40

Média de acessos: $\frac{1+1+2+2+2+2+2+1}{8} = 1,63$

Análise de Endereçamento Aberto

Referência: [Cormen et al.]

- ▶ **Teorema 11.6:** Dada uma tabela hash com fator de carga $\alpha = n/m < 1$, o número esperado de acessos à tabela em uma busca sem sucesso é, no máximo, $1/(1 - \alpha)$, assumindo-se hashing uniforme.
- ▶ **Corolário 11.7:** Inserir um elemento em uma tabela hash com endereçamento aberto com fator de carga α requer no máximo $1/(1 - \alpha)$ acessos em média, assumindo-se hashing uniforme.
- ▶ **Teorema 11.8:** Dada uma tabela hash com endereçamento aberto com fator de carga $\alpha < 1$, o número esperado de acessos à tabela em uma consulta com sucesso é, no máximo, $\frac{1}{\alpha} \ln \frac{1}{1-\alpha}$ assumindo-se hashing uniforme e que todas as chaves da tabela possuem mesma probabilidade de serem consultadas.

Se a tabela estiver 90% cheia, o número esperado de acessos é menor que 2,559.