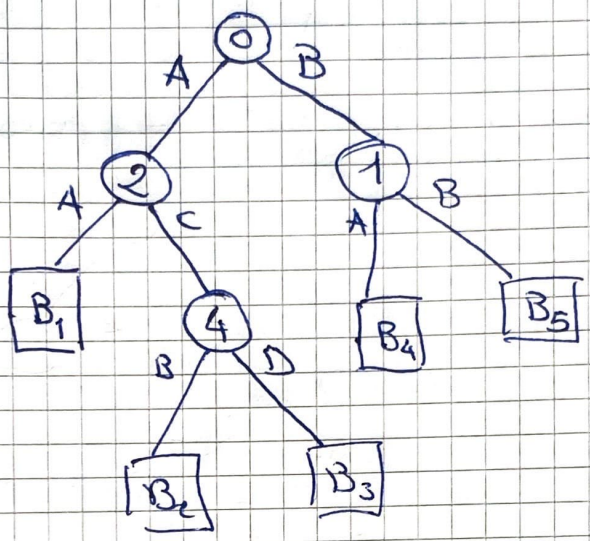
Algorithm Engineering
9 February 2024

Q1

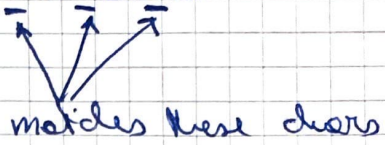
The two level indexing scheme is the following one:



AAAA <2, cA>	AACBB <4, c>	AACBBA <5, B>	BAA <2, B>
s_1 s_2	s_3 s_4	s_5 s_6	s_7 s_8
B_1	B_2	B_3	B_4

BBA <2, B>
B_5

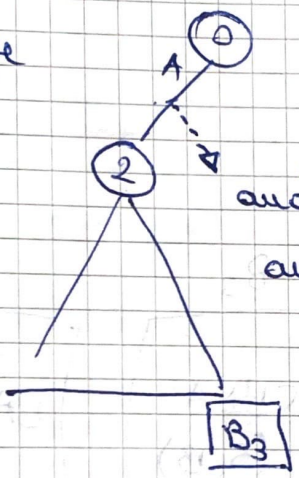
The lexicographic search for ABCBDB



and thus reads the first string of block B_3.

Computes $lcp(ABCBDB, AACBDA) = 1$,
S_5

Traverses upward the Patricia Trie, ends in the edge



and hence it is between block B_3 and B_4.

We access B_3, scan it decompressing its content, and thus find the lexicographic position at the end of B_3, and thus between S_6 and S_7.

=

The prefix search consists of performing two lexicographic searches for AAC\$ and AAC# where \$ < ... < #.
↑
any char.

If one repeats the search steps as above, finds that AAC\$ occurs in B_1 just before S_2, and AAC# occurs in B_3 just after S_6. So the answer To this query is given by the strings S_2, S_3, S_4, S_5, S_6.

Q2

$S = (4, 5, 7)$

$l = 1$, $low = 4$, $r = 3$, $hi = 7$ $\quad n = 3$

$m = 2$, $S[2] = 5$, $\quad range = [low + m - l, \; hi + m - r]$

$= [\underbrace{4 + 2 - 1}_{5}, \; \underbrace{7 + 2 - 3}_{6}]$

encode $5 - 5 = 0$ in $\lceil \log_2 (6 - 5 + 1) \rceil = \log_2 2 = 1$ bit

- output $\emptyset$ for 5
- the left call does not emit bits.
- The right call emits:

$l = 2$, $low = 6$, $r = 3$, $high = 7$

$S[3] = 7$, $\quad range = [6 + 3 - 3, \; 7 + 3 - 3] = [6, 7]$

encode $7 - 6 = 1$ in $\lceil \log_2 (7 - 6 + 1) \rceil = 1$ bit

- output 1

⇒ notice that the 7 could be even not encoded because it is the rightmost value in the sequence and this known to the decoder from "high".

=

Rice code with $K = 2$ means a quotient = 4

$R_2(4) \Rightarrow \frac{4-1}{4} = 0$; $\quad \underbrace{0}_{q} \; \underbrace{11}_{r}$ or $\underbrace{1}_{q} \; \underbrace{11}_{r}$

$R_2(5) \Rightarrow \frac{5-1}{4} = 1$; $\quad \underbrace{10}_{q} \; \underbrace{00}_{r}$ or $\underbrace{01}_{q} \; \underbrace{00}_{r}$

$R_2(7) \Rightarrow \frac{7-1}{4} = 1$; $\underbrace{01}_{9}, \underbrace{10}_{2}$ or $\underbrace{10}_{9}\ \underbrace{10}_{2}$

Q3

Invert RLE 1 given Wheeler's code:

	10	3	10	2	10	3	10
len+1	2		2		2		2
len raw	1		1		1		1
MTF	1	3	1	2	1	3	1

MTF-list = {A, B, R} (1, 2, 3)

A R R A A B B

{R, A, B} {A, R, B}

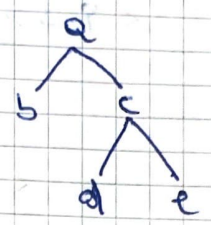
Reinsert $ in position 4 $\Rightarrow$ L = A RR $ AABB

If you invert L using the reversal BWT you get the string

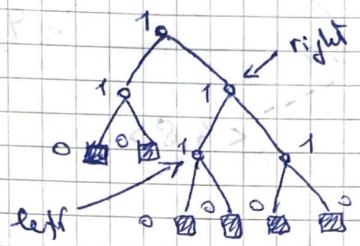
$ → A
A → R
A → R
A → $
B → A
B → A
R → B
R → B

$\Rightarrow$ ABRABRA$

Q5. The binary tree is



we complete its structure with the dummy leaves

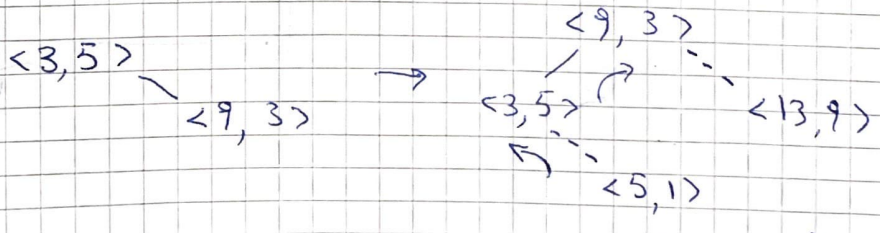


the succinct encoding is:

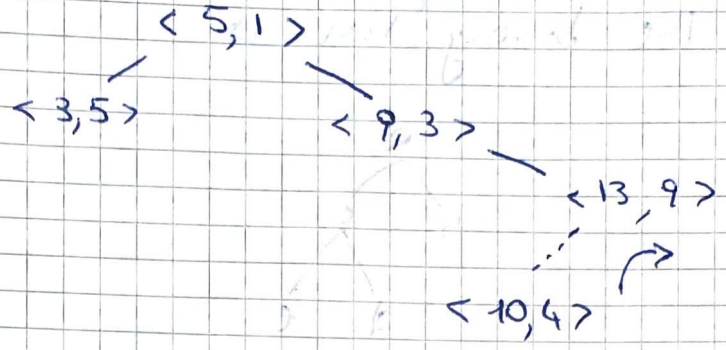
Rank	1	2	3			4	5				
B =	1	1	1	0	0	1	1	0	0	0	0
Pos	1	2	3	4	5	6	7	8	9	10	11

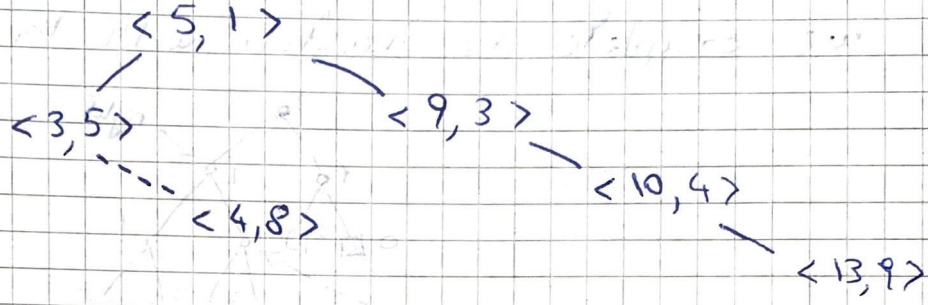
root is in position 1, hence the position of its right child (if any) is $1+2\cdot 1 = 3$. Since $B[3] = 1$ the right child does exist and its rank = id is $Rank_1(3) = 3$.
the left child of this node is in position $2\cdot 3 = 6$, and $B[6] = 1$ so it exists with id $= Rank_1(6) = 4$.

Q6.

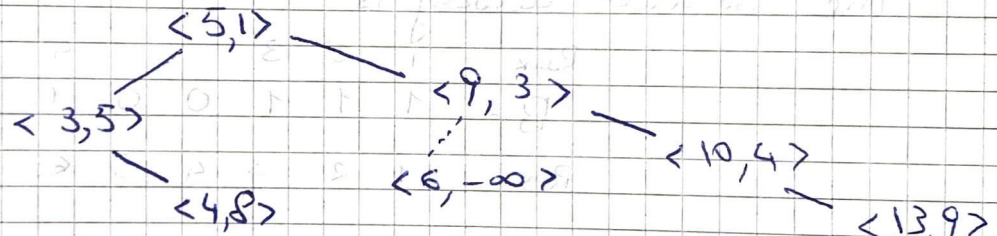


double rotation





If we want to split at key 6, we insert <6, -∞>



after Two rotations: