

Algoritmos e Estruturas de Dados III

Aula 8.3 – Métodos de Dicionário
Conceitos, LZ77 e LZ78

2024



PUC Minas

Métodos de Dicionário



Métodos de Dicionário

- Os símbolos (ou conjunto de símbolos) são substituídos por códigos a partir de um “dicionário”
- Os códigos possuem tamanho fixo
- Os dicionários podem ser estáticos ou dinâmicos
- Ex: LZ77 / LZ78 / LZW

Métodos de Dicionário

Exemplo Simples (Dicionário Estático)

Codificação: considere que a palavra “algoritmo” tem o código 1025 no dicionário. Sua codificação será

00000000010000000001

⇒ Redução de 72 (9 bytes) para 20 bits

considere que a palavra “abc” **não** está no dicionário.

Ela não será codificada, e será necessário armazenar, além do bit extra, o número de bytes não codificados:

100000011011000010110001001100011

tam = 3 | a = 97 | b = 98 | c = 99

⇒ Aumento de 24 (3 bytes) para 33 bits

Métodos de Dicionário

Exemplo Simples (Dicionário Estático)

- um dicionário que possui um código para as palavras da língua portuguesa. Para 500.000 palavras, seriam necessários 19 bits para cada código ($2^{19} = 524.288$)
- o algoritmo irá substituir cada palavra do arquivo pelo seu código. Caso a palavra não existir no dicionário, ela é colocada sem substituição no arquivo de saída
- o arquivo compactado irá conter códigos e palavras

Possível problema: como saber diferenciar entre os códigos e as palavras (todos são conjuntos de bits)

Solução: usar um bit extra para informar se é um código ou não

Métodos de Dicionário

Exemplo Simples (Dicionário Estático)

No caso de muitas falhas (palavras não encontradas no dicionário), o tamanho do arquivo irá aumentar !!!

Perguntas:

- o que aconteceria se o texto estivesse em inglês?

Esse método de dicionário estático não é bom para compactadores de uso geral mas pode adequado para compactadores específicos

Métodos de Dicionário

Codificação de dígrafos (Dicionário Estático)

A Codificação de Dígrafos é uma técnica de compressão que utiliza um dicionário estático (fixo) para substituir caracteres individuais e pares de caracteres (dígrafos) pelos seus códigos.

Conteúdo:

- Todos caracteres do alfabeto fonte.
- Dígrafos mais frequentes.

Métodos de Dicionário

Codificação de digráfos (Dicionário Estático)

Example 5.3.1:

Suppose we have a source with a five-letter alphabet $\mathcal{A} = \{a, b, c, d, r\}$. Based on knowledge about the source, we build the dictionary shown in Table 5.1.

TABLE 5.1 A sample dictionary.

Code	Entry	Code	Entry
000	<i>a</i>	100	<i>r</i>
001	<i>b</i>	101	<i>ab</i>
010	<i>c</i>	110	<i>ac</i>
011	<i>d</i>	111	<i>ad</i>

Suppose we wish to encode the sequence

abracadabra

Métodos de Dicionário

TABLE 5.3 Thirty most frequently occurring pairs of characters in a collection of C programs containing 64,983 characters.

Pair	Count	Pair	Count
<i>bb</i>	5728	<i>st</i>	442
<i>nlb</i>	1471	<i>le</i>	440
<i>;nl</i>	1133	<i>ut</i>	440
<i>in</i>	985	<i>f(</i>	416
<i>nt</i>	739	<i>ar</i>	381
<i>=b</i>	687	<i>or</i>	374
<i>bi</i>	662	<i>rb</i>	373
<i>tb</i>	615	<i>en</i>	371
<i>b=</i>	612	<i>er</i>	358
<i>);</i>	558	<i>ri</i>	357
<i>,b</i>	554	<i>at</i>	352
<i>nl nl</i>	506	<i>pr</i>	351
<i>bf</i>	505	<i>te</i>	349
<i>eb</i>	500	<i>an</i>	348
<i>b*</i>	444	<i>lo</i>	347

Métodos de Dicionário

- Na verdade, os algoritmos de uso geral utilizam dicionários dinâmicos (métodos adaptativos)
- Os dicionários são construídos à medida que o texto vai sendo lido. Se uma palavra não existe no dicionário ela pode ser inserida, e palavras “fora de uso” podem ser removidas do dicionário
- Na década de 70, Abraham Lempel e Jacob Ziv criaram a base para os algoritmos de compactação baseados em dicionários dinâmicos
- A grande maioria dos métodos de dicionários existentes hoje são extensões ou modificações dos algoritmos de Lempel-Ziv (LZ)
- Boa parte dos programas comerciais (zip, compress, formato gif, etc.) se baseiam em variações/combinações destes algoritmos

LZ77

LZ77

Parte do conteúdo já lido é usado como dicionário
Utiliza o conceito de “janela deslizante” (sliding window)

Arquivo

. . . A B A B A B C A C A C B A A A A B C . . .

LZ77

Parte do conteúdo já lido é usado como dicionário
Utiliza o conceito de “janela deslizante” (sliding window)

Arquivo

... A B A B A B C A C A C B A A A A B C ...

Janela Deslizante



LZ77

Parte do conteúdo já lido é usado como dicionário
Utiliza o conceito de “janela deslizante” (sliding window)

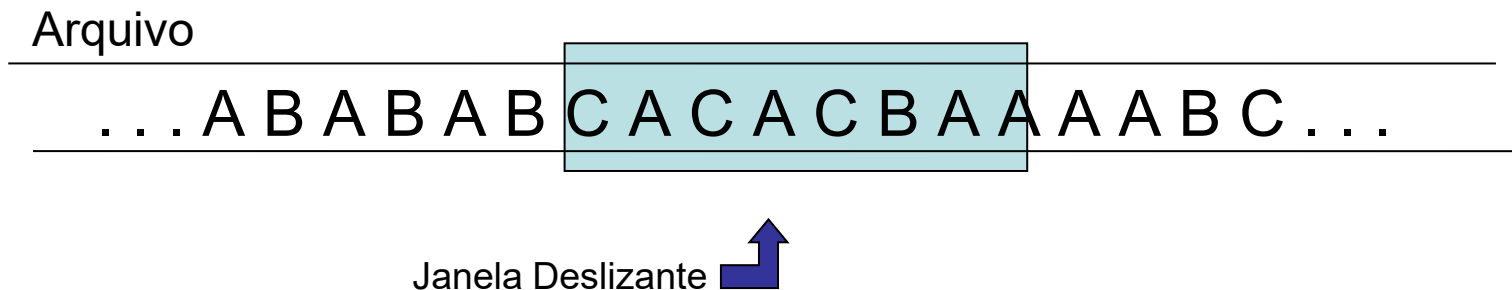
Arquivo

... A B A B A B C A C A C B A A A A B C ...

Janela Deslizante 

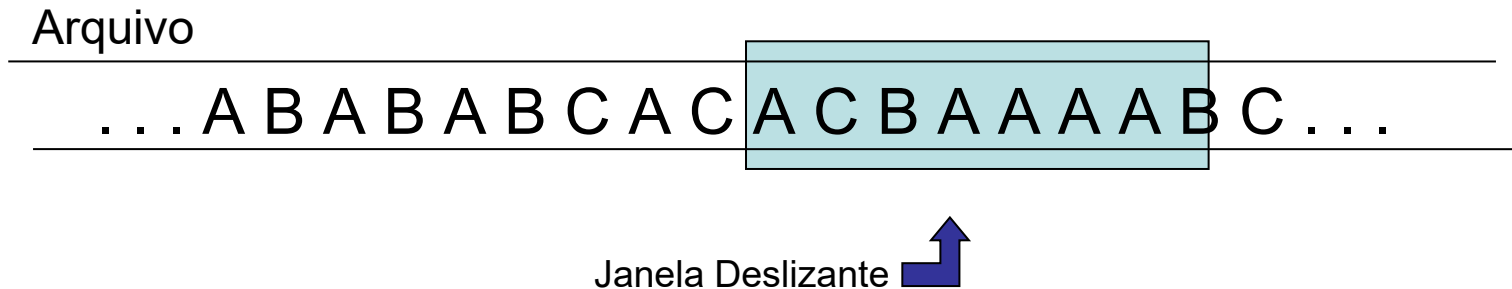
LZ77

Parte do conteúdo já lido é usado como dicionário
Utiliza o conceito de “janela deslizante” (sliding window)



LZ77

Parte do conteúdo já lido é usado como dicionário
Utiliza o conceito de “janela deslizante” (sliding window)



LZ77

A janela deslizante é dividida em duas partes

- 1ª Parte: símbolos previamente lidos (dicionário ou *search buffer*)
- 2ª Parte: símbolos que serão processados (*look-ahead buffer*)

Arquivo

. . . A B A B A B C A C A C B A A A A B C . . .

Dicionário



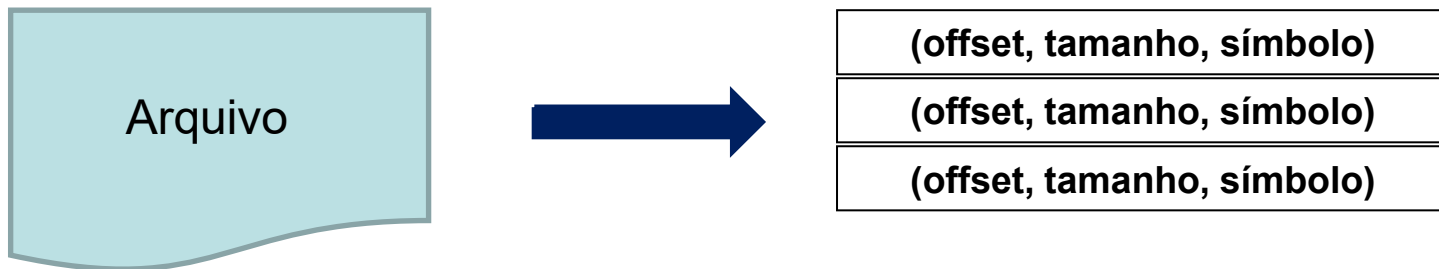
Look-ahead buffer

LZ77 - Compactação



Métodos de Dicionário

- Compactação
- Para cada conjunto de símbolos do buffer, o algoritmo procura pelo maior casamento possível no dicionário
- Se for encontrado, gera como saída um token contendo a posição no dicionário (offset), o tamanho da string e o próximo símbolo do buffer.
- Se não for encontrado, escreve 0 para o offset, 0 para o tamanho e o símbolo não encontrado



Métodos de Dicionário

	sir_sid_eastman_	⇒	(0,0,"s")
	sir_sid_eastman_e	⇒	(0,0,"i")
	sir_sid_eastman_ea	⇒	(0,0,"r")
	sir_sid_eastman_eas	⇒	(0,0,"_")
	sir_sid_eastman_easi	⇒	(4,2,"d")

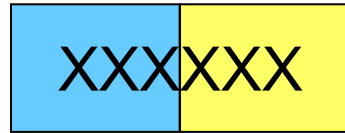
The next step matches the space and encodes the string “_e”.

sir_sid_eastman_easily_	⇒	(4,1,"e")
sir_sid_eastman_easily_te	⇒	(0,0,"a")

LZ77

Exemplo de compactação

Tamanho da Janela = 6 (Dicionário = 3 / *Look-ahead buffer* = 3)



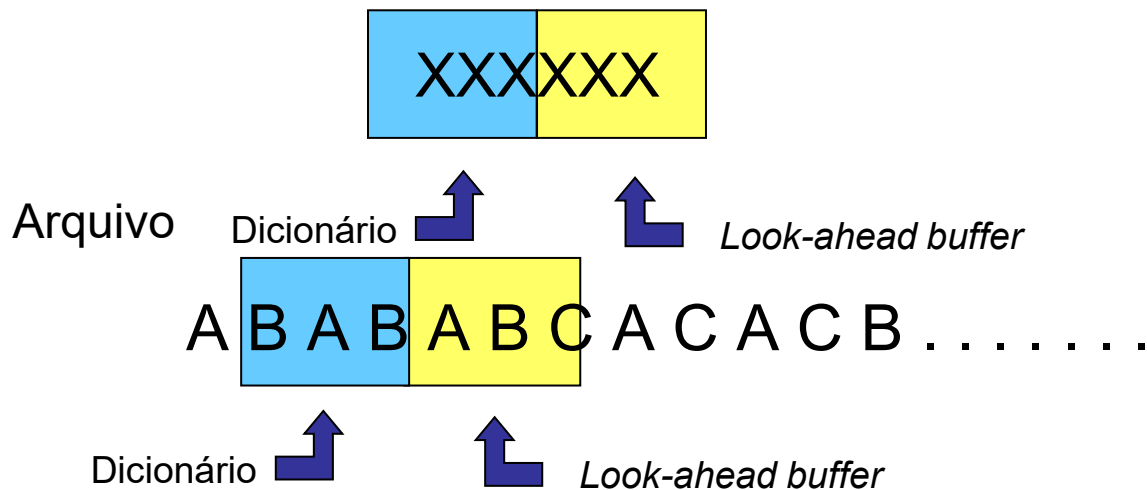
Arquivo Dicionário   *Look-ahead buffer*

A B A B A B C A C A C B

LZ77

Exemplo de compactação

Tamanho da Janela = 6 (Dicionário = 3 / *Look-ahead buffer* = 3)



LZ77

- Exemplo de Compactação

Arquivo



LZ77

- Exemplo de Compactação

Arquivo



?



(0, 0, A)

Não usa nenhum símbolo do dicionário
e escreve o próximo símbolo

LZ77

- Exemplo de Compactação

Arquivo



(0, 0, A)

Avança a janela 1 posição

LZ77

- Exemplo de Compactação

Arquivo

A B A B A B C A C A C B

(0, 0, A)

LZ77

- Exemplo de Compactação

Arquivo

The diagram shows a sequence of characters: A B A B A B C A C A C B The first 'A' is enclosed in a light blue box, and the next 'B' is enclosed in a light yellow box. A green curved arrow points from the 'A' in the blue box to the 'A' in the yellow box. Below the yellow box is a question mark. Below the question mark is the tuple (0, 0, A).

?

(0, 0, A)

LZ77

- Exemplo de Compactação

Arquivo

AB A B A B C A C A C B



?



(0, 0, A)

(0, 0, B)

Não usa nenhum símbolo do dicionário
e escreve o próximo símbolo

LZ77

- Exemplo de Compactação

Arquivo



(0, 0, A)

(0, 0, B)

Avança a janela 1 posição

LZ77

- Exemplo de Compactação

Arquivo

A B A B A B C A C A C B



?

(0, 0, A)

(0, 0, B)

LZ77

- Exemplo de Compactação

Arquivo



(0, 0, A)

(0, 0, B)

LZ77

- Exemplo de Compactação

Arquivo

AB**AB**A B C A C A C B



(0, 0, A)

(0, 0, B)

(2, 2, A)

Volta 2 posições do dicionário, usa 2 símbolos e escreve o próximo símbolo do look ahead

LZ77

- Exemplo de Compactação

Arquivo



(0, 0, A)

(0, 0, B)

(2, **2**, A)

Avança a janela 1+(X) posições
Onde X é o tamanho gravado na "trinca" anterior
Aqui, X = 2

LZ77

- Exemplo de Compactação

Arquivo

A B A B A B C A C A C B

(0, 0, A)

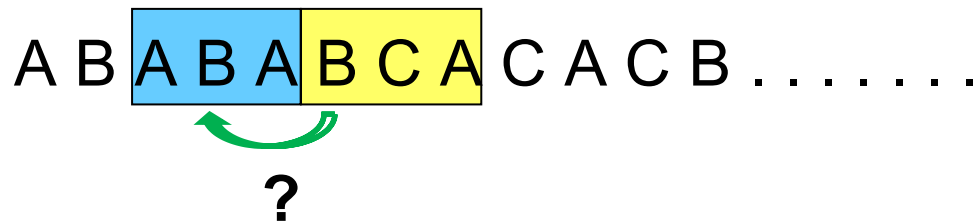
(0, 0, B)

(2, 2, A)

LZ77

- Exemplo de Compactação

Arquivo



(0, 0, A)

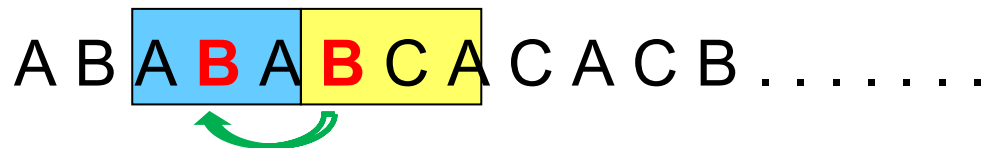
(0, 0, B)

(2, 2, A)

LZ77

- Exemplo de Compactação

Arquivo



(0, 0, A)

(0, 0, B)

(2, 2, A)

LZ77

- Exemplo de Compactação

Arquivo

A B A B A B C A C A C B



(0, 0, A)

(0, 0, B)

(2, 2, A)

(2, 1, C)

Volta 2 posições do
dicionário,

usa 1 símbolo

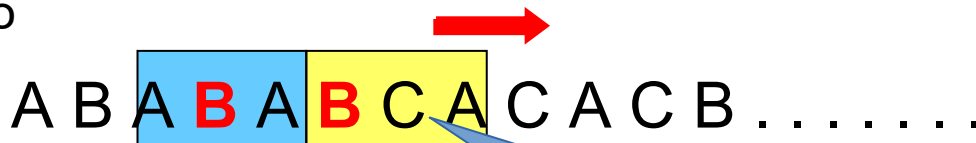
e escreve o próximo símbolo
do look ahead

LZ77

- Exemplo de Compactação

Arquivo

A B A **B** A **B** C A C A C B



The diagram shows a sequence of characters: A, B, A, B, A, B, C, A, C, A, C, B, followed by an ellipsis. A red arrow points from the first 'B' to the second 'B'. A blue callout box points to the second 'B' and contains text explaining the window advancement rule.

(0, 0, A)

(0, 0, B)

(2, 2, A)

(2, **1**, C)

Avança a janela 1+(X) posições
Onde X é o tamanho gravado na “trinca” anterior
Aqui, X = 1

- Exemplo de Compactação

Arquivo

A B A B A B C A C A C B

(0, 0, A)

(0, 0, B)

(2, 2, A)

(2, 1, C)

- Exemplo de Compactação

Arquivo

A B A B A B C A C A C B



?

(0, 0, A)

(0, 0, B)


(2, 2, A)

(2, 1, C)

- Exemplo de Compactação

Arquivo

A B A B A B C A C A C B



(0, 0, A)

(0, 0, B)

(2, 2, A)

(2, 1, C)

LZ77

- Exemplo de Compactação

Arquivo

A B A B **A B C** **A C A** C B



(0, 0, A)

(0, 0, B)

(2, 2, A)

(2, 1, C)

(3, 1, C)

Volta 3 posições do
dicionário,

usa 1 símbolo e escreve

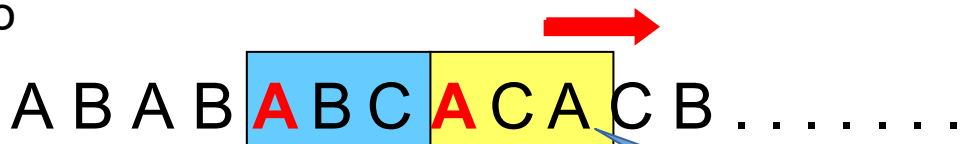
o próximo símbolo do look
ahead

LZ77

- Exemplo de Compactação

Arquivo

A B A B A B C A C A C B



(0, 0, A)

(0, 0, B)

(2, 2, A)

(2, 1, C)

(3, 1, C)

Avança a janela 1+(X) posições
Onde X é o tamanho gravado na
“trinca” anterior
Aqui, X = 1

- Exemplo de Compactação

Arquivo

A B A B A B C A C A C B

(0, 0, A)

(0, 0, B)

(2, 2, A)

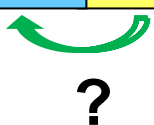
(2, 1, C)

(3, 1, C)

- Exemplo de Compactação

Arquivo

A B A B A B C A C A C B



(0, 0, A)

(0, 0, B)

(2, 2, A)

(2, 1, C)


(3, 1, C)

LZ77

- Exemplo de Compactação

Arquivo

A B A B A B C A C A C B



(0, 0, A)

(0, 0, B)

(2, 2, A)

(2, 1, C)

(3, 1, C)

LZ77

- Exemplo de Compactação

Arquivo

A B A B A B C A C A C B



(0, 0, A)

(0, 0, B)

(2, 2, A)

(2, 1, C)

(3, 1, C)

(2, 2, B)

Volta 2 posições do
dicionário,

usa 2 símbolos e escreve

o próximo símbolo do look
ahead

LZ77

- Exemplo de Compactação

Arquivo original

A B A B A B C A C A C B



Arquivo compactado

0 0 A 0 0 B 2 2 A 2 1 C 3 1 C 2 2 B

ou

0 0 1 0 0 2 2 2 1 2 1 3 3 1 3 2 2 2

Obs: A \equiv 1 B \equiv 2 C \equiv 3

LZ77

a window buffer.

...Mr. alf eastman easily grows alf alfa in his garden...

(28,3,"a") ou (3, 4, " ") ?

LZ77 - Descompactação



LZ77

- Exemplo de descompactação

Arquivo compactado

0 0 A 0 0 B 2 2 A 2 1 C 3 1 C 2 2 B

LZ77

- Exemplo de descompactação

Arquivo compactado

0 0 A 0 0 B 2 2 A 2 1 C 3 1 C 2 2 B

A

LZ77

- Exemplo de descompactação

Arquivo compactado

0 0 A 0 0 B 2 2 A 2 1 C 3 1 C 2 2 B

A B

LZ77

- Exemplo de descompactação

Arquivo compactado

0 0 A 0 0 B 2 2 A 2 1 C 3 1 C 2 2 B

A B A B A

LZ77

- Exemplo de descompactação

Arquivo compactado

0 0 A 0 0 B 2 2 A 2 1 C 3 1 C 2 2 B

A B A B A B C

LZ77

- Exemplo de descompactação

Arquivo compactado

0 0 A 0 0 B 2 2 A 2 1 C 3 1 C 2 2 B

A B A B A B C A C

LZ77

- Exemplo de descompactação

Arquivo compactado

0 0 A 0 0 B 2 2 A 2 1 C 3 1 C 2 2 B

A B A B A B C A C A C B

LZ77

- Exemplo de descompactação

Arquivo compactado

0 0 A 0 0 B 2 2 A 2 1 C 3 1 C 2 2 B

Arquivo original:

A B A B A B C A C A C B

LZ77

- Descompactação
 - Basta manter um buffer do tamanho do dicionário
 - Para cada **token** do arquivo verifica-se qual a seqüência de símbolos correspondente no buffer escrevendo como saída essa seqüência e o símbolo armazenado no **token**, deslocando o buffer para inserir esses novos símbolos



LZ78



LZ78

- Cria-se um dicionário de prefixos (ou sequências) já encontrados no conteúdo
- As sequências são formadas por símbolos que podem representar bytes, caracteres, cores, etc. (isto é, qualquer unidade básica de informação)
- Não utiliza o conceito de “janela deslizante”
- O dicionário é construído dinamicamente a partir das sequências encontradas e é armazenado em uma tabela na memória primária
- Ao final do processo de compactação (ou de descompactação) o dicionário é descartado

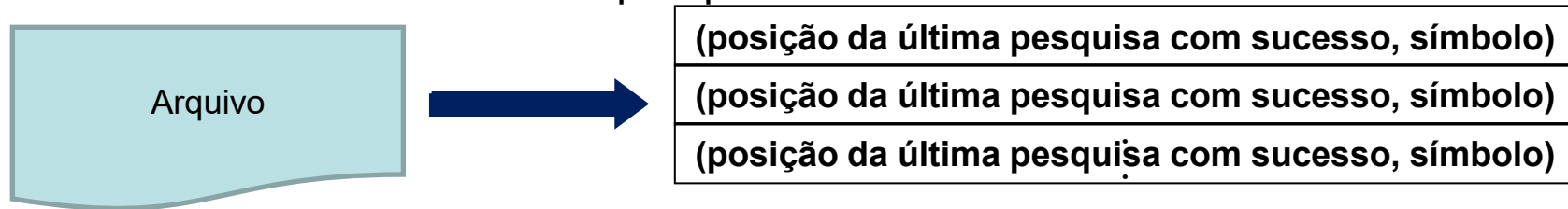
LZ78

- Exemplo de Dicionário
 - Exemplo usando sequências de caracteres

Posição	Prefixo
1	A
2	B
3	AB
4	C
5	BC
6	ABC

LZ78

- Informações importantes para execução
 - Símbolo lido: byte, caractere, etc
 - Prefixo (ou sequência) atual
 - Inicialmente o prefixo é nulo
 - Posição da tabela da última pesquisa com sucesso
 - Usar o valor ZERO para representar o caso em que ainda não se realizou nenhuma pesquisa na tabela



LZ78

- Compactação (Repetir até o final do arquivo)
 - Ler o próximo símbolo do arquivo e concatenar o símbolo do final da sequência (ou prefixo)
 - Fazer uma busca na tabela para encontrar o prefixo
 - Se o prefixo for encontrado, atualizar o indicador de posição da última pesquisa com sucesso na tabela
 - Se o prefixo não for encontrado,
 - Gerar um token contendo a posição da última pesquisa com sucesso e o último símbolo lido (responsável pela falha na busca)
 - Inserir o prefixo na tabela para evitar novas falhas
 - Reinicializar o prefixo (com o valor NULO) e o indicador de posição de pesquisa com sucesso (com o valor ZERO)

LZ78

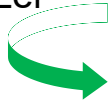
A B A B C A B C A B C A A B C A B

- Símbolo:
- Prefixo :

Posição	Prefixo
1	
2	
3	
4	
5	

LZ78

Ler



A B A B C A B C A B C A A B C A B

- Símbolo: A
- Prefixo :

Posição	Prefixo
1	
2	
3	
4	
5	

LZ78

A B A B C A B C A B C A A B C A B

Anexar



- Símbolo: A
- Prefixo : A

Posição	Prefixo
1	
2	
3	
4	
5	

LZ78

A B A B C A B C A B C A A B C A B

- Símbolo: A
- Prefixo : A

Buscar



Posição	Prefixo
1	
2	
3	
4	
5	

LZ78

A B A B C A B C A B C A A B C A B

- Símbolo: A
- Prefixo : A

Buscar



. Posição	. Prefixo
. 1	
. 2	
. 3	
. 4	
. 5	

FALHA
!!!

LZ78

A B A B C A B C A B C A A B C A B

- Símbolo: A
- Prefixo : A

. Posição	. Prefixo
. 1	
. 2	
. 3	
. 4	
. 5	

Gerar
token

(0, A)

Dicionário

LZ78

A B A B C A B C A B C A A B C A B

- Símbolo: A
- Prefixo : A

(0, A)

. Posição	. Prefixo
. 1	A
. 2	
. 3	
. 4	
. 5	

Inserir
prefixo



LZ78

A B A B C A B C A B C A A B C A B

- Símbolo:
- Prefixo :  Reinicializar

(0, A)

Posição	Prefixo
1	A
2	
3	
4	
5	

LZ78

A B A B C A B C A B C A A B C A B

Ler



- Símbolo: B
- Prefixo :

(0, A)

Posição	Prefixo
1	A
2	
3	
4	
5	

LZ78

A B A B C A B C A B C A A B C A B

Anexar



- Símbolo: B
- Prefixo : B

(0, A)

Posição	Prefixo
1	A
2	
3	
4	
5	


LZ78

A B A B C A B C A B C A A B C A B

- Símbolo: B
- Prefixo : B

(0, A)

Buscar



Posição	Prefixo
1	A
2	
3	
4	
5	


LZ78


A B A B C A B C A B C A A B C A B

- Símbolo: B
- Prefixo : B

(0, A)

Buscar



Posição	Prefixo
1	A
2	
3	
4	
5	
6	

FALHA
!!!

LZ78

• **A B** A B C A B C A B C A A B C A B

- Símbolo: B
- Prefixo : B

Posição	Prefixo
1	A
2	
3	
4	
5	

Gerar
token

(0, A)
(0, B)

LZ78

A B A B C A B C A B C A A B C A B

- Símbolo: B
- Prefixo : B

(0, A)

(0, B)

Posição	Prefixo
1	A
2	B
3	
4	
5	

Inserir
prefixo



LZ78

A B A B C A B C A B C A A B C A B

- Símbolo:

- Prefixo :  Reinicializar

(0, A)

(0, B)

Posição	Prefixo
1	A
2	B
3	
4	
5	

LZ78

A B A B C A B C A B C A A B C A B

Ler



- Símbolo: A
- Prefixo :

(0, A)

(0, B)

Posição	Prefixo
1	A
2	B
3	
4	
5	

LZ78

A B A B C A B C A B C A A B C A B

Anexar



- Símbolo: A
- Prefixo : A

(0, A)

(0, B)

Posição	Prefixo
1	A
2	B
3	
4	
5	

LZ78

A B A B C A B C A B C A A B C A B

- Símbolo: A
- Prefixo : A

Buscar



Posição	Prefixo
1	A
2	B
3	
4	
5	

(0, A)
(0, B)

LZ78

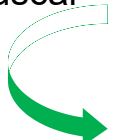
A B A B C A B C A B C A A B C A B

- Símbolo: A
- Prefixo : A

Buscar

(0, A)

(0, B)



Posição	Prefixo
1	A
2	B
3	
4	
5	

Sucesso

LZ78

A B A B C A B C A B C A A B C A B

- Símbolo: A
- Prefixo : A

(0, A)

(0, B)

Última Pos.
de Sucesso

Posição	Prefixo
1	A
2	B
3	
4	
5	

LZ78

A B A B C A B C A B C A A B C A B

Ler



- Símbolo: B
- Prefixo : A

(0, A)

(0, B)

Última Pos.
de Sucesso

Posição	Prefixo
1	A
2	B
3	
4	
5	

LZ78

A B A B C A B C A B C A A B C A B

Anexar



- Símbolo: B
- Prefixo : AB

(0, A)

(0, B)

Última Pos.
de Sucesso

Posição	Prefixo
1	A
2	B
3	
4	
5	


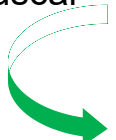
LZ78

A B A B C A B C A B C A A B C A B

- Símbolo: B
- Prefixo : AB

Buscar

Última Pos.
de Sucesso



. Posição	. Prefixo
. 1	A
. 2	B
. 3	
. 4	
. 5	

(0, A)

(0, B)

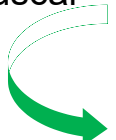
LZ78

A B A B C A B C A B C A A B C A B


- Símbolo: B
- Prefixo : AB

Buscar

Última Pos.
de Sucesso



. Posição	. Prefixo
. 1	A
. 2	B
. 3	
. 4	
. 5	



(0, A)

(0, B)

FALHA
!!!

LZ78

A B A B C A B C A B C A A B C A B

- Símbolo: B
- Prefixo : AB

Última Pos. de Sucesso	• Posição	• Prefixo
	• 1	A
	• 2	B
	• 3	
	• 4	
	• 5	

Gerar
token

(0, A)
(0, B)
(1, B)

LZ78

A B A B C A B C A B C A A B C A B

- Símbolo: B
- Prefixo : AB

(0, A)
(0, B)
(1, B)

Última Pos.
de Sucesso

. Posição	. Prefixo
. 1	A
. 2	B
. 3	AB
. 4	
. 5	

Inserir
prefixo



LZ78

A B A B C A B C A B C A A B C A B

- Símbolo:
- Prefixo :  Reinicializar

(0, A)
(0, B)
(1, B)

. Posição	. Prefixo
. 1	A
. 2	B
. 3	AB
. 4	
. 5	

LZ78

A B A B C A B C A B C A A B C A B

Ler



- Símbolo: C
- Prefixo :

(0, A)
(0, B)
(1, B)

. Posição	. Prefixo
. 1	A
. 2	B
. 3	AB
. 4	
. 5	

LZ78

A B A B C A B C A B C A A B C A B

Anexar



- Símbolo: C
- Prefixo : C

(0, A)
(0, B)
(1, B)

. Posição	. Prefixo
. 1	A
. 2	B
. 3	AB
. 4	
. 5	

LZ78

A B A B C A B C A B C A A B C A B

- Símbolo: C
- Prefixo : C

Buscar



. Posição	. Prefixo
. 1	A
. 2	B
. 3	AB
. 4	
. 5	

(0, A)
(0, B)
(1, B)

LZ78

A B A B C A B C A B C A A B C A B

- Símbolo: C
- Prefixo : C

Buscar



. Posição	. Prefixo
. 1	A
. 2	B
. 3	AB
. 4	
. 5	

(0, A)
(0, B)
(1, B)

FALHA
!!!

LZ78

A B A B C A B C A B C A A B C A B

- Símbolo: C
- Prefixo : C

. Posição	. Prefixo
. 1	A
. 2	B
. 3	AB
. 4	
. 5	

Gerar
token



(0, A)
(0, B)
(1, B)
(0, C)

LZ78

A B A B C A B C A B C A A B C A B

- Símbolo: C
- Prefixo : C

. Posição	. Prefixo
. 1	A
. 2	B
. 3	AB
. 4	C
. 5	
. 6	

(0, A)
(0, B)
(1, B)
(0, C)

Inserir
prefixo



LZ78

A B A B C A B C A B C A A B C A B

- Símbolo:  Reinicializar
- Prefixo :

. Posição	. Prefixo
. 1	A
. 2	B
. 3	AB
. 4	C
. 5	
. 6	

(0, A)
(0, B)
(1, B)
(0, C)

LZ78

A B A B C A B C A B C A A B C A B

Ler



- Símbolo: A
- Prefixo :

. Posição	. Prefixo
. 1	A
. 2	B
. 3	AB
. 4	C
. 5	
. 6	

(0, A)
(0, B)
(1, B)
(0, C)

LZ78

A B A B C A B C A B C A A B C A B

Anexar



- Símbolo: A
- Prefixo : A

. Posição	. Prefixo
. 1	A
. 2	B
. 3	AB
. 4	C
. 5	
. 6	

(0, A)
(0, B)
(1, B)
(0, C)

LZ78

A B A B C A B C A B C A A B C A B

- Símbolo: A
- Prefixo : A

Buscar



. Posição	. Prefixo
. 1	A
. 2	B
. 3	AB
. 4	C
. 5	
. 6	


(0, A)
(0, B)
(1, B)
(0, C)

LZ78

A B A B C A B C A B C A A B C A B

- Símbolo: A
- Prefixo : A

Buscar



Posição	Prefixo
1	A
2	B
3	AB
4	C
5	
6	

Sucesso

(0, A)
(0, B)
(1, B)
(0, C)

LZ78

A B A B C A B C A B C A A B C A B

- Símbolo: A
- Prefixo : A

(0, A)
(0, B)
(1, B)
(0, C)

	Posição	Prefixo
Última Pos. de Sucesso →	1	A
	2	B
	3	AB
	4	C
	5	
	6	

LZ78

A B A B C A B C A B C A A B C A B

Ler



- Símbolo: B
- Prefixo : A

(0, A)
(0, B)
(1, B)
(0, C)

Última Pos.
de Sucesso

Posição	Prefixo
1	A
2	B
3	AB
4	C
5	
6	

LZ78

A B A B C A B C A B C A A B C A B

Anexar



- Símbolo: B
- Prefixo : AB

(0, A)
(0, B)
(1, B)
(0, C)

Última Pos.
de Sucesso

Posição	Prefixo
1	A
2	B
3	AB
4	C
5	
6	

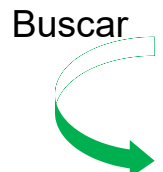
LZ78

A B A B C A B C A B C A A B C A B

- Símbolo: B
- Prefixo : AB

Buscar

Última Pos.
de Sucesso



Posição	Prefixo
1	A
2	B
3	AB
4	C
5	
6	

(0, A)
(0, B)
(1, B)
(0, C)

LZ78

A B A B C A B C A B C A A B C A B

- Símbolo: B
- Prefixo : AB

Buscar

Última Pos.
de Sucesso

Posição	Prefixo
1	A
2	B
3	AB
4	C
5	
6	

(0, A)
(0, B)
(1, B)
(0, C)

Sucesso

LZ78

A B A B C A B C A B C A A B C A B

- Símbolo: B
- Prefixo : AB

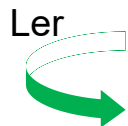
(0, A)
(0, B)
(1, B)
(0, C)

Posição	Prefixo
1	A
2	B
3	AB
4	C
5	
6	

Última Pos.
de Sucesso →

LZ78

A B A B C A B C A B C A A B C A B



- Símbolo: C
- Prefixo : AB

(0, A)
(0, B)
(1, B)
(0, C)

Posição	Prefixo
1	A
2	B
3	AB
4	C
5	
6	

Última Pos.
de Sucesso →

LZ78

A B A B C A B C A B C A A B C A B

Anexar

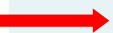


- Símbolo: C
- Prefixo : ABC

(0, A)
(0, B)
(1, B)
(0, C)

Posição	Prefixo
1	A
2	B
3	AB
4	C
5	
6	

Última Pos.
de Sucesso



LZ78

A B A B C A B C A B C A A B C A B

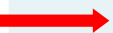
- Símbolo: C
- Prefixo : ABC

Buscar



Posição	Prefixo
1	A
2	B
3	AB
4	C
5	
6	

Última Pos.
de Sucesso



(0, A)
(0, B)
(1, B)
(0, C)

LZ78

A B A B C A B C A B C A A B C A B

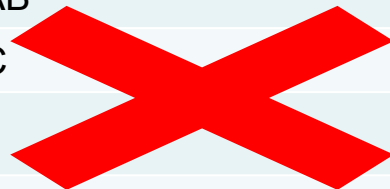
- Símbolo: C
- Prefixo : ABC

Buscar



Posição	Prefixo
1	A
2	B
3	AB
4	C
5	
6	

Última Pos.
de Sucesso



FALHA
!!!

(0, A)
(0, B)
(1, B)
(0, C)

LZ78

A B A B C A B C A B C A A B C A B

- Símbolo: C
- Prefixo : ABC

Posição	Prefixo
1	A
2	B
3	AB
4	C
5	
6	

Última Pos.
de Sucesso



Gerar
token



(0, A)
(0, B)
(1, B)
(0, C)
(3, C)

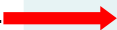
LZ78

A B A B C A B C A B C A A B C A B

- Símbolo: C
- Prefixo : ABC

Posição	Prefixo
1	A
2	B
3	AB
4	C
5	ABC
6	

Última Pos.
de Sucesso



Inserir
prefixo



(0, A)
(0, B)
(1, B)
(0, C)
(3, C)

LZ78

A B A B C A B C A B C A A B C A B

- Símbolo:
- Prefixo :  Reinicializar

Posição	Prefixo
1	A
2	B
3	AB
4	C
5	ABC
6	

(0, A)
(0, B)
(1, B)
(0, C)
(3, C)

LZ78

A B A B C A B C A B C A A B C A B

Ler



- Símbolo: A
- Prefixo :

Posição	Prefixo
1	A
2	B
3	AB
4	C
5	ABC
6	

(0, A)
(0, B)
(1, B)
(0, C)
(3, C)

LZ78

A B A B C A B C A B C A A B C A B

Anexar



- Símbolo: A
- Prefixo : A

Posição	Prefixo
1	A
2	B
3	AB
4	C
5	ABC
6	

(0, A)
(0, B)
(1, B)
(0, C)
(3, C)

LZ78

A B A B C A B C A B C A A B C A B

- Símbolo: A
- Prefixo : A

Buscar



Posição	Prefixo
1	A
2	B
3	AB
4	C
5	ABC
6	


(0, A)
(0, B)
(1, B)
(0, C)
(3, C)

LZ78

A B A B C A B C A B C A A B C A B

- Símbolo: A
- Prefixo : A

Buscar



Posição	Prefixo
1	A
2	B
3	AB
4	C
5	ABC
6	

Sucesso

(0, A)
(0, B)
(1, B)
(0, C)
(3, C)

LZ78

A B A B C A B C A B C A A B C A B

- Símbolo: A
- Prefixo : A

Última Pos.
de Sucesso

. Posição	. Prefixo
. 1	A
. 2	B
. 3	AB
. 4	C
. 5	ABC
. 6	

(0, A)
(0, B)
(1, B)
(0, C)
(3, C)

LZ78

A B A B C A B C A B C A A B C A B

Ler



- Símbolo: B
- Prefixo : A

Última Pos.
de Sucesso

. Posição	. Prefixo
. 1	A
. 2	B
. 3	AB
. 4	C
. 5	ABC
. 6	

(0, A)
(0, B)
(1, B)
(0, C)
(3, C)

LZ78

A B A B C A B C A B C A A B C A B

Anexar



- Símbolo: B
- Prefixo : AB

Última Pos.
de Sucesso

. Posição	. Prefixo
. 1	A
. 2	B
. 3	AB
. 4	C
. 5	ABC
. 6	

(0, A)
(0, B)
(1, B)
(0, C)
(3, C)


LZ78

A B A B C A B C A B C A A B C A B

- Símbolo: B
- Prefixo : AB

Buscar

Última Pos.
de Sucesso



. Posição	. Prefixo
. 1	A
. 2	B
. 3	AB
. 4	C
. 5	ABC
. 6	

(0, A)
(0, B)
(1, B)
(0, C)
(3, C)

LZ78

A B A B C A B C A B C A A B C A B

- Símbolo: B
- Prefixo : AB

Buscar

Última Pos.
de Sucesso

. Posição	. Prefixo
. 1	A
. 2	B
. 3	AB
. 4	C
. 5	ABC
. 6	

(0, A)
(0, B)
(1, B)
(0, C)
(3, C)

Sucesso

LZ78

A B A B C A B C A B C A A B C A B

- Símbolo: B
- Prefixo : AB

. Posição	. Prefixo
. 1	A
. 2	B
. 3	AB
. 4	C
. 5	ABC
. 6	

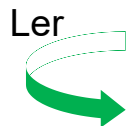
Última Pos.
de Sucesso



(0, A)
(0, B)
(1, B)
(0, C)
(3, C)

LZ78

A B A B C A B C A B C A A B C A B



- Símbolo: C
- Prefixo : AB

. Posição	. Prefixo
. 1	A
. 2	B
. 3	AB
. 4	C
. 5	ABC
. 6	

Última Pos.
de Sucesso



(0, A)
(0, B)
(1, B)
(0, C)
(3, C)

LZ78

A B A B C A B C A B C A A B C A B

Anexar



- Símbolo: C
- Prefixo : ABC

(0, A)
(0, B)
(1, B)
(0, C)
(3, C)

. Posição	. Prefixo
. 1	A
. 2	B
. 3	AB
. 4	C
. 5	ABC
. 6	

Última Pos.
de Sucesso

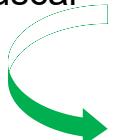


LZ78

A B A B C A B C A B C A A B C A B

- Símbolo: C
- Prefixo : ABC

Buscar



. Posição	. Prefixo
. 1	A
. 2	B
. 3	AB
. 4	C
. 5	ABC
. 6	

Última Pos.
de Sucesso

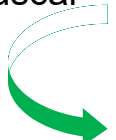
(0, A)
(0, B)
(1, B)
(0, C)
(3, C)

LZ78

A B A B C A B C A B C A A B C A B

- Símbolo: C
- Prefixo : ABC

Buscar



. Posição	. Prefixo
. 1	A
. 2	B
. 3	AB
. 4	C
. 5	ABC
. 6	

Última Pos.
de Sucesso

(0, A)
(0, B)
(1, B)
(0, C)
(3, C)

Sucesso

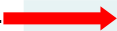
LZ78

A B A B C A B C A B C A A B C A B

- Símbolo: C
- Prefixo : ABC

. Posição	. Prefixo
. 1	A
. 2	B
. 3	AB
. 4	C
. 5	ABC
. 6	

Última Pos.
de Sucesso



(0, A)
(0, B)
(1, B)
(0, C)
(3, C)

LZ78

A B A B C A B C A B C A A B C A B

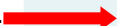
Ler



- Símbolo: A
- Prefixo : ABC

. Posição	. Prefixo
. 1	A
. 2	B
. 3	AB
. 4	C
. 5	ABC
. 6	

Última Pos.
de Sucesso



(0, A)
(0, B)
(1, B)
(0, C)
(3, C)

LZ78

A B A B C A B C A B C A A B C A B

Anexar



- Símbolo: A
- Prefixo : ABCA

. Posição	. Prefixo
. 1	A
. 2	B
. 3	AB
. 4	C
. 5	ABC
. 6	

Última Pos.
de Sucesso



(0, A)
(0, B)
(1, B)
(0, C)
(3, C)

LZ78

A B A B C A B C A B C A A B C A B

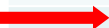
- Símbolo: A
- Prefixo : ABCA

Buscar



. Posição	. Prefixo
. 1	A
. 2	B
. 3	AB
. 4	C
. 5	ABC
. 6	

Última Pos.
de Sucesso



(0, A)
(0, B)
(1, B)
(0, C)
(3, C)

LZ78

A B A B C A B C A B C A A B C A B

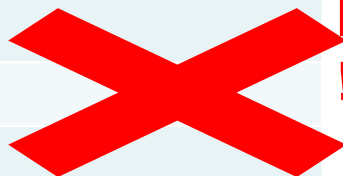
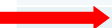
- Símbolo: A
- Prefixo : ABCA

Buscar



. Posição	. Prefixo
. 1	A
. 2	B
. 3	AB
. 4	C
. 5	ABC
. 6	

Última Pos.
de Sucesso



FALHA
!!!

(0, A)
(0, B)
(1, B)
(0, C)
(3, C)

LZ78

A B A B C A B C A B C A A B C A B

- Símbolo: A
- Prefixo : ABCA

. Posição	. Prefixo
. 1	A
. 2	B
. 3	AB
. 4	C
. 5	ABC
. 6	

Última Pos. de Sucesso →

Gerar token

(0, A)
(0, B)
(1, B)
(0, C)
(3, C)
(5, A)

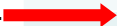
LZ78

A B A B C A B C A B C A A B C A B

- Símbolo: A
- Prefixo : ABCA

. Posição	. Prefixo
. 1	A
. 2	B
. 3	AB
. 4	C
. 5	ABC
. 6	ABCA

Última Pos.
de Sucesso



Inserir
prefixo

se Estruturas de Dados III

(0, A)
(0, B)
(1, B)
(0, C)
(3, C)
(5, A)

LZ78

A B A B C A B C A B C A A B C A B

- Símbolo:  Reinicializar
- Prefixo :

. Posição	. Prefixo
. 1	A
. 2	B
. 3	AB
. 4	C
. 5	ABC
. 6	ABCA

(0, A)
(0, B)
(1, B)
(0, C)
(3, C)
(5, A)

LZ78

A B A B C A B C A B C A A B C A B

- Símbolo:
- Prefixo : **ABCA**

. Posição	. Prefixo
. 1	A
. 2	B
. 3	AB
. 4	C
. 5	ABC
. 6	ABCA

(0, A)
(0, B)
(1, B)
(0, C)
(3, C)
(5, A)

LZ78

A B A B C A B C A B C A A B C A B

- Símbolo:
- Prefixo : **ABCA**

. Posição	. Prefixo
. 1	A
. 2	B
. 3	AB
. 4	C
. 5	ABC
. 6	ABCA

Gerar
token



(0, A)
(0, B)
(1, B)
(0, C)
(3, C)
(5, A)
(6, B)

LZ78

Arquivo original

A B A B C A B C A B C A ..



Arquivo compactado

0 A 0 B 1 B 0 C 3 C 5 A

ou

0 1 0 2 1 2 0 3 3 3 5 1

Obs: A \equiv 1 B \equiv 2 C \equiv 3

. Posição	. Prefixo
. 1	A
. 2	B
. 3	AB
. 4	C
. 5	ABC
. 6	ABCA

(0, A)
(0, B)
(1, B)
(0, C)
(3, C)
(5, A)

LZ78

- Descompactação
 - A medida que os **tokens** vão sendo lidos, o dicionário vai sendo reconstruído da mesma forma
 - Para cada **token** do arquivo compactado, é gerada como saída a sequência do dicionário indicada pelo campo posição concatenada com o símbolo armazenado no **token**

