

---

---

# Algoritmo LZW

## (Lempel – Ziv – Welch)

— Ing. Max Alejandro Antonio  
Cerna Flores —

---

---

# Agenda

Definición

¿Cómo trabaja?

Decodificación

# Definición

Compresión basada en diccionario, que no requiere información previa sobre el flujo de datos de entrada.

Se basa en patrones recurrentes para ahorrar espacio de datos, solo necesita un simple código o símbolo para representar una subcadena.

No tiene pérdidas, lo que significa que no se pierden datos al comprimir.

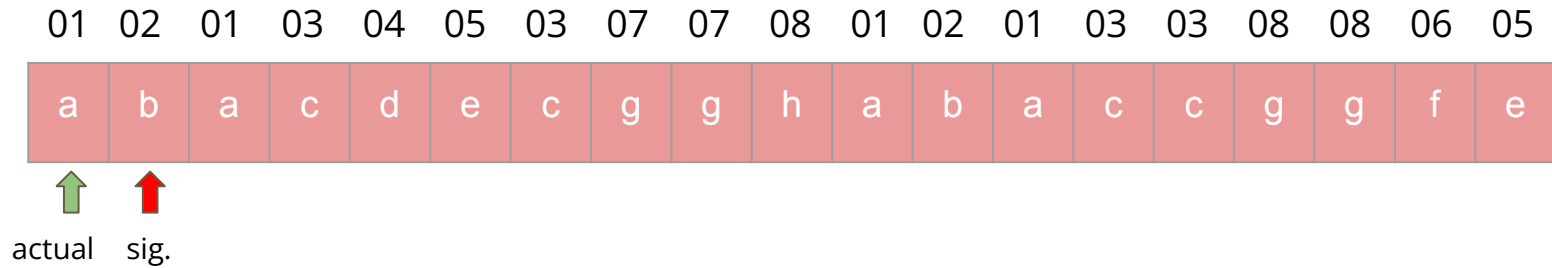
Se utiliza normalmente en GIF y, opcionalmente, en PDF y TIFF.

# ¿Cómo trabaja?

- Se define un diccionario dado, por ejemplo ASCII (8 bits - 256 caracteres)
- Para el ejemplo se trabajara con un diccionario de 8 caracteres (3 bits)

Bin	Dec	Sim
000	01	a
001	02	b
010	03	c
011	04	d
100	05	e
101	06	f
110	07	g
111	08	h

# ¿Cómo trabaja?



## Compresión (output)

01

## Diccionario (en memoria)

09 -> ab

# ¿Cómo trabaja?



## Compresión (output)

01 02

## Diccionario (en memoria)

09 -> ab

10 -> ba

# ¿Cómo trabaja?



## Compresión (output)

01 02 01

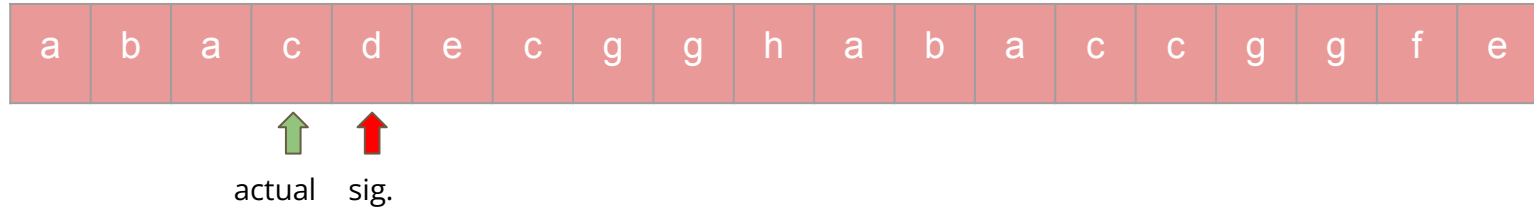
## Diccionario (en memoria)

09 -> ab

10 -> ba

11 -> ac

# ¿Cómo trabaja?



## Compresión (output)

01 02 01 03

## Diccionario (en memoria)

09 -> ab

10 -> ba

11 -> ac

12 -> cd



# ¿Cómo trabaja?



## Compresión (output)

01 02 01 03 04

## Diccionario (en memoria)

09 -> ab

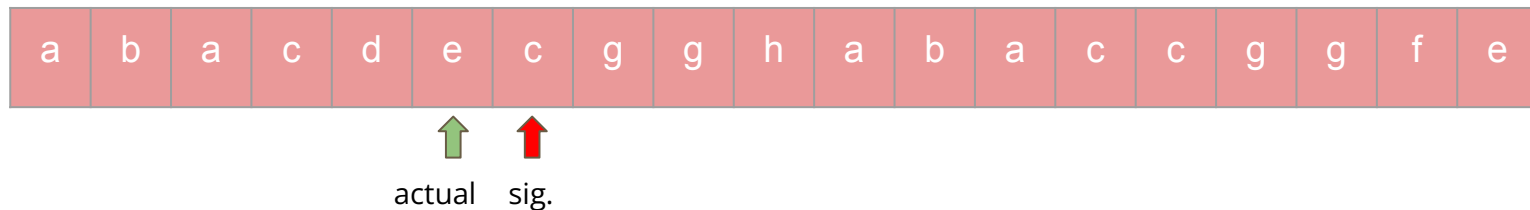
10 -> ba

11 -> ac

12 -> cd

13 -> de

# ¿Cómo trabaja?



## Compresión (output)

01 02 01 03 04 05

## Diccionario (en memoria)

09 -> ab

10 -> ba

11 -> ac

12 -> cd

13 -> de

14 -> ec

# ¿Cómo trabaja?



## Compresión (output)

01 02 01 03 04 05 03

## Diccionario (en memoria)

09 -> ab

10 -> ba

11 -> ac

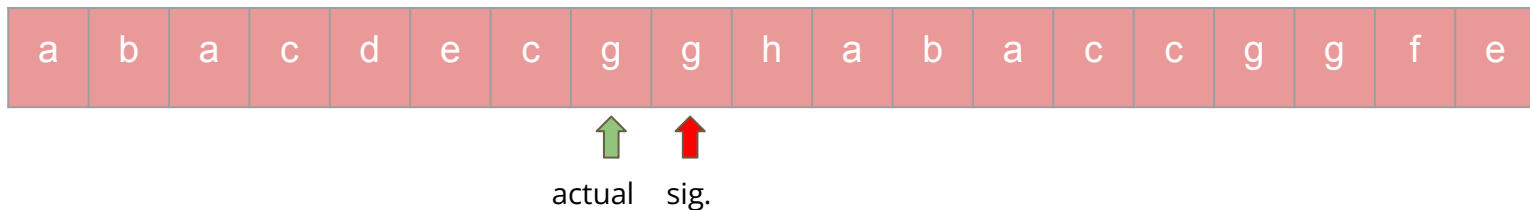
12 -> cd

13 -> de

14 -> ec

15 -> cg

# ¿Cómo trabaja?



## Compresión (output)

01 02 01 03 04 05 03 07

## Diccionario (en memoria)

09 -> ab

10 -> ba

11 -> ac

12 -> cd

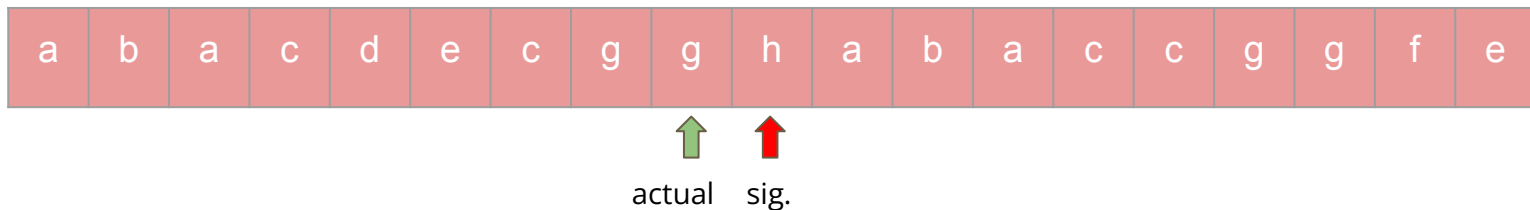
13 -> de

14 -> ec

15 -> cg

16 -> gg

# ¿Cómo trabaja?



## Compresión (output)

01 02 01 03 04 05 03 07 07

## Diccionario (en memoria)

09 -> ab

10 -> ba

11 -> ac

12 -> cd

13 -> de

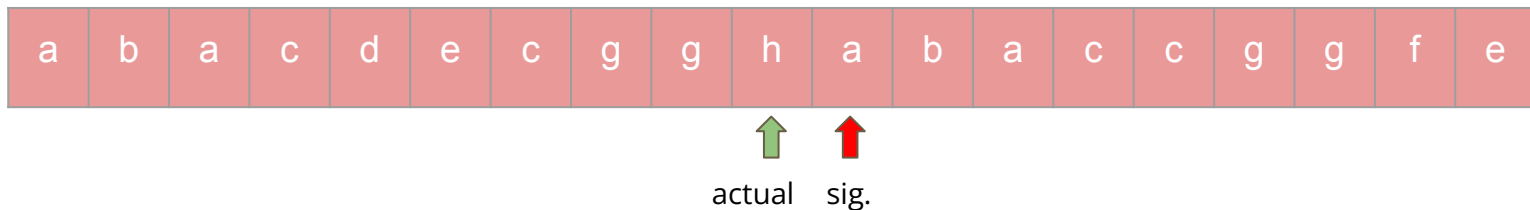
14 -> ec

15 -> cg

16 -> gg

17 -> gh

# ¿Cómo trabaja?



## Compresión (output)

01 02 01 03 04 05 03 07 07 08

## Diccionario (en memoria)

09 -> ab

10 -> ba

11 -> ac

12 -> cd

13 -> de

14 -> ec

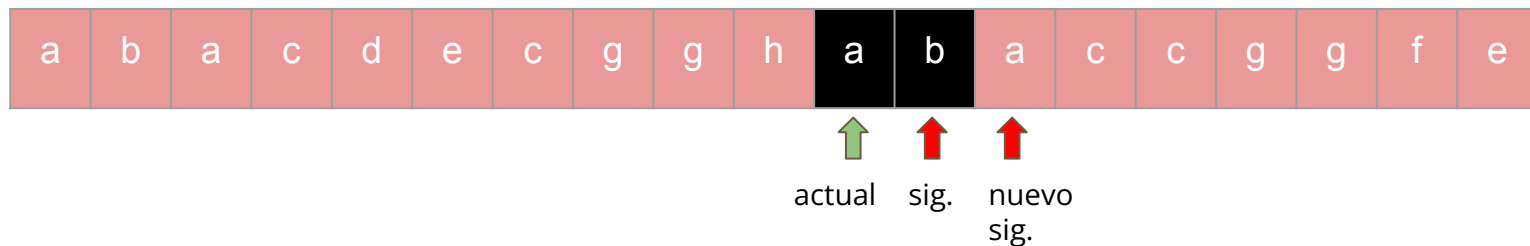
15 -> cg

16 -> gg

17 -> gh

19 -> ha

# ¿Cómo trabaja?



## Compresión (output)

01 02 01 03 04 05 03 07 07 08 09

## Diccionario (en memoria)

09 -> ab

10 -> ba

11 -> ac

12 -> cd

13 -> de

14 -> ec

15 -> cg

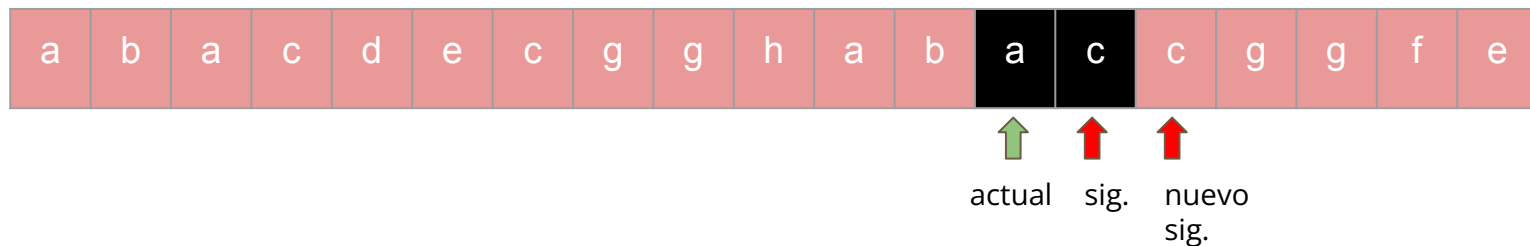
16 -> gg

17 -> gh

18 -> ha

19 -> aba

# ¿Cómo trabaja?



## Compresión (output)

01 02 01 03 04 05 03 07 07 08 09 **11**

## Diccionario (en memoria)

09 -> ab

10 -> ba

**11 -> ac**

12 -> cd

13 -> de

14 -> ec

15 -> cg

16 -> gg

17 -> gh

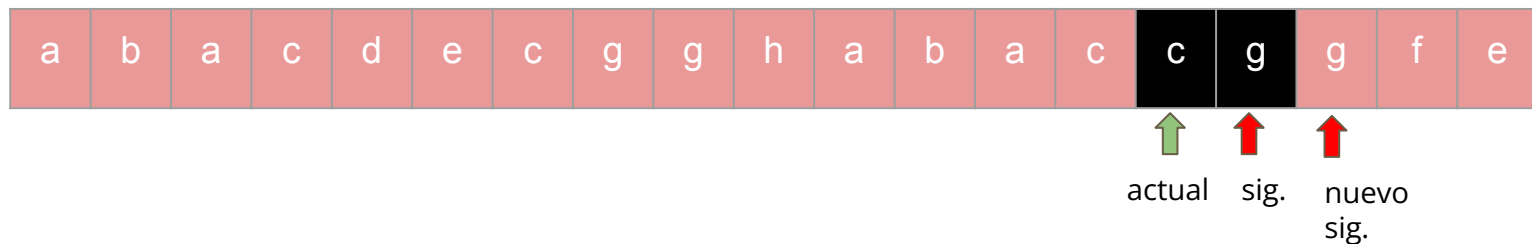
18 -> ha

19 -> aba

**20 -> acc**



# ¿Cómo trabaja?



## Compresión (output)

01 02 01 03 04 05 03 07 07 08 09 11 15

## Diccionario (en memoria)

09 -> ab

10 -> ba

11 -> ac

12 -> cd

13 -> de

14 -> ec

15 -> **cg**

16 -> gg

17 -> gh

18 -> ha

29 -> aba

20 -> acc

21 -> **cgg**

# ¿Cómo trabaja?



## Compresión (output)

01 02 01 03 04 05 03 07 07 08 09 11 15 07

## Diccionario (en memoria)

09 -> ab

10 -> ba

11 -> ac

12 -> cd

13 -> de

14 -> ec

15 -> cg

16 -> gg

17 -> gh

18 -> ha

19 -> aba

20 -> acc

21 -> cgg

22 -> gf

# ¿Cómo trabaja?



## Compresión (output)

01 02 01 03 04 05 03 07 07 08 09 11 15 07 06

## Diccionario (en memoria)

09 -> ab

10 -> ba

11 -> ac

12 -> cd

13 -> de

14 -> ec

15 -> cg

16 -> gg

17 -> gh

18 -> ha

19 -> aba

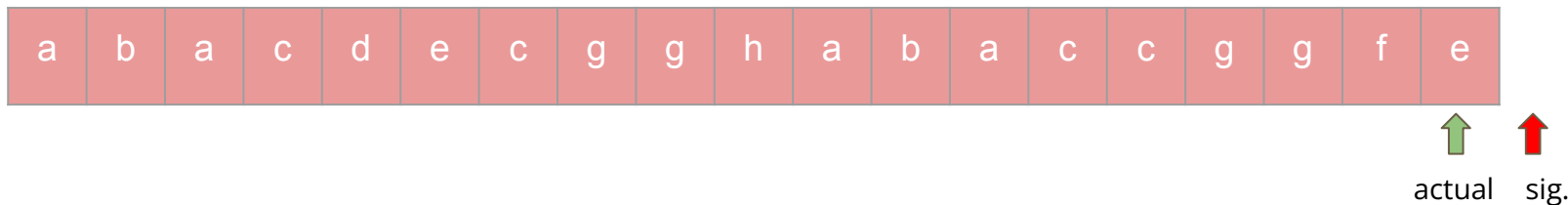
20 -> acc

21 -> cgg

22 -> gf

23 -> fe

# ¿Cómo trabaja?



## Compresión (output)

01 02 01 03 04 05 03 07 07 08 09 11 15 07 06 05 eof

## Diccionario (en memoria)

09 -> ab

10 -> ba

11 -> ac

12 -> cd

13 -> de

14 -> ec

15 -> cg

16 -> gg

17 -> gh

18 -> ha

19 -> aba

20 -> acc

21 -> cgg

22 -> gf

23 -> fe

# ¿Cómo trabaja?

## **input**

01 02 01 03 04 05 03 07 07 08 01 02 01 03 03 08 08 06 05 eof

## **Compresión (output)**

01 02 01 03 04 05 03 07 07 08 09 11 15 07 06 05 eof

# Descompresión

01	02	01	03	04	05	03	07	07	08	09	11	15	07	06	05
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----



actual sig.

**output:**



01

**Diccionario (en memoria)**

09 -> 01 02 (ab)

# Descompresión

01	02	01	03	04	05	03	07	07	08	09	11	15	07	06	05
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

   
actual sig.

**output:**

01 02

**Diccionario (en memoria)**

09 -> 01 02 (ab)

10 -> 02 01 (ba)

# Descompresión



**output:**

01 02 01

## **Diccionario (en memoria)**

09 -> 01 02 (ab)



10 -> 02 01 (ba)

11 -> 01 03 (ac)



# Descompresión

01	02	01	03	04	05	03	07	07	08	09	11	15	07	06	05
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

   
actual sig.

**output:**

01 02 01 03

## **Diccionario (en memoria)**

09 -> 01 02 (ab)

10 -> 02 01 (ba)

11 -> 01 03 (ac)

12 -> 03 04 (cd)

# Descompresión

01	02	01	03	04	05	03	07	07	08	09	11	15	07	06	05
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----



actual



sig.

**output:**

01 02 01 03 04

**Diccionario (en memoria)**

09 -> 01 02 (ab)

10 -> 02 01 (ba)

11 -> 01 03 (ac)

12 -> 03 04 (cd)

13 -> 04 05

# Descompresión

01	02	01	03	04	05	03	07	07	08	09	11	15	07	06	05
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----



actual

sig.

**output:**

01 02 01 03 04 05

## Diccionario (en memoria)

09 -> 01 02 (ab)

10 -> 02 01 (ba)

11 -> 01 03 (ac)



12 -> 03 04 (cd)

13 -> 04 05 (de)

14 -> 05 03 (ec)

# Descompresión

01	02	01	03	04	05	03	07	07	08	09	11	15	07	06	05
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

   
actual sig.

**output:**

01 02 01 03 04 05 03

## Diccionario (en memoria)

09 -> 01 02 (ab)

10 -> 02 01 (ba)

11 -> 01 03 (ac)

12 -> 03 04 (cd)



13 -> 04 05 (de)

14 -> 05 03 (ec)

15 -> 03 07 (cg)

# Descompresión

01	02	01	03	04	05	03	07	07	08	09	11	15	07	06	05
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

   
actual sig.

**output:**

01 02 01 03 04 05 03 07

## Diccionario (en memoria)

09 -> 01 02 (ab)

10 -> 02 01 (ba)

11 -> 01 03 (ac)

12 -> 03 04 (cd)

13 -> 04 05 (de)



14 -> 05 03 (ec)

15 -> 03 07 (cg)

16 -> 07 07 (gg)

# Descompresión

01	02	01	03	04	05	03	07	07	08	09	11	15	07	06	05
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

   
actual sig.

**output:**

01 02 01 03 04 05 03 07 07

## Diccionario (en memoria)

09 -> 01 02 (ab)

10 -> 02 01 (ba)

11 -> 01 03 (ac)

12 -> 03 04 (cd)

13 -> 04 05 (de)

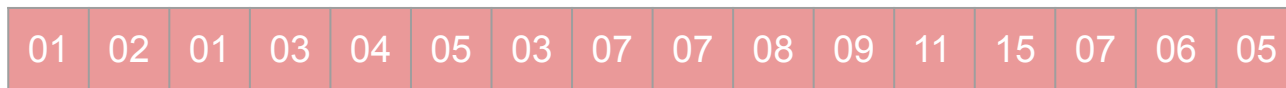
14 -> 05 03 (ec)



15 -> 03 07 (cg)

16 -> 07 07 (gg)

17 -> 07 08 (gh)

# Descompresión



   
actual sig.

**output:**

01 02 01 03 04 05 03 07 07 08

## Diccionario (en memoria)

09 -> 01 02 (ab)

10 -> 02 01 (ba)

11 -> 01 03 (ac)

12 -> 03 04 (cd)

13 -> 04 05 (de)

14 -> 05 03 (ec)

15 -> 03 07 (cg)

16 -> 07 07 (gg)

17 -> 07 08 (gh)

18 -> 08 09 (hab)

# Descompresión

01	02	01	03	04	05	03	07	07	08	09	11	15	07	06	05
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----



actual

sig.

**output:**

01 02 01 03 04 05 03 07 07 08 **01 02**

## Diccionario (en memoria)

**09 -> 01 02 (ab)**

10 -> 02 01 (ba)

11 -> 01 03 (ac)

12 -> 03 04 (cd)

13 -> 04 05 (de)

14 -> 05 03 (ec)

15 -> 03 07 (cg)

16 -> 07 07 (gg)

17 -> 07 08 (gh)

18 -> 08 09 (hab)

19 -> 09 11 (abac)



# Descompresión

01	02	01	03	04	05	03	07	07	08	09	11	15	07	06	05
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----



actual



sig.

**output:**

01 02 01 03 04 05 03 07 07 08 01 02 **01 03**

## Diccionario (en memoria)

09 -> 01 02 (ab)

10 -> 02 01 (ba)

**11 -> 01 03 (ac)**

12 -> 03 04 (cd)

13 -> 04 05 (de)

14 -> 05 03 (ec)

15 -> 03 07 (cg)

16 -> 07 07 (gg)

17 -> 07 08 (gh)

18 -> 08 09 (hab)

19 -> 09 11 (abac)

20 -> 11 15 (accg)

# Descompresión

01	02	01	03	04	05	03	07	07	08	09	11	15	07	06	05
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----



actual



sig.

**output:**

01 02 01 03 04 05 03 07 07 08 01 02 01 03 **03 07**

## Diccionario (en memoria)

09 -> 01 02 (ab)

10 -> 02 01 (ba)

11 -> 01 03 (ac)

12 -> 03 04 (cd)

13 -> 04 05 (de)

14 -> 05 03 (ec)

**15 -> 03 07 (cg)**

16 -> 07 07 (gg)

17 -> 07 08 (gh)

18 -> 08 09 (hab)

19 -> 09 11 (abac)

20 -> 11 15 (accg)

21 -> 15 07 (cgg)

# Descompresión

01	02	01	03	04	05	03	07	07	08	09	11	15	07	06	05
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----



actual



sig.

**output:**

01 02 01 03 04 05 03 07 07 08 01 02 01 03 03 07 07

## Diccionario (en memoria)

09 -> 01 02 (ab)

10 -> 02 01 (ba)

11 -> 01 03 (ac)

12 -> 03 04 (cd)

13 -> 04 05 (de)

14 -> 05 03 (ec)

15 -> 03 07 (cg)

16 -> 07 07 (gg)

17 -> 07 08 (gh)

18 -> 08 09 (hab)

19 -> 09 11 (abac)

20 -> 11 15 (accg)

21 -> 15 07 (cgg)

22 -> 07 06 (gf)

# Descompresión

01	02	01	03	04	05	03	07	07	08	09	11	15	07	06	05
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----



actual



sig.

**output:**

01 02 01 03 04 05 03 07 07 08 01 02 01 03 03 07 07 06

## Diccionario (en memoria)

09 -> 01 02 (ab)

10 -> 02 01 (ba)

11 -> 01 03 (ac)

12 -> 03 04 (cd)

13 -> 04 05 (de)

14 -> 05 03 (ec)

15 -> 03 07 (cg)

16 -> 07 07 (gg)

17 -> 07 08 (gh)

18 -> 08 09 (hab)

19 -> 09 11 (abac)

20 -> 11 15 (accg)

21 -> 15 07 (cgg)

22 -> 07 06 (gf)

23 -> 06 05 (fe)

# Descompresión

01	02	01	03	04	05	03	07	07	08	09	11	15	07	06	05
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----



actual



sig.

**output:**

01 02 01 03 04 05 03 07 07 08 01 02 01 03 03 07 07 06 05

## Diccionario (en memoria)

09 -> 01 02 (ab)

10 -> 02 01 (ba)

11 -> 01 03 (ac)

12 -> 03 04 (cd)

13 -> 04 05 (de)

14 -> 05 03 (ec)

15 -> 03 07 (cg)

16 -> 07 07 (gg)

17 -> 07 08 (gh)

18 -> 08 09 (hab)

19 -> 09 11 (abac)

20 -> 11 15 (accg)

21 -> 15 07 (cgg)

22 -> 07 06 (gf)

23 -> 06 05 (fe)

# Descompresión

## compresión

01 02 01 03 04 05 03 07 07 08 09 11 15 07 06 05

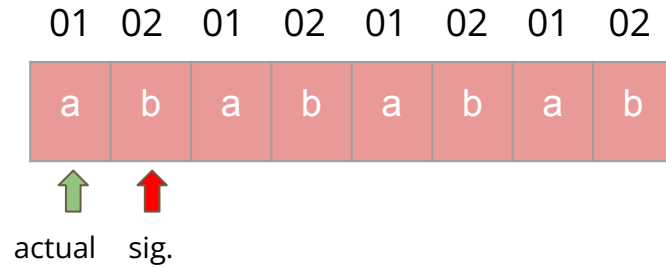
## resultado

01 02 01 03 04 05 03 07 07 08 01 02 01 03 03 07 07 06 05  
a b a c d e c g g h a b a c c g g f e

## mensaje original

a	b	a	c	d	e	c	g	g	h	a	b	a	c	c	g	g	f	e
---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---	---

# Otro ejemplo



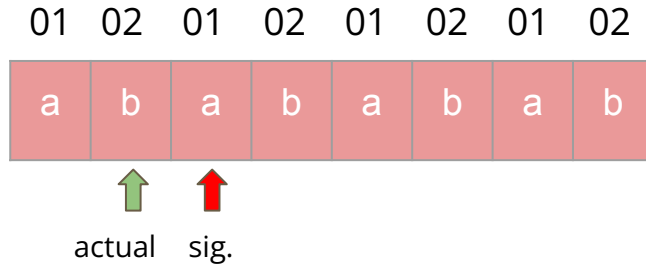
**Compresión (output)**

01

**Diccionario (en memoria)**

09 -> ab

# Otro ejemplo



## Compresión (output)

01 02

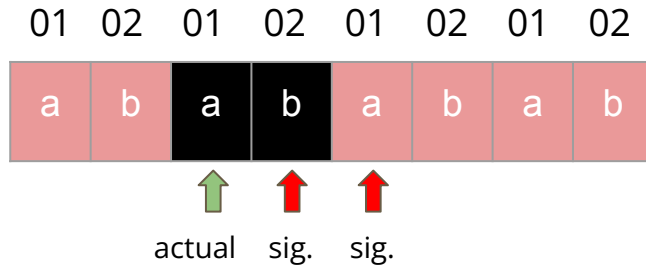
## Diccionario (en memoria)

09 -> ab

10 -> ba



## Otro ejemplo



### Compresión (output)

01 02 09

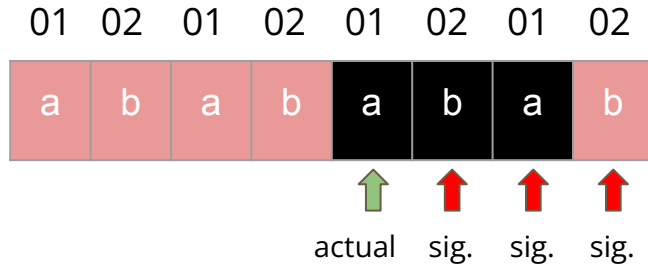
### Diccionario (en memoria)

09 -> ab

10 -> ba

11 -> aba

## Otro ejemplo



### Compresión (output)

01 02 09 11

### Diccionario (en memoria)

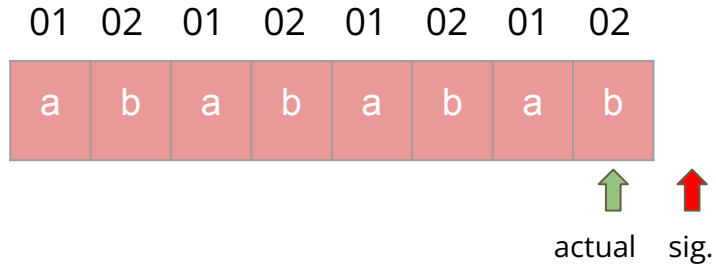
09 -> ab

10 -> ba

11 -> aba

12 -> abab

# Otro ejemplo



## Compresión (output)

01 02 09 11 02

## Diccionario (en memoria)

09 -> ab

10 -> ba

11 -> aba

12 -> abab

# Otro ejemplo

**input**

01 02 01 02 01 02 01 02

**Compresión (output)**

01 02 09 11 02

# Descompresión

01	02	09	11	02
----	----	----	----	----



actual sig.

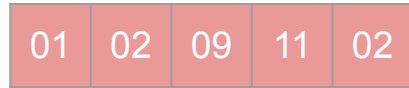
**output:**

01

**Diccionario (en memoria)**

09 -> 01 02 (ab)

# Descompresión



**output:**

01 02

**Diccionario (en memoria)**

09 -> 01 02 (ab)

10 -> 02 01 (ba)

# Descompresión

01	02	09	11	02
----	----	----	----	----

↑    ↑  
actual sig.

**output:**

01 02 **01 02**

**Diccionario (en memoria)**

09 -> 01 02 (ab)

10 -> 02 01 (ba)



11 -> 09 11 ?

**Solución**

09 (ab) + 01 (a) -> 11 => aba

# Descompresión

01	02	09	11	02
----	----	----	----	----

   
actual sig.

**output:**

01 02 01 02 **01 02 01**

**Diccionario (en memoria)**

09 -> 01 02 (ab)

10 -> 02 01 (ba)

**11 -> 09 01 (aba)**



# Descompresión

01	02	09	11	02
----	----	----	----	----



actual



sig.

**output:**

01 02 01 02 01 02 01 **02**

**Diccionario (en memoria)**

09 -> 01 02 (ab)

10 -> 02 01 (ba)

11 -> 09 01 (aba)

# Descompresión

## compresión

01 02 09 11 02

## resultado

01 02 01 02 01 02 01 02  
a b a b a b a b

## mensaje original

a	b	a	b	a	b	a	b
---	---	---	---	---	---	---	---