

COMPACTED TIE

- given a set of strings

$S_1: abaa$

$S_2: abc$

$S_3: amb a$

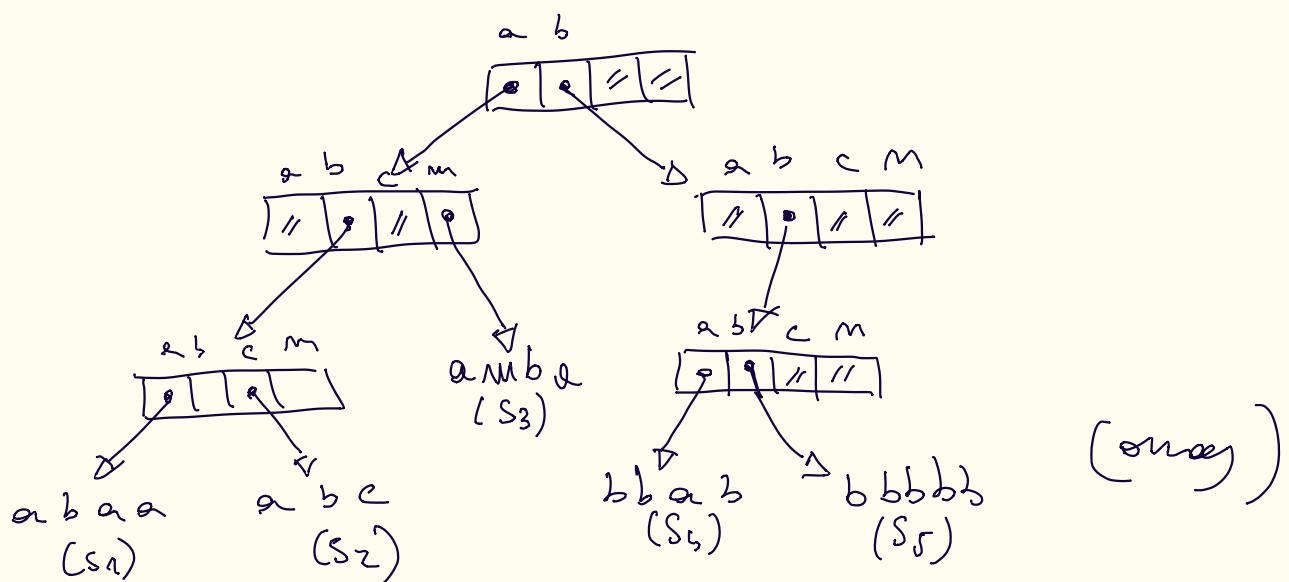
$S_4: bba b$

$S_5: b b b b b$

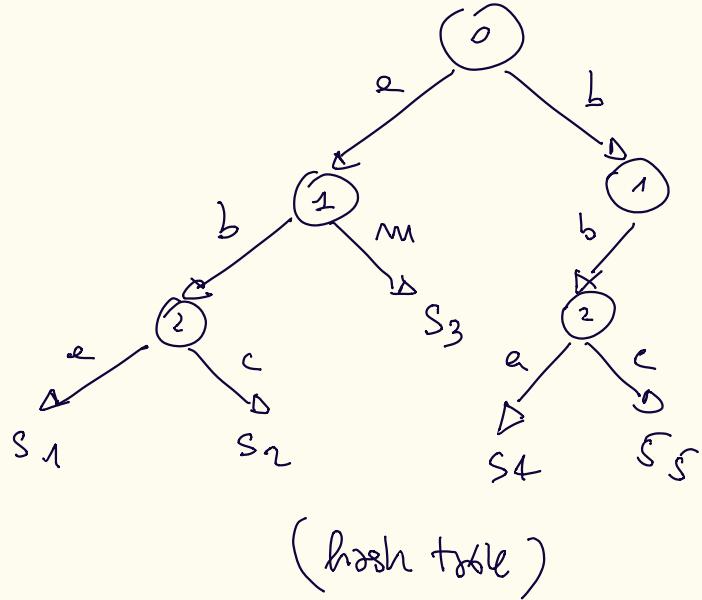
=> Draw first the UNCOMPACTED TIE
and then the COMPACTED TIE
• what is the lexicographical position of
any

$Q = abb?$

UNCOMPACTED TIE: an array in my book, size of the alphabet
 $\rightarrow \Sigma = \{a, b, c, m\}$ $|\Sigma| = 4$
 # of distinct letters of the strings



=> can rewrite the uncompactified with letters in the edge
 (skipping null pointers)



$S_1: abaa$

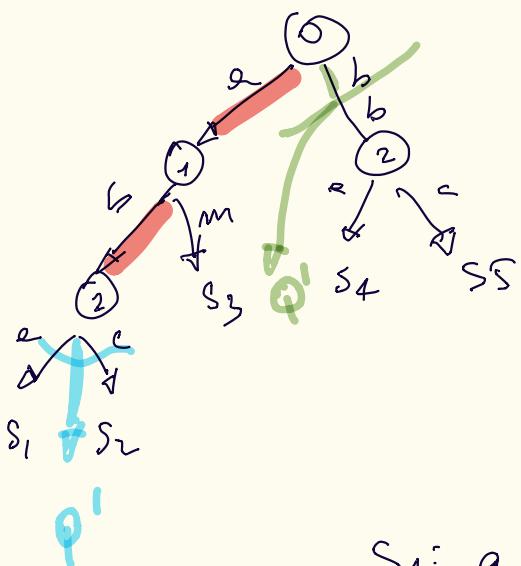
$S_2: abc$

$S_3: amb a$

$S_4: bba b$

$S_5: b b b b b$

COMPACTED TIE: compact many paths



ex. pos of

$Q = \underline{a} \ b \ b$
MisMatch

\Rightarrow lexical graphic
pos is between: $S_1 \& S_2$

$S_1 < Q < S_2$

— = —

$Q' = \underline{b} \ a$
 \times

$S_3 < Q' < S_4$

$S_1: abaa$

$S_2: abc$

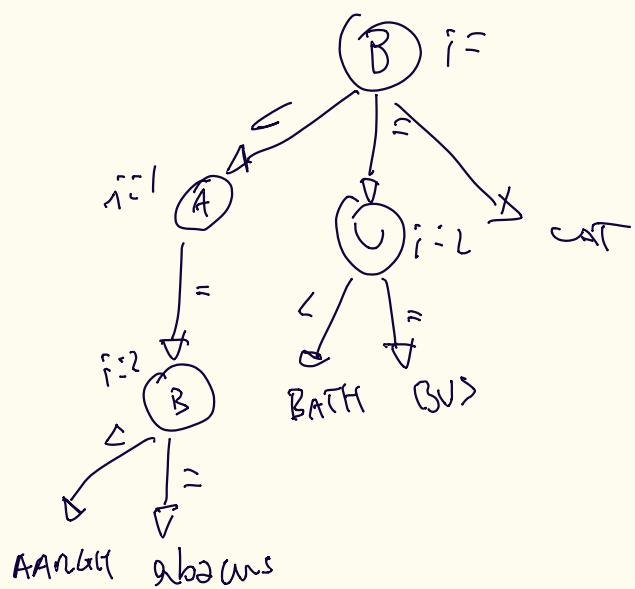
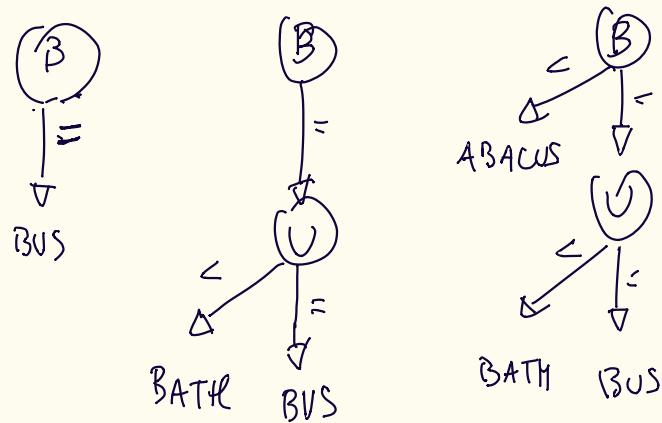
$S_3: amb a$

$S_4: bba b$

$S_5: b b b b$

TERNARY SEARCH TREE

$S = \{ \text{BUS}, \text{BATH}, \text{ABACUS}, \text{AARGH}, \text{CAT} \}$



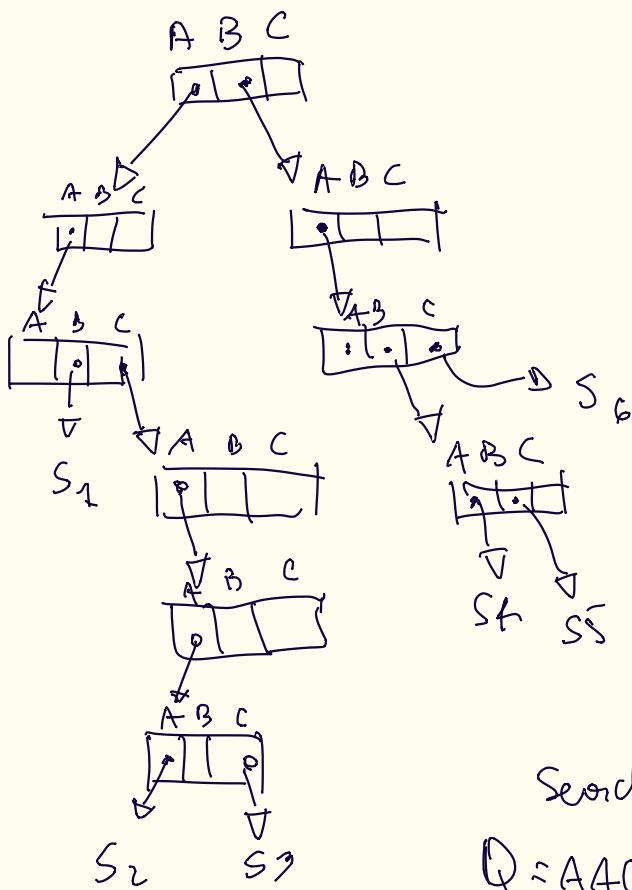
PATRICIA TREE + lex. pos

$S_1 \quad S_2 \quad S_3 \quad S_4 \quad S_5 \quad S_6$

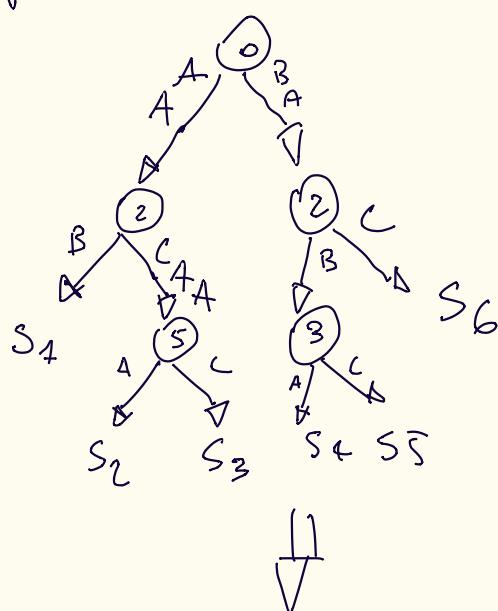
$$S = \{ AABA, \underline{A\ A C A A A C}, \underline{A\ A C A A C C}, BABA, BABB, BACA \}$$

start with a compacted tree

UNCOMPACTED

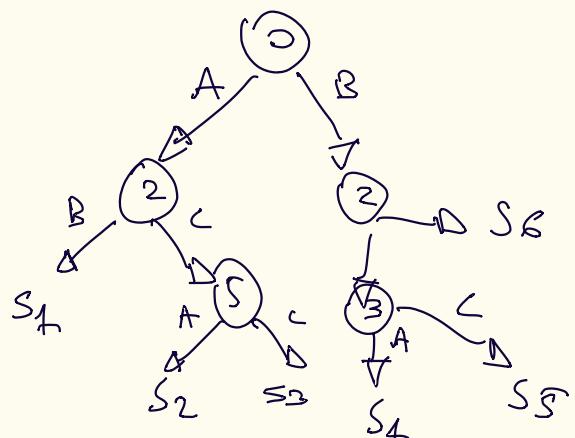


COMPACTED



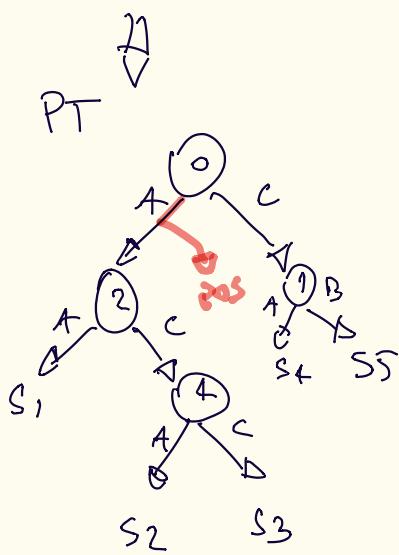
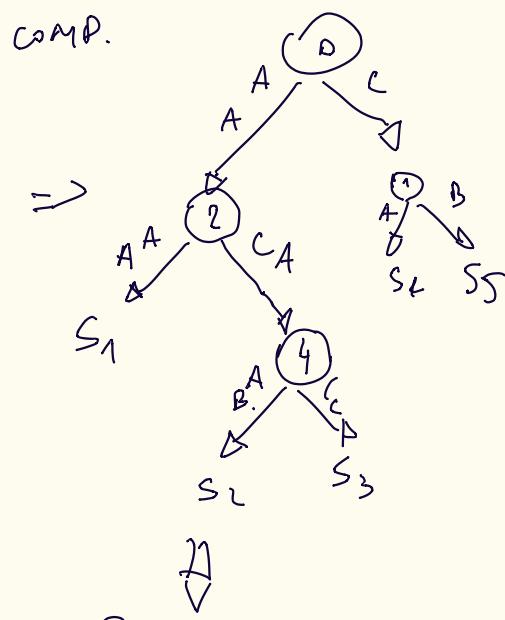
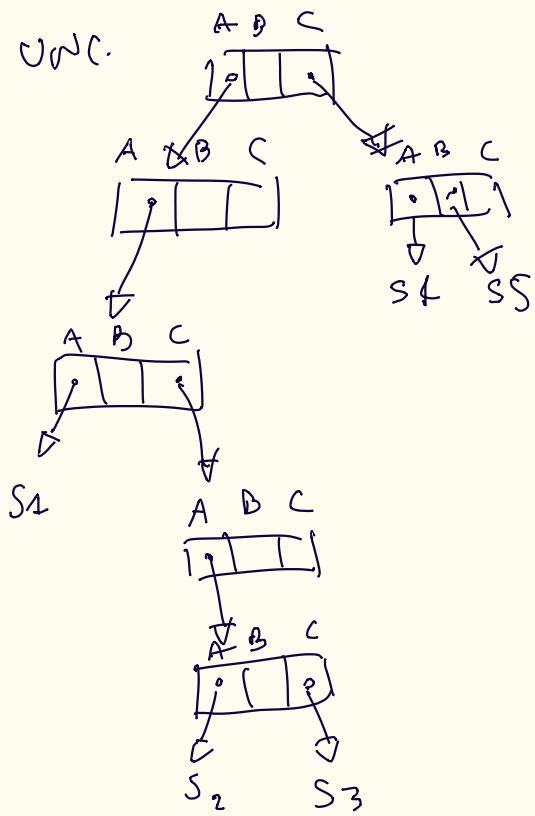
Patricia tree

Search
 $Q = AACBACD$



Patricia & lexic. search

$$S = \{ \underline{AAA}, \underline{ACABA}B, \underline{ACACC}, CA, CB \}$$



$$P = ABC$$

1) Select a discoloring leaf

$S_2 \sim A \underset{\substack{\gtrless \\ \vdots}}{A} CA \dots$

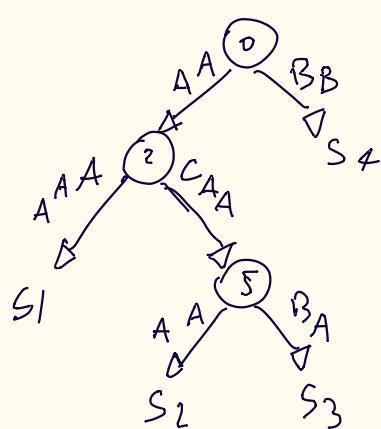
\Rightarrow the position
is to the right
of s_3

PATRICIA + LCP
 $S = \{ \underline{\text{A} \text{A} \text{A} \text{A} \text{A}}; \underline{\text{A} \text{A} \text{C} \text{A} \text{A} \text{A} \text{A}}, \underline{\text{A} \text{A} \text{C} \text{A} \text{A} \text{B} \text{A}}, \underline{\text{B} \text{B}} \}$

lex pos of

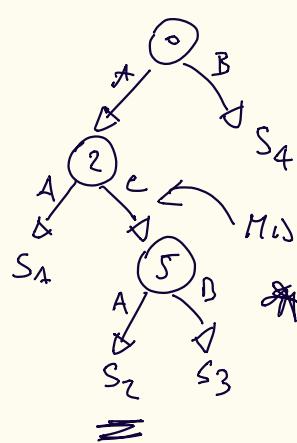
$P = \underline{\text{a} \text{a} \text{c} \text{b} \text{b} \text{a}}$

Compressed tree



PATRICIA tree

\Rightarrow



- 1) START from root
- 2) go down matching chars
- 3) return a leaf

$P = \underline{\text{a} \text{a} \text{c} \text{b} \text{b} \text{a}}$
 we found in S_2

$$\Rightarrow LCP(P, S_2) = 3$$

mismatch

S_2 is A
 in P is b

The mismatch is located

$\Rightarrow P$ mismatch is b
 on the edge I've an

A $\rightsquigarrow S_2$ B between
 S_3 & S_4

$O\left(\frac{P}{B}\right)$

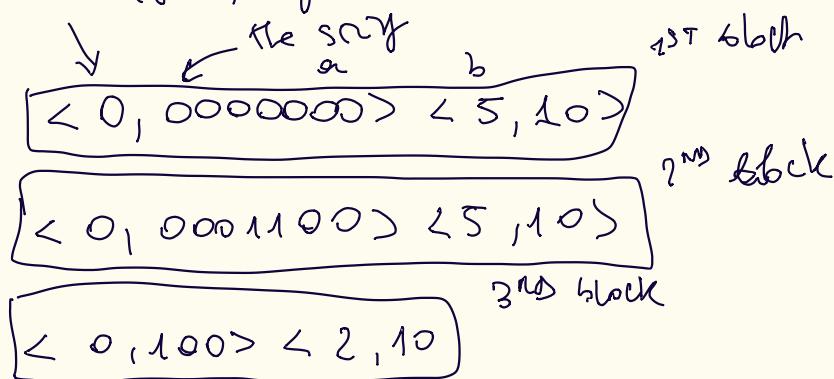
Given a set of strings; consider a 2-level indexing with patricia tree

$A = 0000000$ } block 1
 $B = 0000010$ }
 $C = 0001100$ } block 2
 $D = 0001110$
 $E = 100$ } block 3
 $F = 1010$

seed a block of 2 strings each

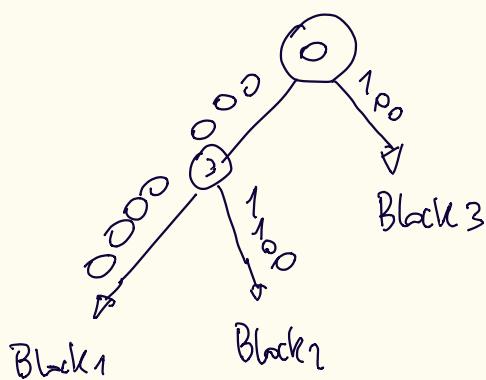
- 2 level: take strings
- sort them lexicographically
- partition in blocks of 2 strings
- compress each block by front-coding

don't copy anything before

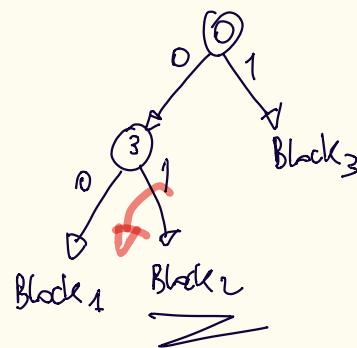


\Rightarrow copy first string of a block to the upper level, building a Patricia tree

COMPACTED TREE



PT



\Rightarrow PT in internal memory

Search for Lexicog. pos $P = \underline{0}00\underline{1}01$

$$LCP(P, \text{first string of block 2}) = 4$$

\Rightarrow since we go to the left, is referred to block 1

\Rightarrow scan the block and search lexicographic position

\Rightarrow decompress S_2 inside B_1

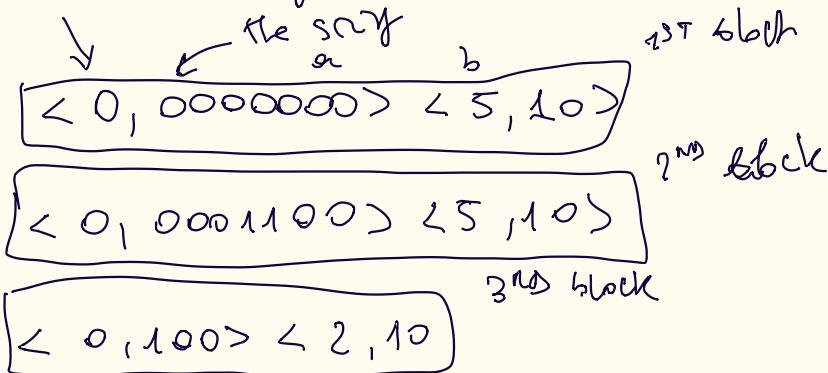
$A < B < P \Rightarrow$ lexicographical pos.

of the P is after
 the end of B_1
 and smaller than B_2

$A = 0000000$
 $B = 0000010$
 $C = 0001100$
 $D = 0001110$
 $E = 100$
 $F = 1010$

block 1
block 2
block 3

don't copy shifting before



search P#

LCP = +

SCAN BLOCK 1

none is matched

we cover $B_1 \uparrow B_2$

left extreme

search P#

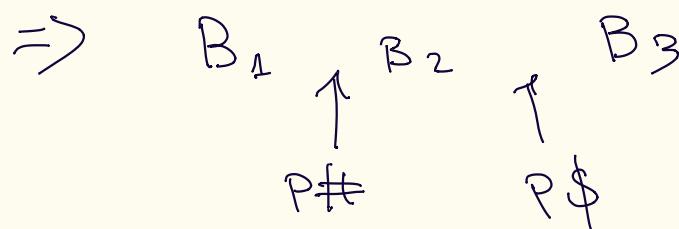
\Rightarrow we're between $B_1 \uparrow B_2$

\Rightarrow jump in B_2 (always to the left)

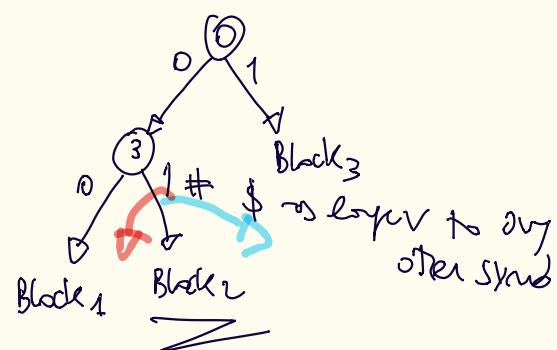
\Rightarrow compare the shifts in the block w.r.t. P\$

\Rightarrow nth extreme
before B_3

greater than others



\Rightarrow 2 occurrences \Rightarrow shifts inside Block 2



- Return all strings prefixed by P, if any:

P = 0 0 0 1 #
0 1 \$

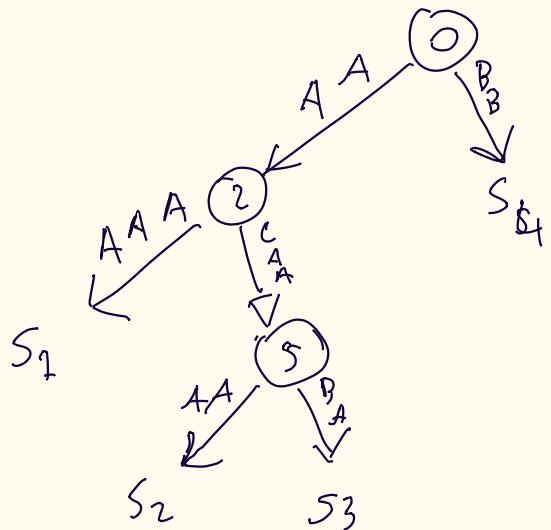
\Rightarrow Search P# (smallest)
& P\$ (biggest)

Build Patricia tree

$S = \underline{AA} \underline{A} AA, \underline{A} \underline{A} C \underline{AA} AA$
 $\underline{AA} \underline{C} AA \underline{B} A, \underline{B} \underline{B}$

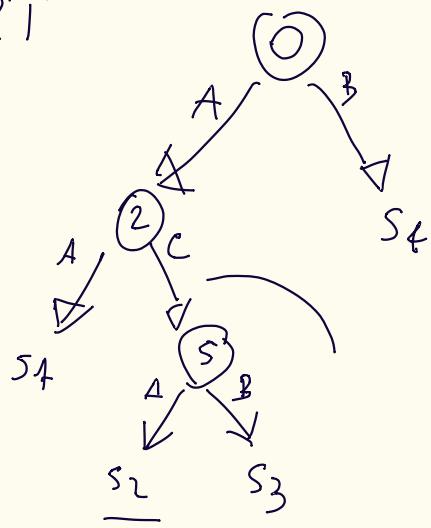
lexic pos $\Rightarrow P = aacbbca$

COMPACTED tree



PT

\Rightarrow



$$1) P = \underline{\underline{A}} \underline{\underline{A}} \underline{\underline{C}} \underline{\underline{B}} \underline{\underline{B}} \underline{\underline{A}}$$

$\Rightarrow S_2$

$$2) LCP(P, S2) = 3$$

migration: $P \rightarrow B$

$S_2 \rightarrow A$

3) we percolate upward the tree to the edge
 and swap

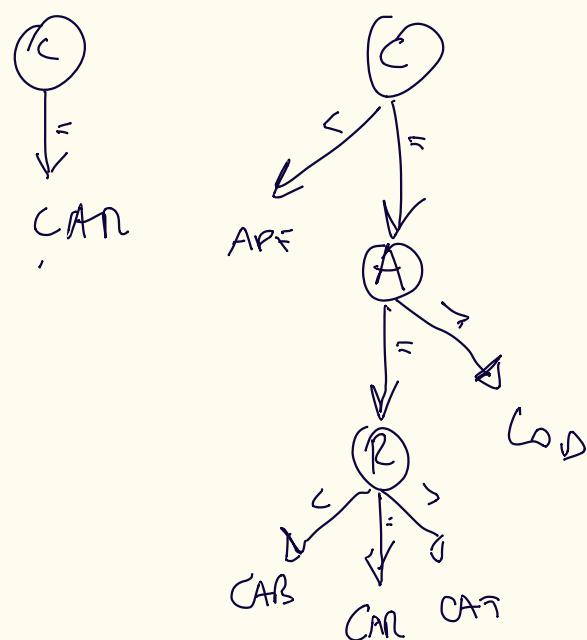


$B > A \Rightarrow \text{lex pos}$
 between

S_3 and S_4

TERNARY S.T

CAN CAT CAB COD APE



$$\frac{M}{DB} \cdot \log \frac{M}{DB} \frac{M}{M}$$