



Posgrado en  
**Biología  
Integrativa**



# Transcriptómica

*Clase 4 - Alineamiento y cuantificación*

Biología Computacional 2019

Selene L. Fernández-Valverde

[regRNAlab.github.io](https://github.com/regRNAlab/regRNAlab)

@Selfdz

# Objetivos de aprendizaje

En esta clase aprenderemos:

- A alinear datos de RNA-Seq a una referencia
  - Lecturas crudas
  - Transcritos generados *de novo*
- Entender los formatos SAM y BAM.

# Ensamblando transcriptomas

- Tophat

- Bowtie

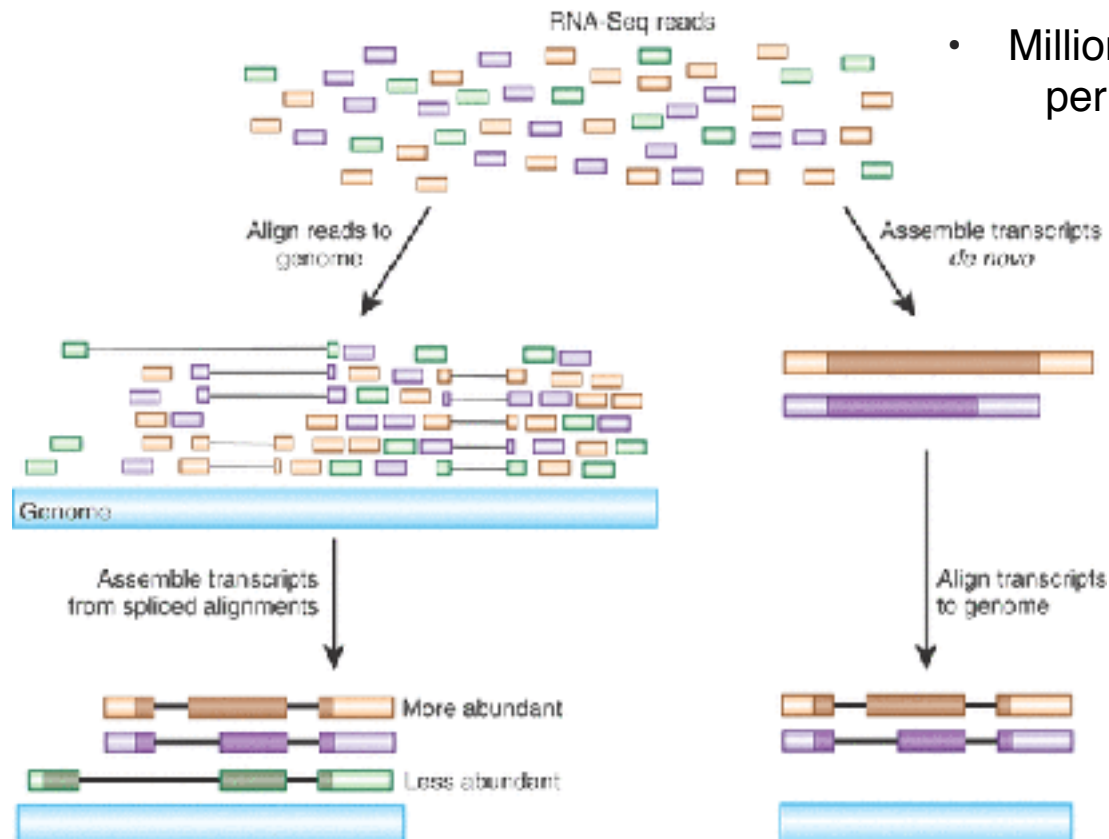
- HiSat

- STAR

- StringTie

- Cufflinks

- Scripture



- Millions of reads per sample !

- Trinity

- Oases de Novo

- trans-ABYSS

- GMAP

- Blat

- Exonerate

# Ensamblando transcriptomas

- Tophat

- Bowtie

- HiSat

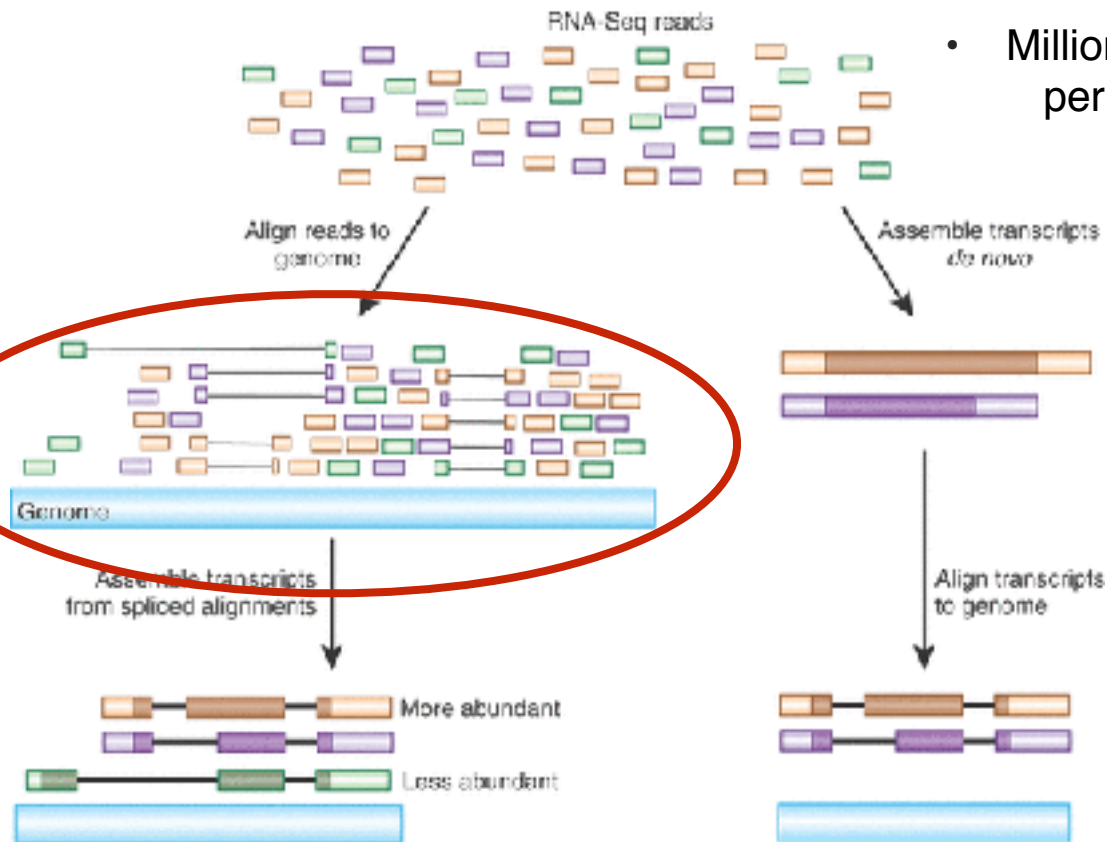
- STAR

- StringTie

- Cufflinks

- Scripture

- Millions of reads per sample !



- Trinity
- Oases de Novo
- trans-ABYSS
- GMAP
- Blat
- Exonerate

# ¿Qué significa alinear (mapear) una secuencia?

- Es identificar la posición de origen (alta similitud) de **lecturas** o transcritos secuenciados en una **secuencia de referencia** (genomas o transcritos)

## Secuencia de referencia

GATCACGAAAGCACTTTACTGGGTAAATAAAGTAC

|||||

CGAAAGCACTTTATTGG

Lectura



Error o Mismatch

# No podemos usar BLAST

- BLAST hace un alineamiento local, lo cual lo hace muy útil para buscar alineamientos parciales y/o divergentes en bases de datos grandes.
- BLAST es muy lento para alinear secuencias, lo que lo hace poco práctico alinear millones de secuencias.
- Dado que generalmente esperamos un alto nivel de similitud con la referencia en un experimento de secuenciación masiva necesitamos un algoritmo de alineamiento semi-global y muy rápido.

# Burrows-Wheeler transform (BWT)

- Descubierta por David Wheeler en 1983.
- Permutación reversible de los caracteres en una cadena - usada originalmente para comprimir datos.
- En 2005 se encontró que era extremadamente útil para encontrar subcadenas.
- En 2009 se comenzó a usar para alinear lecturas resultado de experimentos de secuenciación masiva.
- En conjunto con índices comprimidos (e.g. FM index) permite que el tiempo de alineamiento crece de manera lineal con la cantidad de secuencias.
- Permite alinear ~100 millones de lecturas por hora (Bowtie - 1 solo thread)

# Generando una BWT

ATCTTATC\$

*\$ - Caracter que indica el final de una cadena*



# Generando una BWT

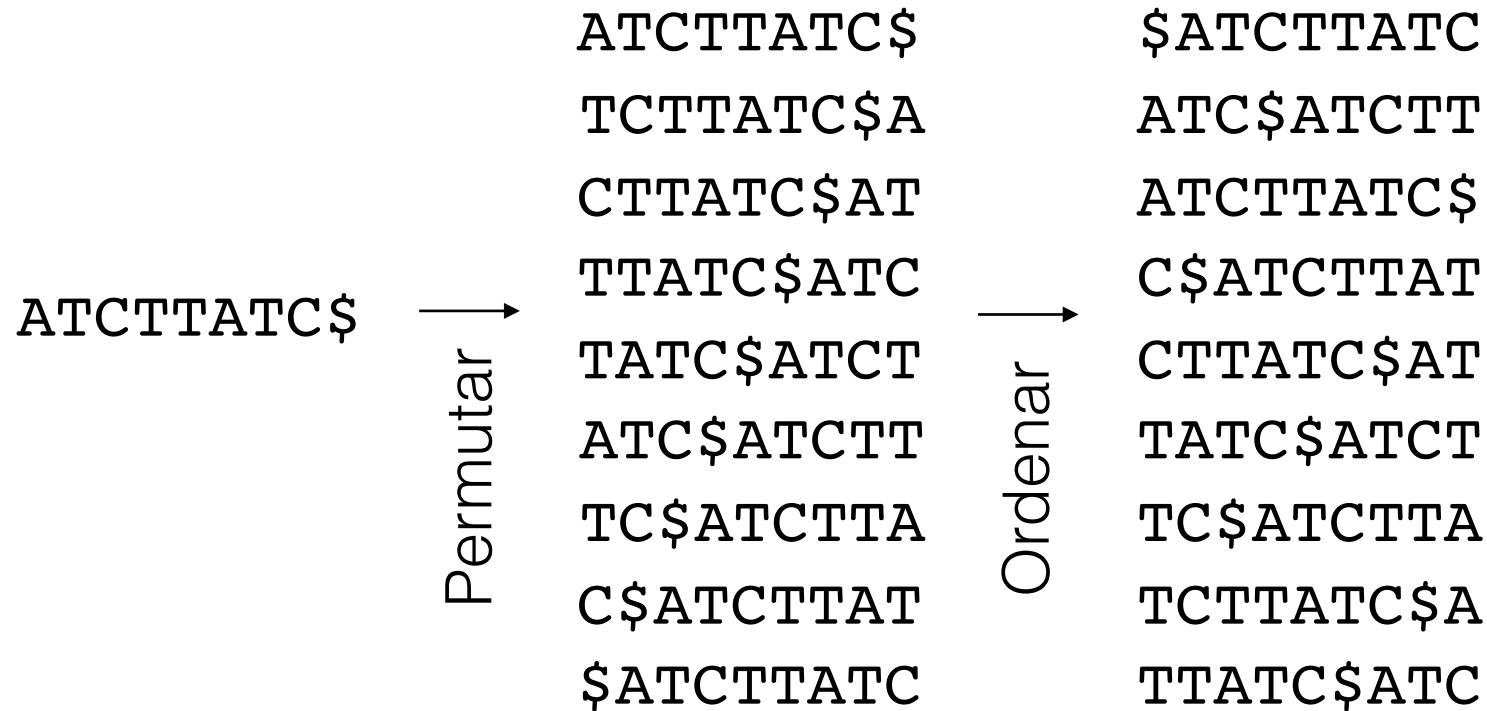
ATCTTATC\$

↓  
Permutar

ATCTTATC\$  
TCTTATC\$A  
CTTATC\$AT  
TTATC\$ATC  
TATC\$ATCT  
ATC\$ATCTT  
TC\$ATCTTA  
C\$ATCTTAT  
\$ATCTTATC

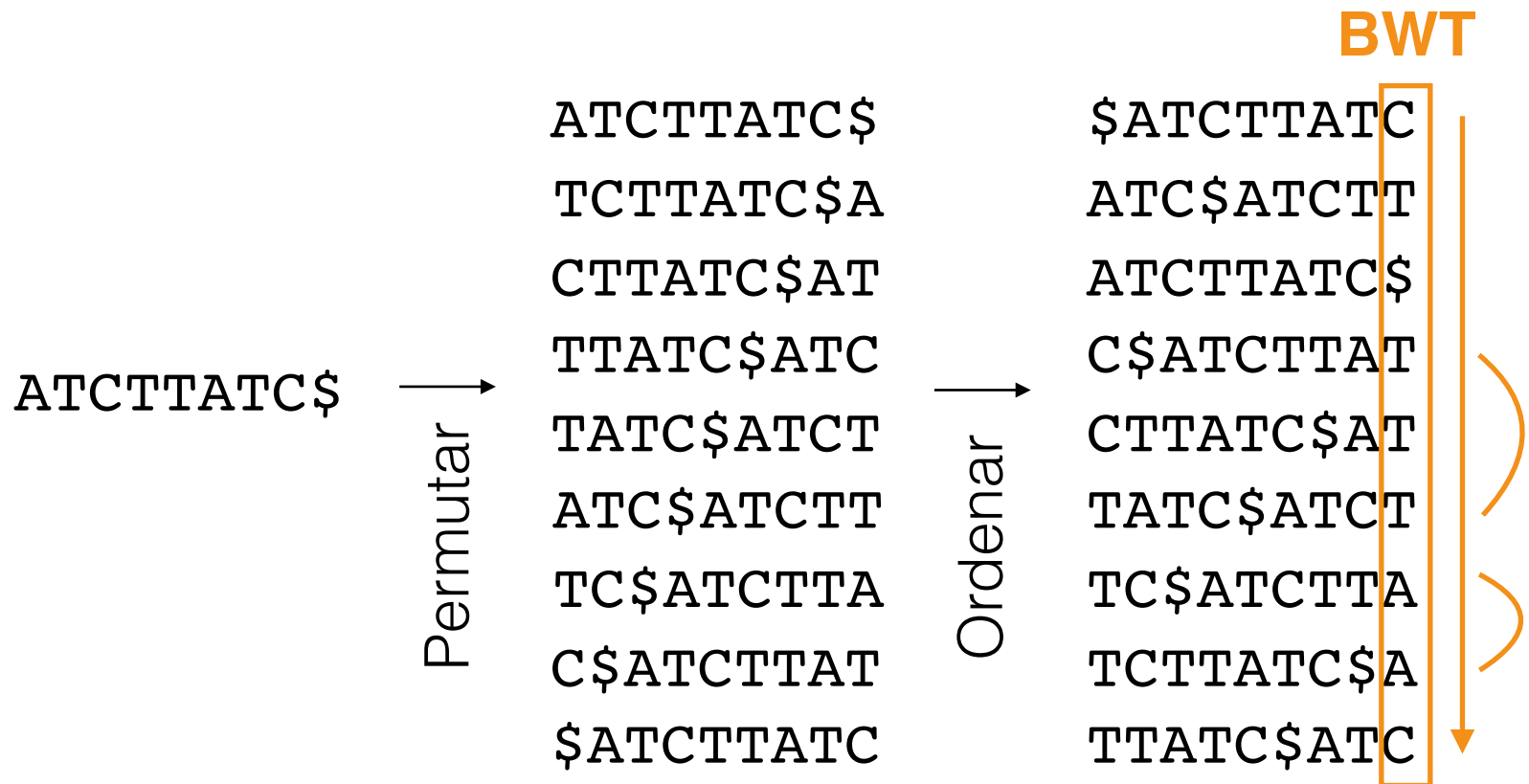
\$ - Caracter que indica el final de una cadena

# Generando una BWT



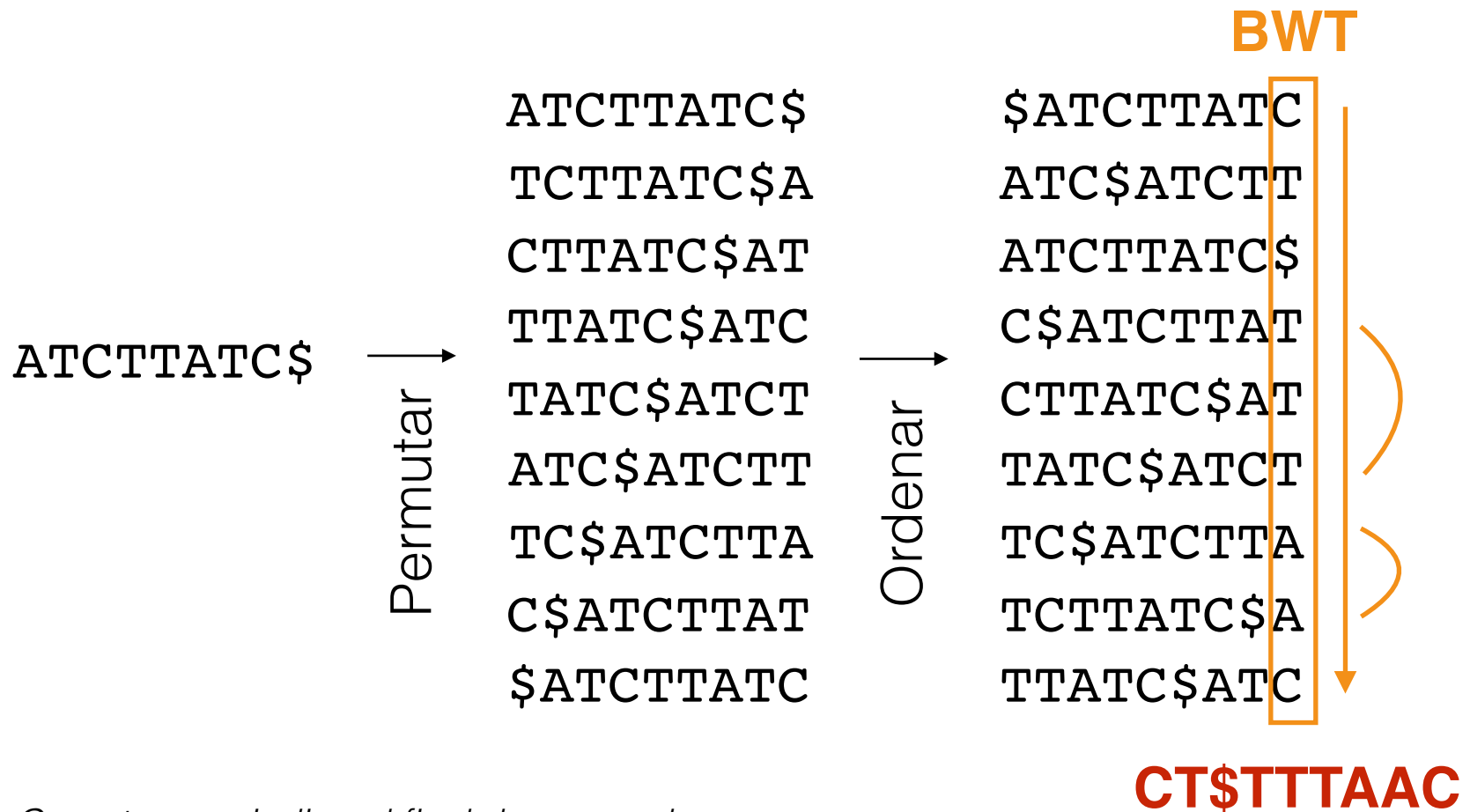
\$ - Caracter que indica el final de una cadena

# Generando una BWT



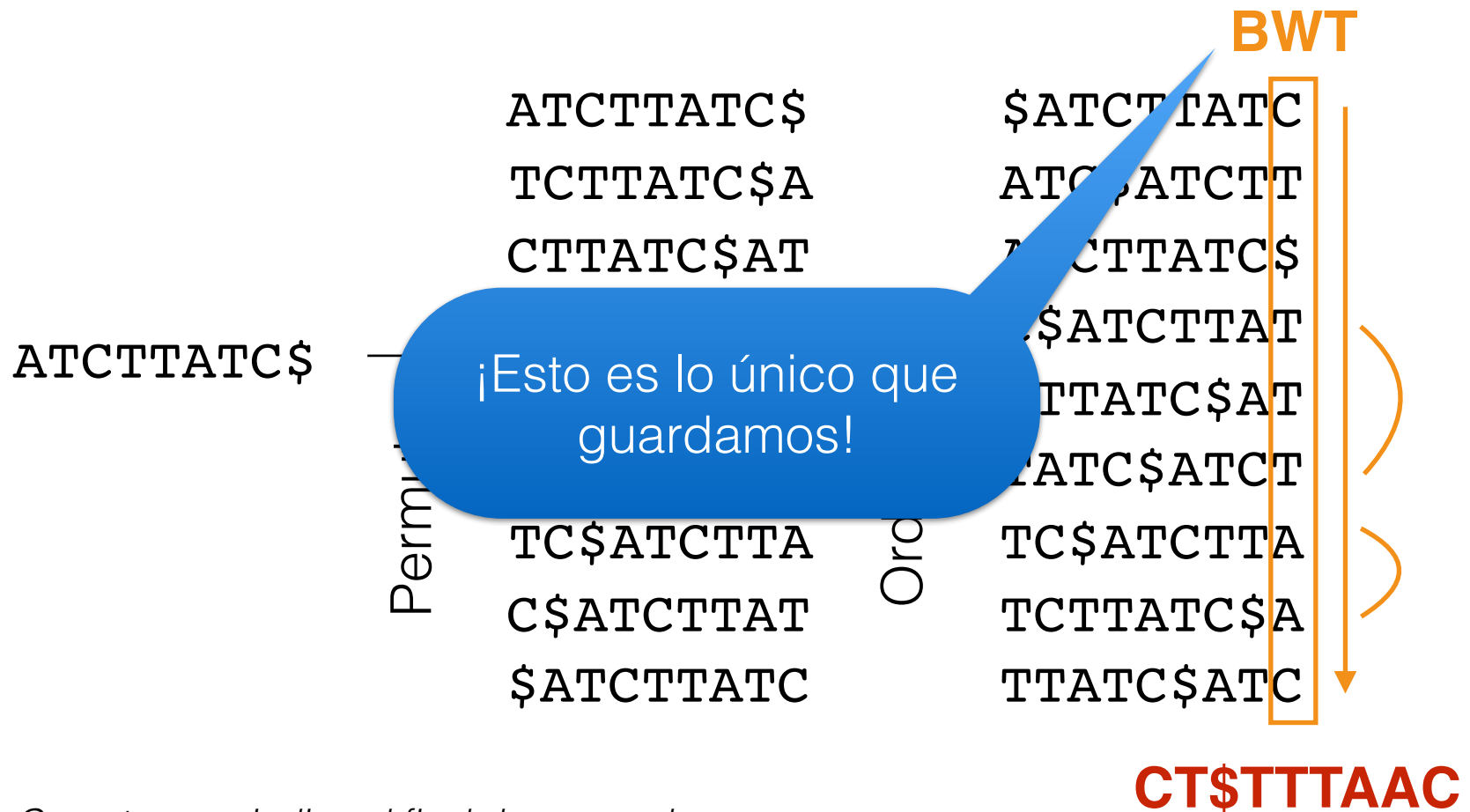
\$ - Caracter que indica el final de una cadena

# Generando una BWT



\$ - Caracter que indica el final de una cadena

# Generando una BWT



\$ - Caracter que indica el final de una cadena

# Propiedad FT

Renglón

0	\$ <sub>0</sub>	ATCTTAT	C <sub>0</sub>
1	A <sub>0</sub>	TC\$ATCT	T <sub>0</sub>
2	A <sub>1</sub>	TCTTATC	\$ <sub>0</sub>
3	C <sub>0</sub>	\$ATCTTA	T <sub>1</sub>
4	C <sub>1</sub>	TTATC\$A	T <sub>2</sub>
5	T <sub>0</sub>	ATC\$ATC	T <sub>3</sub>
6	T <sub>1</sub>	C\$ATCTT	A <sub>0</sub>
7	T <sub>2</sub>	CTTATC\$	A <sub>1</sub>
8	T <sub>3</sub>	TATC\$AT	C <sub>1</sub>

# Propiedad FT

Renglón

BWT

0	\$ <sub>0</sub>	ATCTTAT	C <sub>0</sub>
1	A <sub>0</sub>	TC\$ATCT	T <sub>0</sub>
2	A <sub>1</sub>	TCTTATC	\$ <sub>0</sub>
3	C <sub>0</sub>	\$ATCTTA	T <sub>1</sub>
4	C <sub>1</sub>	TTATC\$A	T <sub>2</sub>
5	T <sub>0</sub>	ATC\$ATC	T <sub>3</sub>
6	T <sub>1</sub>	C\$ATCTT	A <sub>0</sub>
7	T <sub>2</sub>	CTTATC\$	A <sub>1</sub>
8	T <sub>3</sub>	TATC\$AT	C <sub>1</sub>

*F- First*

*L- Last*

# Propiedad FT

Renglón

0	\$ <sub>0</sub>	ATCTTAT	C <sub>0</sub>
1	A <sub>0</sub>	TC\$ATCT	T <sub>0</sub>
2	A <sub>1</sub>	TCTTATC	\$ <sub>0</sub>
3	C <sub>0</sub>	\$ATCTTA	T <sub>1</sub>
4	C <sub>1</sub>	TTATC\$A	T <sub>2</sub>
5	T <sub>0</sub>	ATC\$ATC	T <sub>3</sub>
6	T <sub>1</sub>	C\$ATCTT	A <sub>0</sub>
7	T <sub>2</sub>	CTTATC\$	A <sub>1</sub>
8	T <sub>3</sub>	TATC\$AT	C <sub>1</sub>

**BWT**

El rango de los caracteres se mantiene en la primera (F) y última (L) columna.

La primera columna se puede reconstruir ordenando la última

*F- First*

*L- Last*



# Revirtiendo la transformación BWT

Renglón

0	$\$0$	$C_0$
1	$A_0$	$T_0$
2	$A_1$	$\$0$
3	$C_0$	$T_1$
4	$C_1$	$T_2$
5	$T_0$	$T_3$
6	$T_1$	$A_0$
7	$T_2$	$A_1$
8	$T_3$	$C_1$

Secuencia original

# Revirtiendo la transformación BWT

Renglón

0	\$ <sub>0</sub>	C <sub>0</sub>
1	A <sub>0</sub>	T <sub>0</sub>
2	A <sub>1</sub>	\$ <sub>0</sub>
3	C <sub>0</sub>	T <sub>1</sub>
4	C <sub>1</sub>	T <sub>2</sub>
5	T <sub>0</sub>	T <sub>3</sub>
6	T <sub>1</sub>	A <sub>0</sub>
7	T <sub>2</sub>	A <sub>1</sub>
8	T <sub>3</sub>	C <sub>1</sub>

Secuencia original

# Revirtiendo la transformación BWT

Renglón

0	\$ <sub>0</sub>
1	A <sub>0</sub>
2	A <sub>1</sub>
3	C <sub>0</sub>
4	C <sub>1</sub>
5	T <sub>0</sub>
6	T <sub>1</sub>
7	T <sub>2</sub>
8	T <sub>3</sub>

C <sub>0</sub>
T <sub>0</sub>
\$ <sub>0</sub>
T <sub>1</sub>
T <sub>2</sub>
T <sub>3</sub>
A <sub>0</sub>
A <sub>1</sub>
C <sub>1</sub>

\$<sub>0</sub>  
Secuencia original

# Revirtiendo la transformación BWT

Renglón

0	\$ <sub>0</sub>	→	C <sub>0</sub>
1	A <sub>0</sub>		T <sub>0</sub>
2	A <sub>1</sub>		\$ <sub>0</sub>
3	C <sub>0</sub>		T <sub>1</sub>
4	C <sub>1</sub>		T <sub>2</sub>
5	T <sub>0</sub>		T <sub>3</sub>
6	T <sub>1</sub>		A <sub>0</sub>
7	T <sub>2</sub>		A <sub>1</sub>
8	T <sub>3</sub>		C <sub>1</sub>

C<sub>0</sub> \$<sub>0</sub>

Secuencia original

# Revirtiendo la transformación BWT

Renglón

0	\$ <sub>0</sub>	→	C <sub>0</sub>
1	A <sub>0</sub>		T <sub>0</sub>
2	A <sub>1</sub>	↙	\$ <sub>0</sub>
3	C <sub>0</sub>		T <sub>1</sub>
4	C <sub>1</sub>		T <sub>2</sub>
5	T <sub>0</sub>		T <sub>3</sub>
6	T <sub>1</sub>		A <sub>0</sub>
7	T <sub>2</sub>		A <sub>1</sub>
8	T <sub>3</sub>		C <sub>1</sub>

C<sub>0</sub> \$<sub>0</sub>

Secuencia original

# Revirtiendo la transformación BWT

Renglón

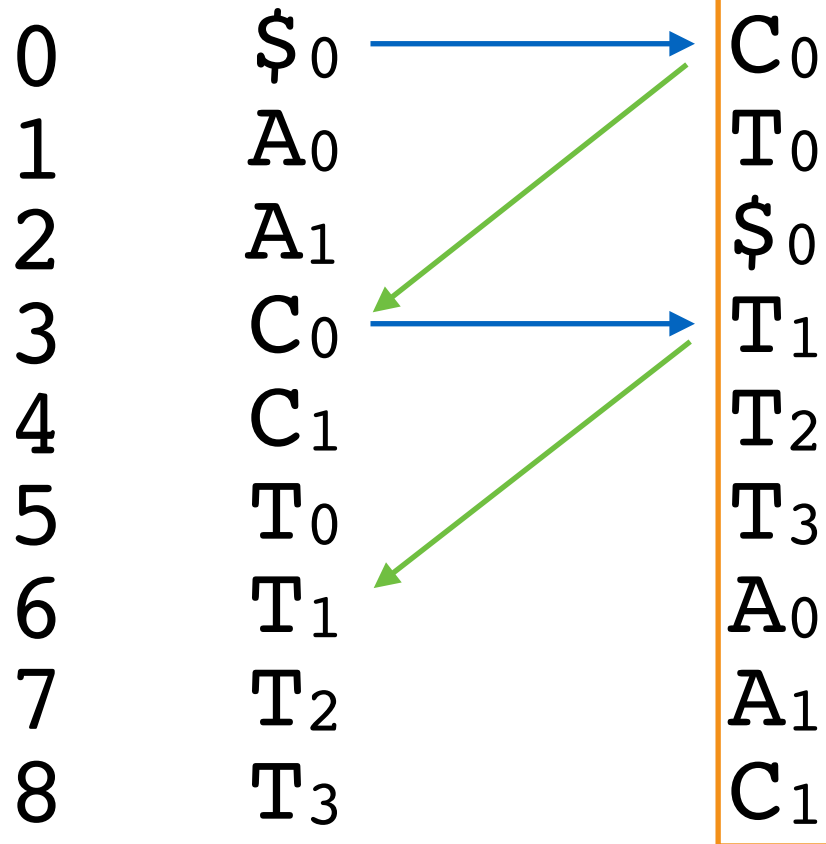
0	\$ <sub>0</sub>	→	C <sub>0</sub>
1	A <sub>0</sub>		T <sub>0</sub>
2	A <sub>1</sub>		\$ <sub>0</sub>
3	C <sub>0</sub>	→	T <sub>1</sub>
4	C <sub>1</sub>		T <sub>2</sub>
5	T <sub>0</sub>		T <sub>3</sub>
6	T <sub>1</sub>		A <sub>0</sub>
7	T <sub>2</sub>		A <sub>1</sub>
8	T <sub>3</sub>		C <sub>1</sub>

T<sub>1</sub> C<sub>0</sub> \$<sub>0</sub>

Secuencia original

# Revirtiendo la transformación BWT

Renglón

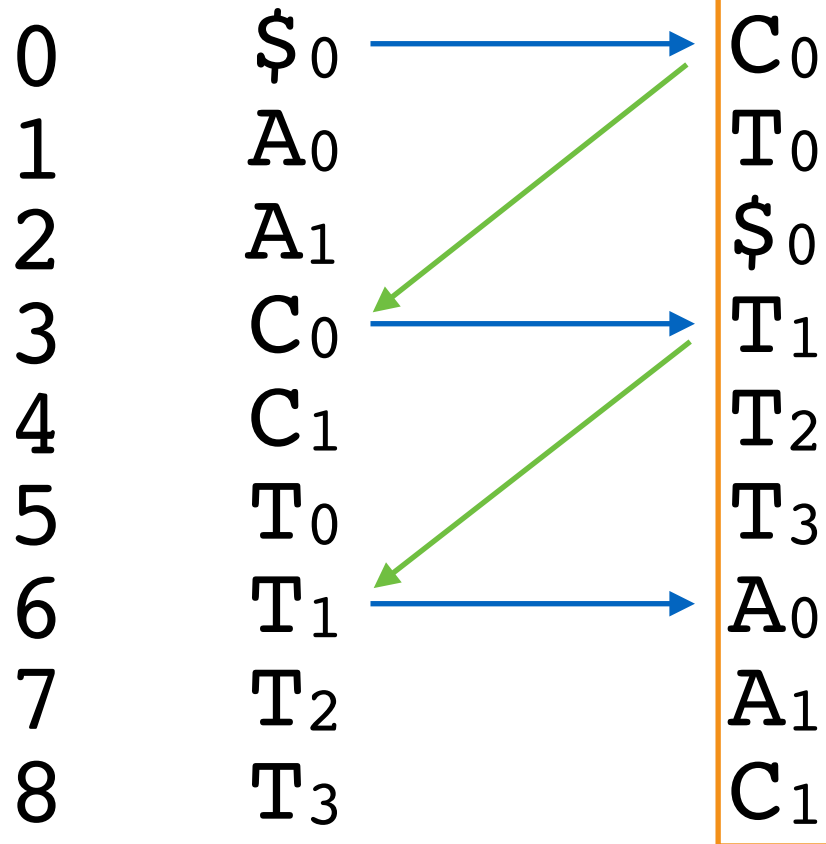


T<sub>1</sub> C<sub>0</sub> \$<sub>0</sub>

Secuencia original

# Revirtiendo la transformación BWT

Renglón

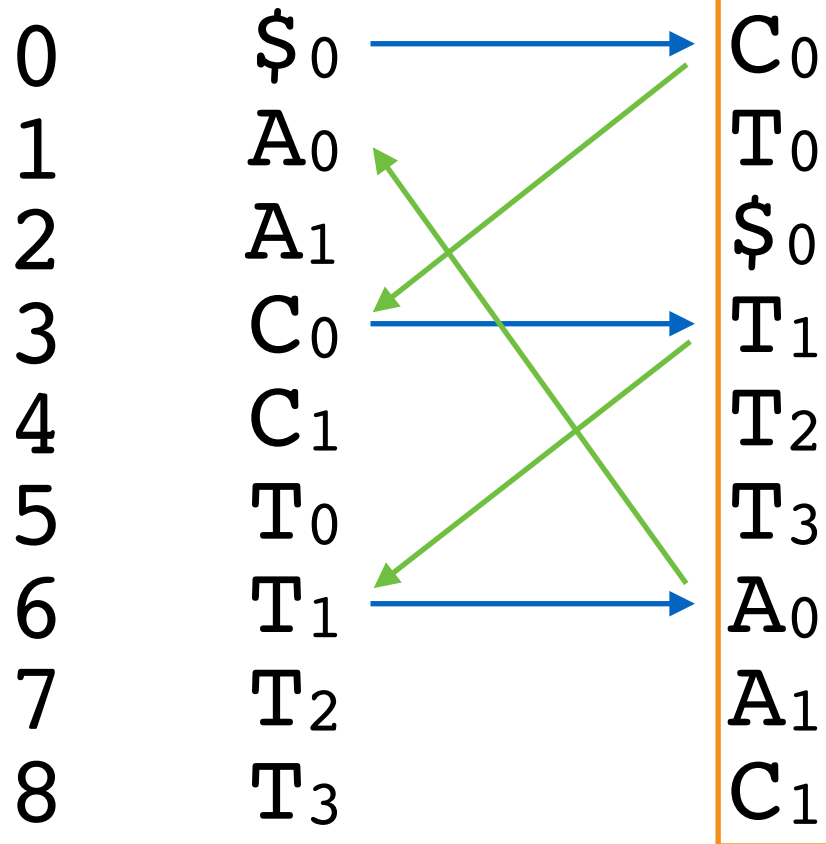


A<sub>0</sub> T<sub>1</sub> C<sub>0</sub> \$<sub>0</sub>  
Secuencia original



# Revirtiendo la transformación BWT

Renglón

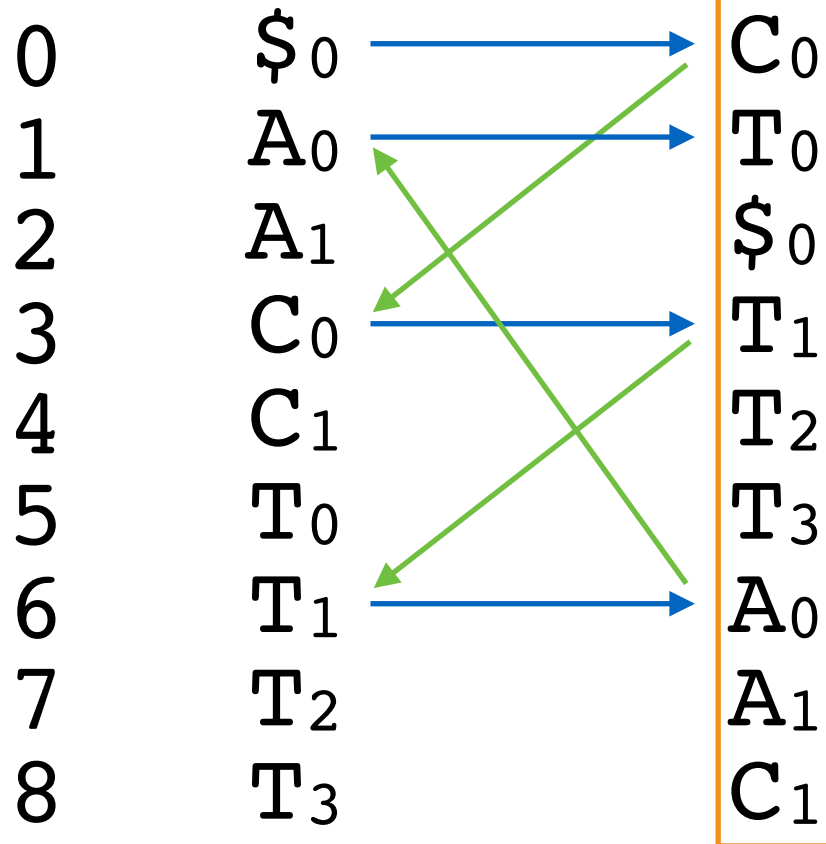


A<sub>0</sub> T<sub>1</sub> C<sub>0</sub> \$<sub>0</sub>

Secuencia original

# Revirtiendo la transformación BWT

Renglón

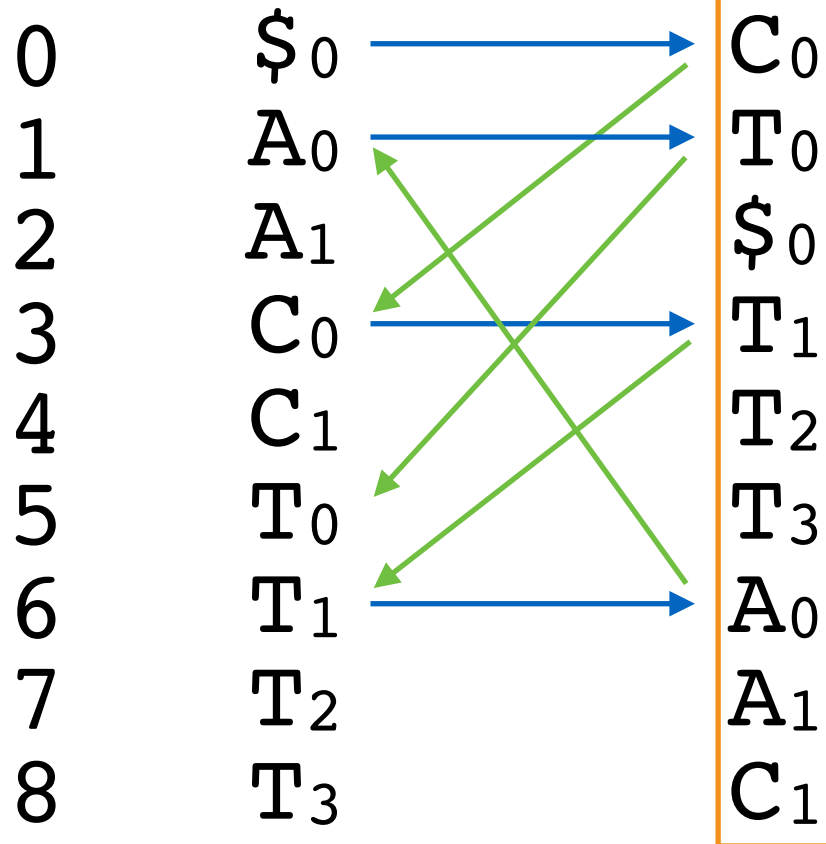


T<sub>0</sub> A<sub>0</sub> T<sub>1</sub> C<sub>0</sub> \$<sub>0</sub>

Secuencia original

# Revirtiendo la transformación BWT

Renglón

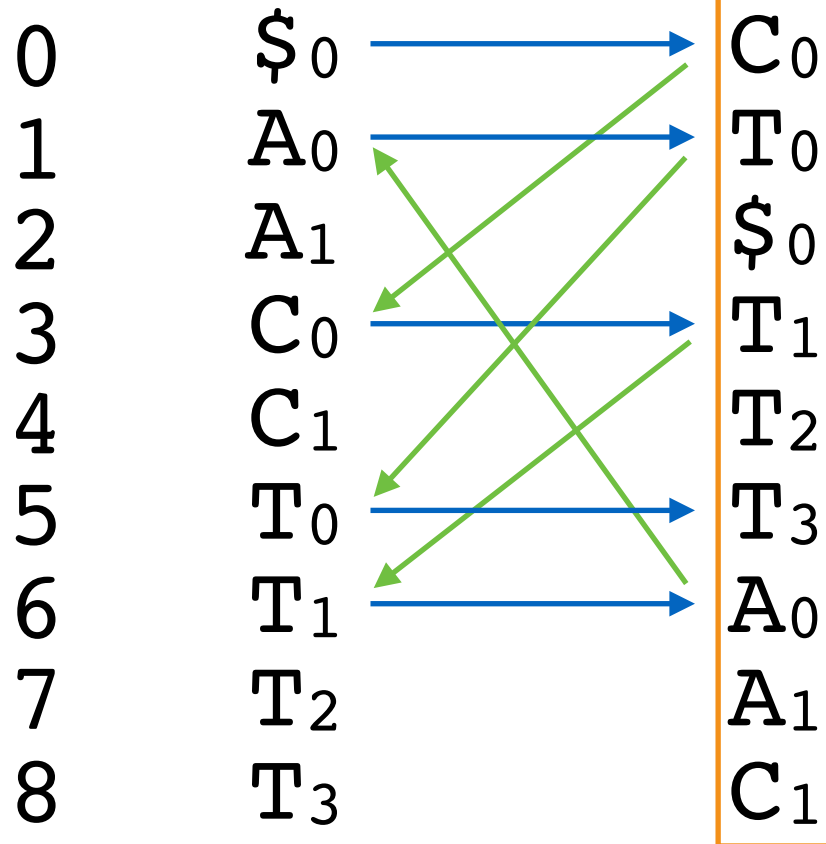


T<sub>0</sub> A<sub>0</sub> T<sub>1</sub> C<sub>0</sub> \$<sub>0</sub>

Secuencia original

# Revirtiendo la transformación BWT

Renglón

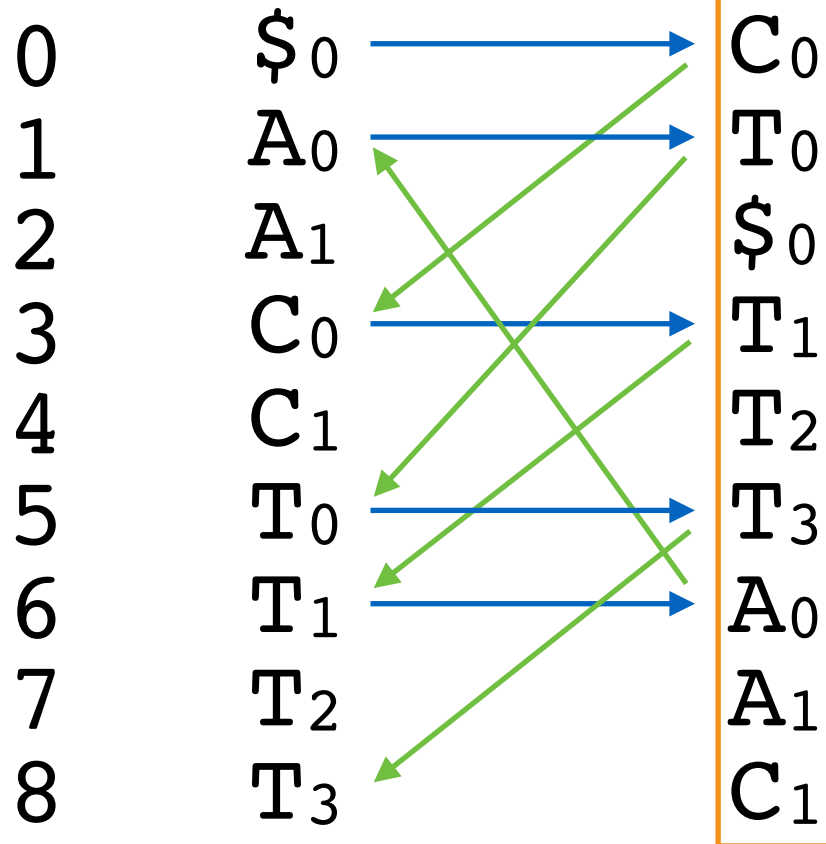


T<sub>3</sub> T<sub>0</sub> A<sub>0</sub> T<sub>1</sub> C<sub>0</sub> \$<sub>0</sub>

Secuencia original

# Revirtiendo la transformación BWT

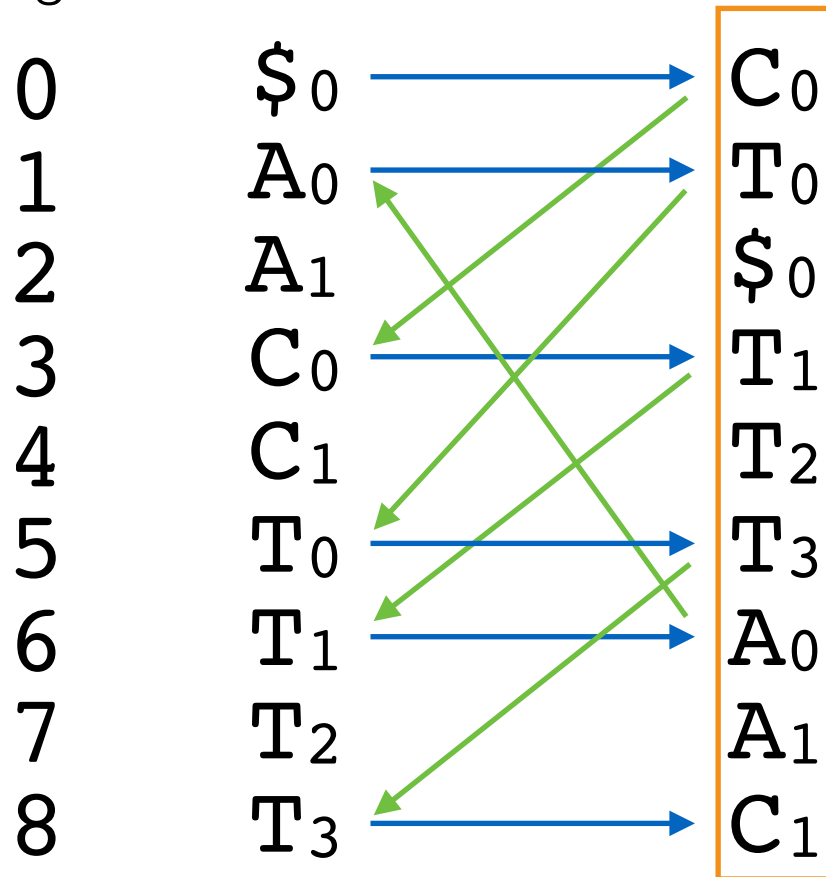
Renglón



T<sub>3</sub> T<sub>0</sub> A<sub>0</sub> T<sub>1</sub> C<sub>0</sub> \$<sub>0</sub>  
Secuencia original

# Revirtiendo la transformación BWT

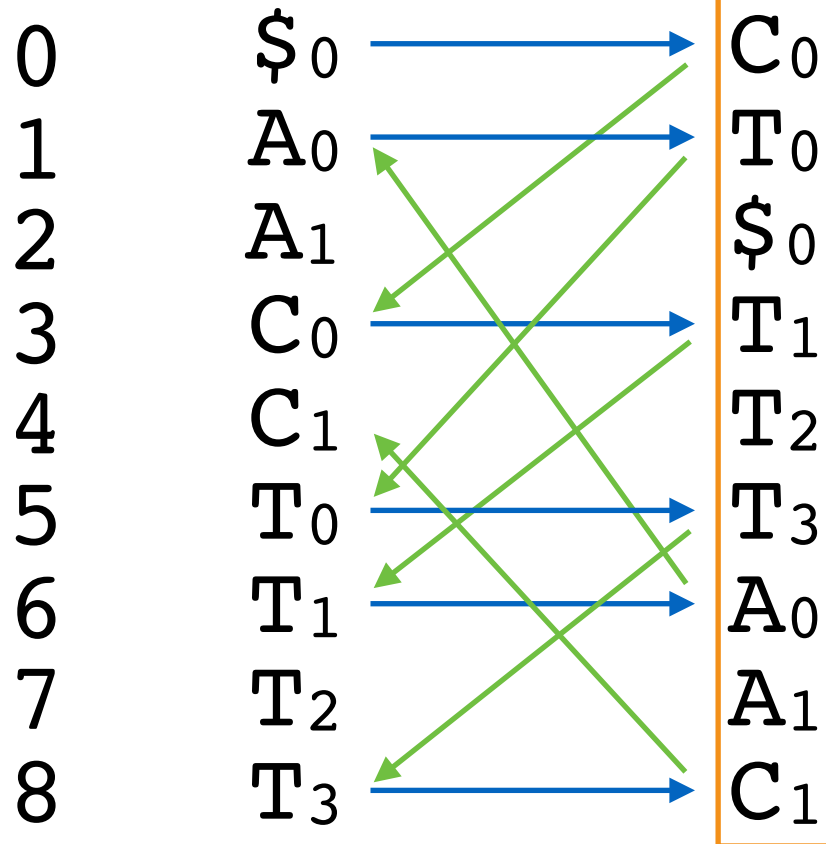
Renglón



C<sub>1</sub> T<sub>3</sub> T<sub>0</sub> A<sub>0</sub> T<sub>1</sub> C<sub>0</sub> \$<sub>0</sub>  
Secuencia original

# Revirtiendo la transformación BWT

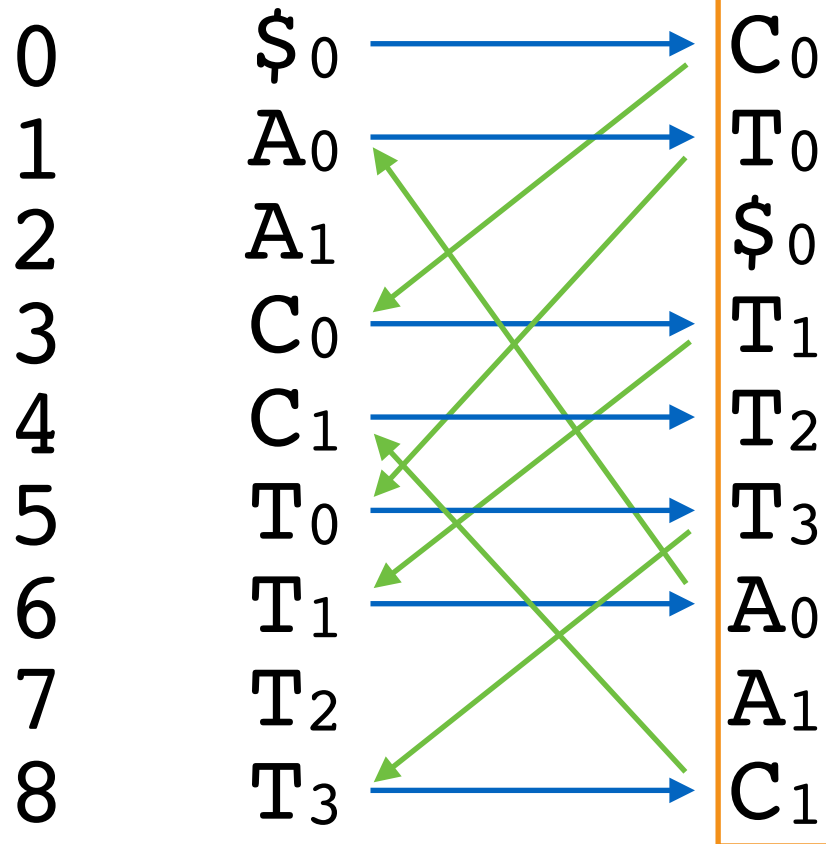
Renglón



C<sub>1</sub> T<sub>3</sub> T<sub>0</sub> A<sub>0</sub> T<sub>1</sub> C<sub>0</sub> \$<sub>0</sub>  
Secuencia original

# Revirtiendo la transformación BWT

Renglón

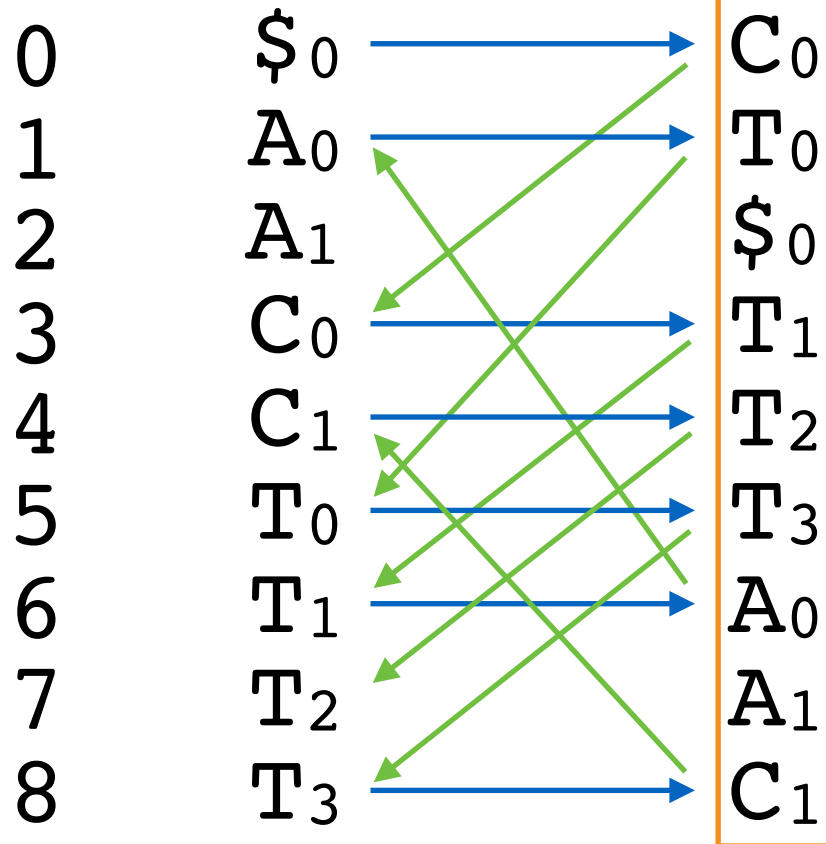


T<sub>2</sub> C<sub>1</sub> T<sub>3</sub> T<sub>0</sub> A<sub>0</sub> T<sub>1</sub> C<sub>0</sub> \$<sub>0</sub>  
Secuencia original



# Revirtiendo la transformación BWT

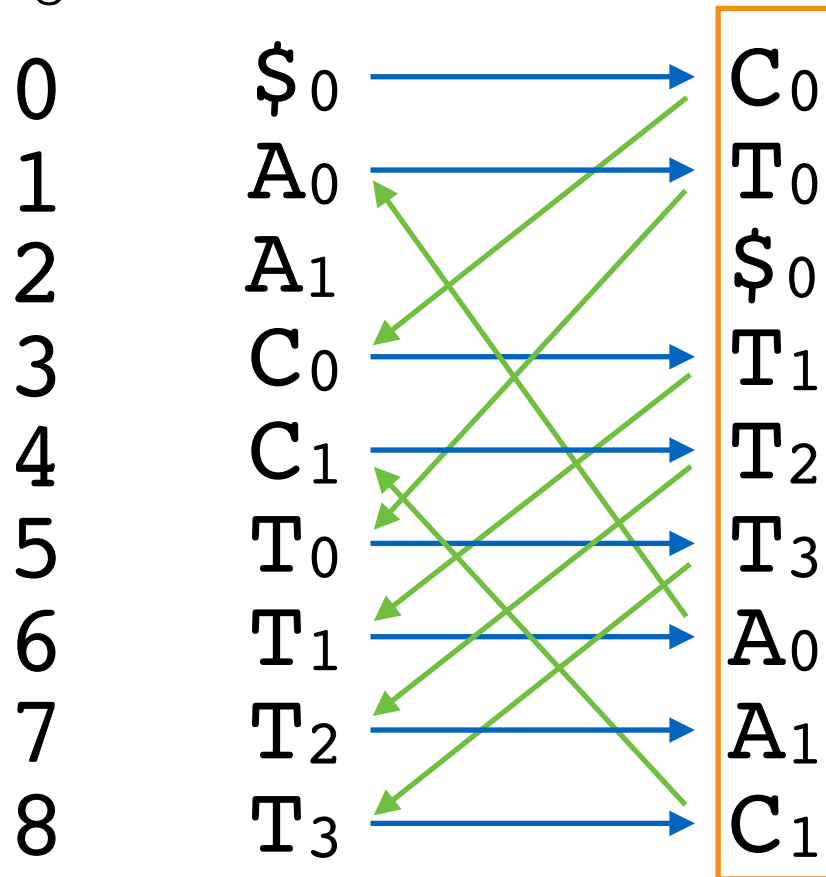
Renglón



$T_2 C_1 T_3 T_0 A_0 T_1 C_0 \$0$   
Secuencia original

# Revirtiendo la transformación BWT

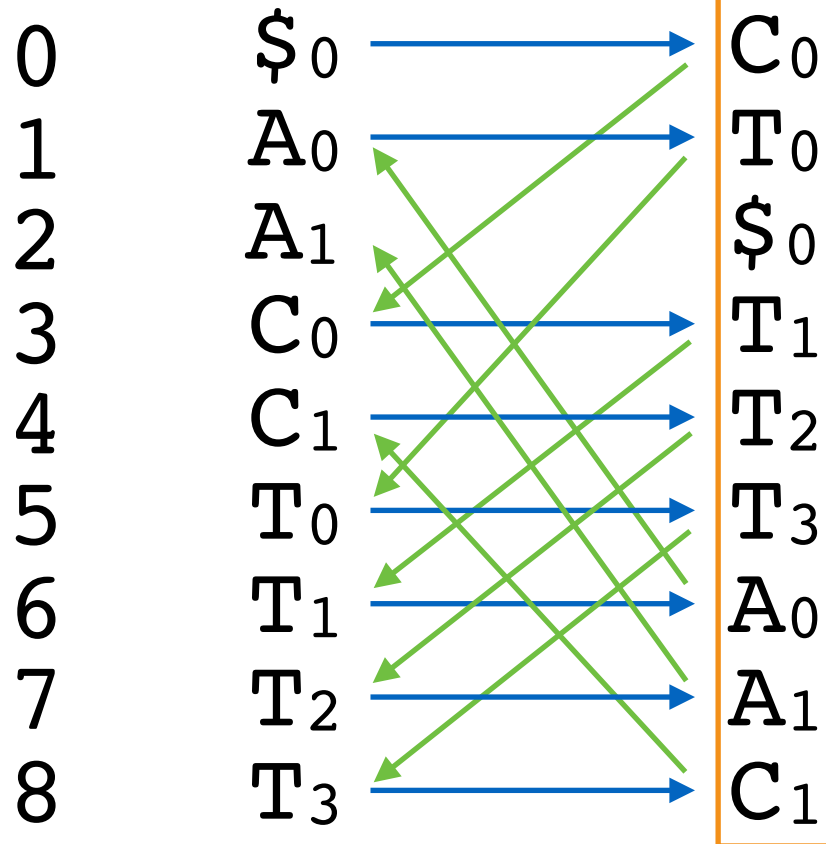
Renglón



A<sub>1</sub> T<sub>2</sub> C<sub>1</sub> T<sub>3</sub> T<sub>0</sub> A<sub>0</sub> T<sub>1</sub> C<sub>0</sub> \$<sub>0</sub>  
Secuencia original

# Revirtiendo la transformación BWT

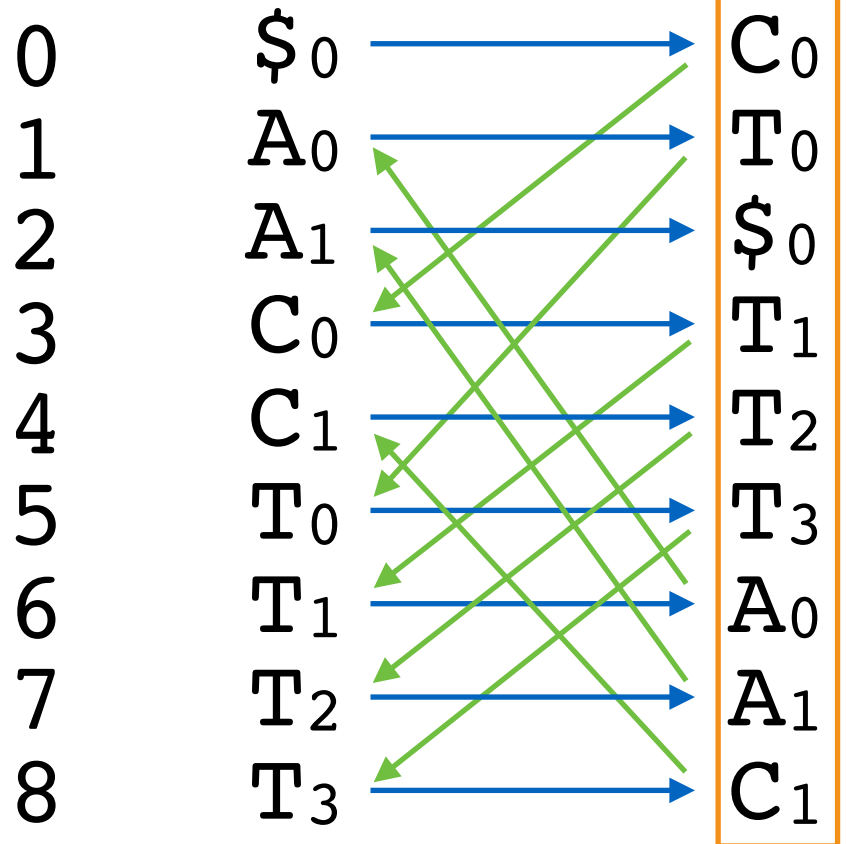
Renglón



A<sub>1</sub> T<sub>2</sub> C<sub>1</sub> T<sub>3</sub> T<sub>0</sub> A<sub>0</sub> T<sub>1</sub> C<sub>0</sub> \$<sub>0</sub>  
Secuencia original

# Revirtiendo la transformación BWT

Renglón



A<sub>1</sub> T<sub>2</sub> C<sub>1</sub> T<sub>3</sub> T<sub>0</sub> A<sub>0</sub> T<sub>1</sub> C<sub>0</sub> \$<sub>0</sub>  
Secuencia original

# Usando BWT para mapear

Renglón

0	\$ <sub>0</sub>	C <sub>0</sub>
1	A <sub>0</sub>	T <sub>0</sub>
2	A <sub>1</sub>	\$ <sub>0</sub>
3	C <sub>0</sub>	T <sub>1</sub>
4	C <sub>1</sub>	T <sub>2</sub>
5	T <sub>0</sub>	T <sub>3</sub>
6	T <sub>1</sub>	A <sub>0</sub>
7	T <sub>2</sub>	A <sub>1</sub>
8	T <sub>3</sub>	C <sub>1</sub>

# Usando BWT para mapear

Renglón

0	\$ <sub>0</sub>
1	A <sub>0</sub>
2	A <sub>1</sub>
3	C <sub>0</sub>
4	C <sub>1</sub>
5	T <sub>0</sub>
6	T <sub>1</sub>
7	T <sub>2</sub>
8	T <sub>3</sub>

**BWT**

C <sub>0</sub>
T <sub>0</sub>
\$ <sub>0</sub>
T <sub>1</sub>
T <sub>2</sub>
T <sub>3</sub>
A <sub>0</sub>
A <sub>1</sub>
C <sub>1</sub>

Lectura: TTATC

# Usando BWT para mapear

Renglón

0	\$ <sub>0</sub>
1	A <sub>0</sub>
2	A <sub>1</sub>
3	C <sub>0</sub>
4	C <sub>1</sub>
5	T <sub>0</sub>
6	T <sub>1</sub>
7	T <sub>2</sub>
8	T <sub>3</sub>

BWT

C <sub>0</sub>
T <sub>0</sub>
\$ <sub>0</sub>
T <sub>1</sub>
T <sub>2</sub>
T <sub>3</sub>
A <sub>0</sub>
A <sub>1</sub>
C <sub>1</sub>

# Usando BWT para mapear

Renglón

0	\$ <sub>0</sub>	C <sub>0</sub>
1	A <sub>0</sub>	T <sub>0</sub>
2	A <sub>1</sub>	\$ <sub>0</sub>
3	C <sub>0</sub>	T <sub>1</sub>
4	C <sub>1</sub>	T <sub>2</sub>
5	T <sub>0</sub>	T <sub>3</sub>
6	T <sub>1</sub>	A <sub>0</sub>
7	T <sub>2</sub>	A <sub>1</sub>
8	T <sub>3</sub>	C <sub>1</sub>

Lectura: TTATC



# Usando BWT para mapear

Renglón

0	\$ <sub>0</sub>	<b>C<sub>0</sub></b>
1	A <sub>0</sub>	T <sub>0</sub>
2	A <sub>1</sub>	\$ <sub>0</sub>
3	<b>C<sub>0</sub></b>	<b>T<sub>1</sub></b>
4	<b>C<sub>1</sub></b>	<b>T<sub>2</sub></b>
5	T <sub>0</sub>	T <sub>3</sub>
6	T <sub>1</sub>	A <sub>0</sub>
7	T <sub>2</sub>	A <sub>1</sub>
8	T <sub>3</sub>	<b>C<sub>1</sub></b>

Lectura: TTAT**C**

# Usando BWT para mapear

Renglón

0	\$ <sub>0</sub>		<b>C<sub>0</sub></b>
1	A <sub>0</sub>		T <sub>0</sub>
2	A <sub>1</sub>		\$ <sub>0</sub>
3	<b>C<sub>0</sub></b>	→	<b>T<sub>1</sub></b>
4	<b>C<sub>1</sub></b>		<b>T<sub>2</sub></b>
5	T <sub>0</sub>		T <sub>3</sub>
6	T <sub>1</sub>		A <sub>0</sub>
7	T <sub>2</sub>		A <sub>1</sub>
8	T <sub>3</sub>		<b>C<sub>1</sub></b>

Lectura: TTAT**C**

# Usando BWT para mapear

Renglón

0	\$ <sub>0</sub>		<b>C<sub>0</sub></b>
1	A <sub>0</sub>		T <sub>0</sub>
2	A <sub>1</sub>		\$ <sub>0</sub>
3	<b>C<sub>0</sub></b>	→	<b>T<sub>1</sub></b>
4	<b>C<sub>1</sub></b>	→	<b>T<sub>2</sub></b>
5	T <sub>0</sub>		T <sub>3</sub>
6	T <sub>1</sub>		A <sub>0</sub>
7	T <sub>2</sub>		A <sub>1</sub>
8	T <sub>3</sub>		<b>C<sub>1</sub></b>

Lectura: TTAT**C**

# Usando BWT para mapear

Renglón

0	\$ <sub>0</sub>
1	A <sub>0</sub>
2	A <sub>1</sub>
3	C <sub>0</sub>
4	C <sub>1</sub>
5	T <sub>0</sub>
6	T <sub>1</sub>
7	T <sub>2</sub>
8	T <sub>3</sub>

BWT

C <sub>0</sub>
T <sub>0</sub>
\$ <sub>0</sub>
T <sub>1</sub>
T <sub>2</sub>
T <sub>3</sub>
A <sub>0</sub>
A <sub>1</sub>
C <sub>1</sub>

# Usando BWT para mapear

Renglón

0	\$ <sub>0</sub>
1	A <sub>0</sub>
2	A <sub>1</sub>
3	C <sub>0</sub>
4	C <sub>1</sub>
5	T <sub>0</sub>
6	T <sub>1</sub>
7	T <sub>2</sub>
8	T <sub>3</sub>

BWT

C <sub>0</sub>
T <sub>0</sub>
\$ <sub>0</sub>
T <sub>1</sub>
T <sub>2</sub>
T <sub>3</sub>
A <sub>0</sub>
A <sub>1</sub>
C <sub>1</sub>

Lectura: TTATC

# Usando BWT para mapear

Renglón

0	\$ <sub>0</sub>
1	A <sub>0</sub>
2	A <sub>1</sub>
3	C <sub>0</sub>
4	C <sub>1</sub>
5	T <sub>0</sub>
6	T <sub>1</sub>
7	T <sub>2</sub>
8	T <sub>3</sub>

BWT

C <sub>0</sub>
T <sub>0</sub>
\$ <sub>0</sub>
T <sub>1</sub>
T <sub>2</sub>
T <sub>3</sub>
A <sub>0</sub>
A <sub>1</sub>
C <sub>1</sub>

Lectura: TTATC

# Usando BWT para mapear

Renglón

0	\$ <sub>0</sub>
1	A <sub>0</sub>
2	A <sub>1</sub>
3	C <sub>0</sub>
4	C <sub>1</sub>
5	T <sub>0</sub>
6	T <sub>1</sub>
7	T <sub>2</sub>
8	T <sub>3</sub>

BWT

C <sub>0</sub>
T <sub>0</sub>
\$ <sub>0</sub>
T <sub>1</sub>
T <sub>2</sub>
T <sub>3</sub>
A <sub>0</sub>
A <sub>1</sub>
C <sub>1</sub>

Lectura: TTA**T**C

# Usando BWT para mapear

Renglón

0	\$ <sub>0</sub>
1	A <sub>0</sub>
2	A <sub>1</sub>
3	C <sub>0</sub>
4	C <sub>1</sub>
5	T <sub>0</sub>
6	T <sub>1</sub>
7	T <sub>2</sub>
8	T <sub>3</sub>

BWT

C <sub>0</sub>
T <sub>0</sub>
\$ <sub>0</sub>
T <sub>1</sub>
T <sub>2</sub>
T <sub>3</sub>
A <sub>0</sub>
A <sub>1</sub>
C <sub>1</sub>

Lectura: TTATC



# Usando BWT para mapear

Renglón

0	\$ <sub>0</sub>		C <sub>0</sub>
1	A <sub>0</sub>		T <sub>0</sub>
2	A <sub>1</sub>		\$ <sub>0</sub>
3	C <sub>0</sub>		T <sub>1</sub>
4	C <sub>1</sub>		T <sub>2</sub>
5	T <sub>0</sub>		T <sub>3</sub>
6	T <sub>1</sub>	→	A <sub>0</sub>
7	T <sub>2</sub>	→	A <sub>1</sub>
8	T <sub>3</sub>		C <sub>1</sub>

BWT

Lectura: TTATC

# Usando BWT para mapear

Renglón

0	\$ <sub>0</sub>
1	A <sub>0</sub>
2	A <sub>1</sub>
3	C <sub>0</sub>
4	C <sub>1</sub>
5	T <sub>0</sub>
6	T <sub>1</sub>
7	T <sub>2</sub>
8	T <sub>3</sub>

BWT

C <sub>0</sub>
T <sub>0</sub>
\$ <sub>0</sub>
T <sub>1</sub>
T <sub>2</sub>
T <sub>3</sub>
A <sub>0</sub>
A <sub>1</sub>
C <sub>1</sub>

# Usando BWT para mapear

Renglón

0	\$ <sub>0</sub>
1	A <sub>0</sub>
2	A <sub>1</sub>
3	C <sub>0</sub>
4	C <sub>1</sub>
5	T <sub>0</sub>
6	T <sub>1</sub>
7	T <sub>2</sub>
8	T <sub>3</sub>

BWT

C <sub>0</sub>
T <sub>0</sub>
\$ <sub>0</sub>
T <sub>1</sub>
T <sub>2</sub>
T <sub>3</sub>
A <sub>0</sub>
A <sub>1</sub>
C <sub>1</sub>

Lectura: TTATC

# Usando BWT para mapear

Renglón

0	\$ <sub>0</sub>
1	A <sub>0</sub>
2	A <sub>1</sub>
3	C <sub>0</sub>
4	C <sub>1</sub>
5	T <sub>0</sub>
6	T <sub>1</sub>
7	T <sub>2</sub>
8	T <sub>3</sub>

BWT

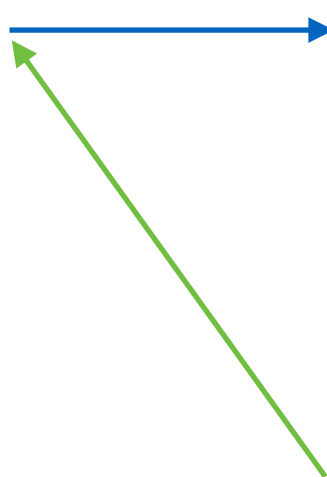
C <sub>0</sub>
T <sub>0</sub>
\$ <sub>0</sub>
T <sub>1</sub>
T <sub>2</sub>
T <sub>3</sub>
A <sub>0</sub>
A <sub>1</sub>
C <sub>1</sub>

Lectura: TTATC

# Usando BWT para mapear

Renglón

		BWT
0	\$ <sub>0</sub>	C <sub>0</sub>
1	A <sub>0</sub>	T <sub>0</sub>
2	A <sub>1</sub>	\$ <sub>0</sub>
3	C <sub>0</sub>	T <sub>1</sub>
4	C <sub>1</sub>	T <sub>2</sub>
5	T <sub>0</sub>	T <sub>3</sub>
6	T <sub>1</sub>	A <sub>0</sub>
7	T <sub>2</sub>	A <sub>1</sub>
8	T <sub>3</sub>	C <sub>1</sub>



Lectura: TTATC

# Usando BWT para mapear

Renglón

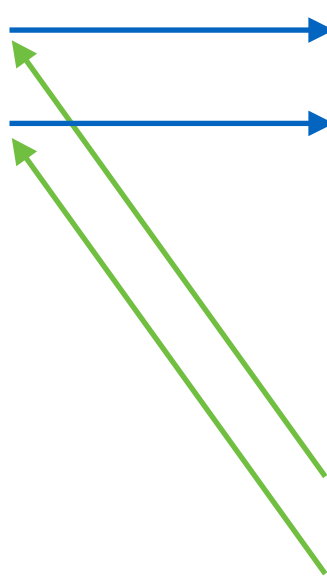
0	\$ <sub>0</sub>		C <sub>0</sub>
1	A <sub>0</sub>	→	T <sub>0</sub>
2	A <sub>1</sub>	↖	\$ <sub>0</sub>
3	C <sub>0</sub>	↖	T <sub>1</sub>
4	C <sub>1</sub>		T <sub>2</sub>
5	T <sub>0</sub>		T <sub>3</sub>
6	T <sub>1</sub>		A <sub>0</sub>
7	T <sub>2</sub>		A <sub>1</sub>
8	T <sub>3</sub>		C <sub>1</sub>

Lectura: TT**A**TC

# Usando BWT para mapear

Renglón

		BWT
0	\$ <sub>0</sub>	C <sub>0</sub>
1	A <sub>0</sub>	T <sub>0</sub>
2	A <sub>1</sub>	\$ <sub>0</sub>
3	C <sub>0</sub>	T <sub>1</sub>
4	C <sub>1</sub>	T <sub>2</sub>
5	T <sub>0</sub>	T <sub>3</sub>
6	T <sub>1</sub>	A <sub>0</sub>
7	T <sub>2</sub>	A <sub>1</sub>
8	T <sub>3</sub>	C <sub>1</sub>



Lectura: TT**A**TC

# Usando BWT para mapear

Renglón

0	\$ <sub>0</sub>
1	A <sub>0</sub>
2	A <sub>1</sub>
3	C <sub>0</sub>
4	C <sub>1</sub>
5	T <sub>0</sub>
6	T <sub>1</sub>
7	T <sub>2</sub>
8	T <sub>3</sub>

BWT

C <sub>0</sub>
T <sub>0</sub>
\$ <sub>0</sub>
T <sub>1</sub>
T <sub>2</sub>
T <sub>3</sub>
A <sub>0</sub>
A <sub>1</sub>
C <sub>1</sub>



# Usando BWT para mapear

Renglón

0	\$ <sub>0</sub>
1	A <sub>0</sub>
2	A <sub>1</sub>
3	C <sub>0</sub>
4	C <sub>1</sub>
5	T <sub>0</sub>
6	T <sub>1</sub>
7	T <sub>2</sub>
8	T <sub>3</sub>

BWT

C <sub>0</sub>
T <sub>0</sub>
\$ <sub>0</sub>
T <sub>1</sub>
T <sub>2</sub>
T <sub>3</sub>
A <sub>0</sub>
A <sub>1</sub>
C <sub>1</sub>

Lectura: **T**TATC

# Usando BWT para mapear

Renglón

0	\$ <sub>0</sub>
1	A <sub>0</sub>
2	A <sub>1</sub>
3	C <sub>0</sub>
4	C <sub>1</sub>
5	T <sub>0</sub>
6	T <sub>1</sub>
7	T <sub>2</sub>
8	T <sub>3</sub>

BWT

C <sub>0</sub>
T <sub>0</sub>
\$ <sub>0</sub>
T <sub>1</sub>
T <sub>2</sub>
T <sub>3</sub>
A <sub>0</sub>
A <sub>1</sub>
C <sub>1</sub>

Lectura: **T**TATC

# Usando BWT para mapear

Renglón

0	\$ <sub>0</sub>
1	A <sub>0</sub>
2	A <sub>1</sub>
3	C <sub>0</sub>
4	C <sub>1</sub>
5	T <sub>0</sub>
6	T <sub>1</sub>
7	T <sub>2</sub>
8	T <sub>3</sub>

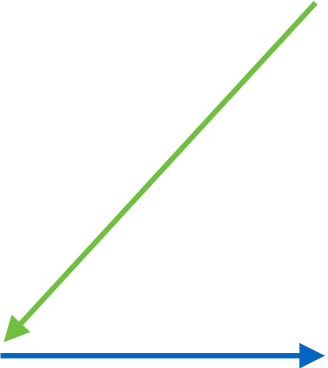
BWT

C <sub>0</sub>
T <sub>0</sub>
\$ <sub>0</sub>
T <sub>1</sub>
T <sub>2</sub>
T <sub>3</sub>
A <sub>0</sub>
A <sub>1</sub>
C <sub>1</sub>

Lectura: **T**TATC

# Usando BWT para mapear

Renglón		BWT
0	\$ <sub>0</sub>	C <sub>0</sub>
1	A <sub>0</sub>	T <sub>0</sub>
2	A <sub>1</sub>	\$ <sub>0</sub>
3	C <sub>0</sub>	T <sub>1</sub>
4	C <sub>1</sub>	T <sub>2</sub>
5	T <sub>0</sub>	T <sub>3</sub>
6	T <sub>1</sub>	A <sub>0</sub>
7	T <sub>2</sub>	A <sub>1</sub>
8	T <sub>3</sub>	C <sub>1</sub>



Lectura: **T**TATC

La lectura  
mapea a nuestra  
secuencia pero ...  
¿dónde está en el  
genoma?

# Usando BWT para mapear

Renglón                      Suffix array

0	\$ <sub>0</sub>	C <sub>0</sub>	8
1	A <sub>0</sub>	T <sub>0</sub>	5
2	A <sub>1</sub>	\$ <sub>0</sub>	0
3	C <sub>0</sub>	T <sub>1</sub>	7
4	C <sub>1</sub>	T <sub>2</sub>	2
5	T <sub>0</sub>	T <sub>3</sub>	4
6	T <sub>1</sub>	A <sub>0</sub>	6
7	T <sub>2</sub>	A <sub>1</sub>	1
8	T <sub>3</sub>	C <sub>1</sub>	3

**BWT**

Un sufijo podría indicarnos donde se encuentra en la secuencia original. Usa mucho espacio si tenemos millones de posiciones

# Usando BWT para mapear

Renglón                      Suffix array

0	\$ <sub>0</sub>	C <sub>0</sub>	8
1	A <sub>0</sub>	T <sub>0</sub>	5
2	A <sub>1</sub>	\$ <sub>0</sub>	0
3	C <sub>0</sub>	T <sub>1</sub>	7
4	C <sub>1</sub>	T <sub>2</sub>	2
5	T <sub>0</sub>	T <sub>3</sub>	4
6	T <sub>1</sub>	A <sub>0</sub>	6
7	T <sub>2</sub>	A <sub>1</sub>	1
8	T <sub>3</sub>	C <sub>1</sub>	3

**BWT**

Lectura: **T**TATC

Un sufijo podría indicarnos donde se encuentra en la secuencia original. Usa mucho espacio si tenemos millones de posiciones

# Usando BWT para mapear

Renglón

Suffix array

0	\$ <sub>0</sub>	C <sub>0</sub>	8
1	A <sub>0</sub>	T <sub>0</sub>	5
2	A <sub>1</sub>	\$ <sub>0</sub>	0
3	C <sub>0</sub>	T <sub>1</sub>	7
4	C <sub>1</sub>	T <sub>2</sub>	2
5	T <sub>0</sub>	T <sub>3</sub>	4
6	T <sub>1</sub>	A <sub>0</sub>	6
7	T <sub>2</sub>	A <sub>1</sub>	1
8	T <sub>3</sub>	C <sub>1</sub>	3

**BWT**

Lectura: **T**TATC

A<sub>1</sub> T<sub>2</sub> C<sub>1</sub> **T**<sub>3</sub> T<sub>0</sub> A<sub>0</sub> T<sub>1</sub> C<sub>0</sub> \$<sub>0</sub>

Un sufijo podría indicarnos donde se encuentra en la secuencia original. Usa mucho espacio si tenemos millones de posiciones

# Full-text Minute-size (FM) index

Renglón

0	\$ <sub>0</sub>
1	A <sub>0</sub>
2	A <sub>1</sub>
3	C <sub>0</sub>
4	C <sub>1</sub>
5	T <sub>0</sub>
6	T <sub>1</sub>
7	T <sub>2</sub>
8	T <sub>3</sub>

Checkpoints

C <sub>0</sub>
T <sub>0</sub>
\$ <sub>0</sub>
T <sub>1</sub>
T <sub>2</sub>
T <sub>3</sub>
A <sub>0</sub>
A <sub>1</sub>
C <sub>1</sub>

[A:0,T:1,C:1,G:0]

[A:2,T:4,C:1,G:0]

**BWT**

Lo que hacemos es utilizar “checkpoints” a lo largo del BWT para indicarnos la posición. Cuando encontramos un match, buscamos el “checkpoint” más cercano para identificar su posición en la referencia (genoma o transcriptoma).

A esto se le conoce como FM index y es muy pequeño.



# Full-text Minute-size (FM) index

Renglón

0	\$ <sub>0</sub>
1	A <sub>0</sub>
2	A <sub>1</sub>
3	C <sub>0</sub>
4	C <sub>1</sub>
5	T <sub>0</sub>
6	T <sub>1</sub>
7	T <sub>2</sub>
8	T <sub>3</sub>

Checkpoints

C <sub>0</sub>
T <sub>0</sub>
\$ <sub>0</sub>
T <sub>1</sub>
T <sub>2</sub>
T <sub>3</sub>
A <sub>0</sub>
A <sub>1</sub>
C <sub>1</sub>

[A:0,T:1,C:1,G:0]

[A:2,T:4,C:1,G:0]

**BWT**

Lectura: **T**TATC

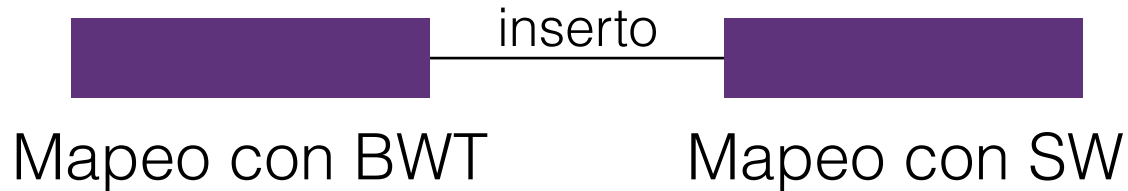
Lo que hacemos es utilizar “checkpoints” a lo largo del BWT para indicarnos la posición. Cuando encontramos un match, buscamos el “checkpoint” más cercano para identificar su posición en la referencia (genoma o transcriptoma).

A esto se le conoce como FM index y es muy pequeño.

# Errores o Mismatches

- De no identificarse ningún alineamiento perfecto de la lectura a la secuencia de referencia se toman los alineamientos parciales y se permuta el nucleótido candidato a mismatch (A,T, C,G) y se trata de seguir extendiendo el sitio con similitud a la lectura de interés.
- A esto se le conoce como “backtracking” y generalmente se limita a un número arbitrario de ciclos para evitar incrementar demasiado el tiempo de alineamiento.
- Se hace más backtracking en nucleótidos con baja calidad.
- Dado que el tiempo de cálculo es lineal, no es tan tardado tratar de hacer esto para buscar el lugar de origen de lecturas con errores.

# Lecturas en pares (paired-end)



- Muchas veces una sola lectura se encuentra usando alineamiento via BWT. Dado que sabemos el tamaño aproximado del inserto algunos algoritmos utilizan alineamientos Smith-Waterman (SW) para encontrar su par en la región vecina.

# Programas para alinear lecturas a una referencia

- bowtie2 - **TopHat** (<https://ccb.jhu.edu/software/tophat/index.shtml>)
- bowtie (<http://bowtie-bio.sourceforge.net/index.shtml>)
- HISAT2 (<https://ccb.jhu.edu/software/hisat2/manual.shtml>)
- STAR (<https://github.com/alexdobin/STAR>)

# Programas para alinear **transcritos** a una referencia

- GMAP (<http://research-pub.gene.com/gmap/>)
- Blat (<https://genome.ucsc.edu/goldenpath/help/blatSpec.html>)
- Exonerate (<http://www.animalgenome.org/bioinfo/resources/manuals/exonerate/beginner.html>)

# Práctica - alineando lecturas usando Bowtie

[https://liz-fernandez.github.io/PBI\\_transcriptomics/](https://liz-fernandez.github.io/PBI_transcriptomics/)