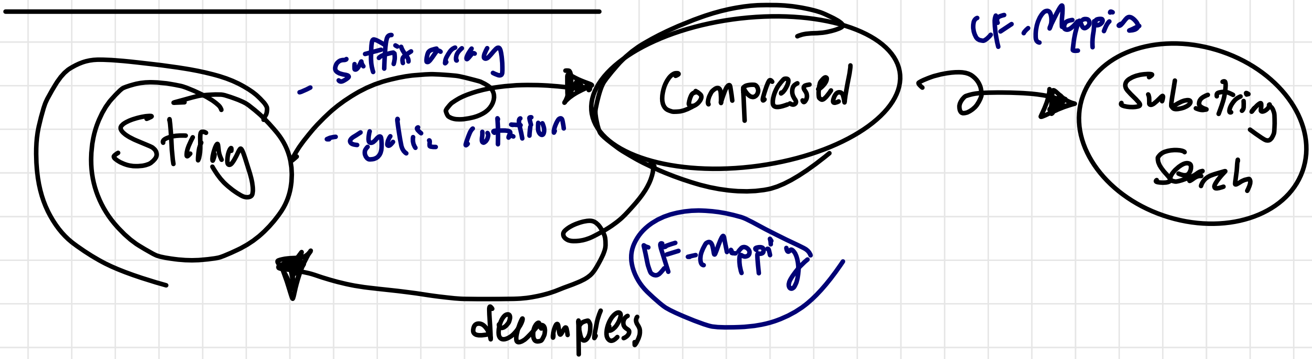
BWT - Burrows- Wheeler's Transform



BWT

[1D-1]

" cyclic rotation "

KOKOKRUNCH$

Suffix

```
KOKOKRUNCH$
OKOKRUNCH$K
KOKRUNCH$KO
OKRUNCH$KOK
KRUNCH$KOKO
RUNCH$KOKOK
UNCH$KOKOKR
NCH$KOKOKRU
CH$KOKOKRUN
H$KOKOKRUNC
$KOKOKRUNCH
```

— sort →

[1D] [1D-1]

```
$KOKOKRUNCH
CH$KOKOKRUN
H$KOKOKRUNC
KOKOKRUNCH$
KOKRUNCH$KO
KRUNCH$KOKO
NCH$KOKOKRU
OKOKRUNCH$K
OKRUNCH$KOK
RUNCH$KOKOK
UNCH$KOKOKR
```

HNC$OOUKKKR

BWT

ID-1

KNICKKNACKATORY$

K_1
N_1
I
C_1
K_2
K_3
N_2
A_1
C_2
K_4
A_2
T
O
R
Y
$

Suffix array

$
A_1
A_2
C_2
C_1
I
K_4
K_2
K_3
K_1
N_2
N_1
O
R
T
Y

ID-1

Y
N
K
A
I
N
C
C
K
$
K
K
T
O
A
R

BWT

YNKAINCCK$KKTOAR

Y N K A I N C C K $ K K T O A R

decompression

count sort

first	last
$	Y
A	N_1
A	K_1
C	A_1
C	I
I	N_2
K	C_1
K	C_2
K	K_2
K	$
N	K_3
N	K_4
O	T
R	O
T	A
Y	R

first ↔ last

① Map

② Loop right

iteration (start w/ $)

$ → Y

map y to first array

$$\begin{tikzcd} & Y \arrow[dl] \\ Y \arrow[r] & R \end{tikzcd}$$

map R to first array

⋮

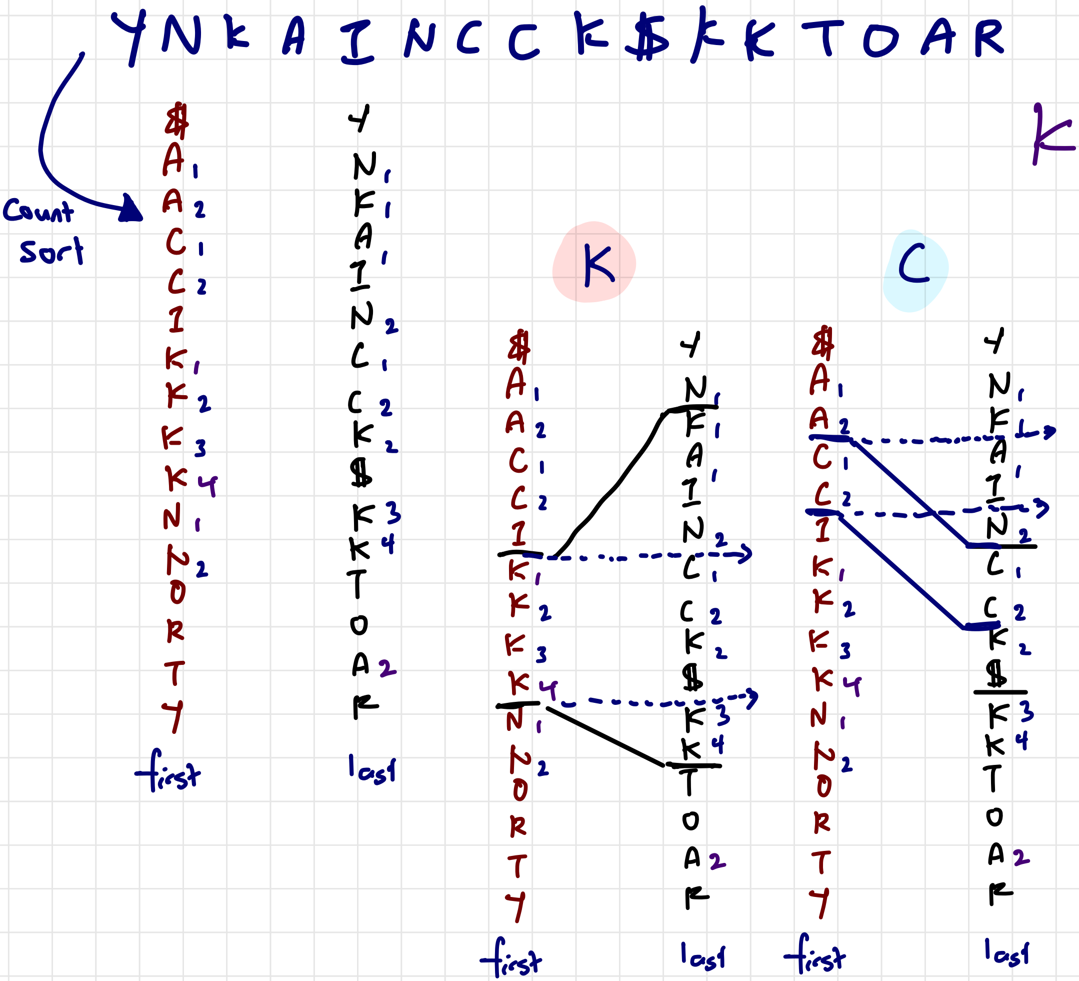
repeat until finish word.

generate backwards

KNICKKNACKATORY$

SUBSTRING SEARCH

KNACK ?



first	last
$	Y
A 1	N 1
A 2	K 1
C 1	A 1
C 2	I
I	N 2
K 1	C 1
K 2	C 2
K 3	K 2
K 4	$
N 1	K 3
N 2	K 4
O	T
R	O
T	A 2
Y	R

KNACK

① map search space

② align right

...... repeat until either found substring or not found