Information Retrieval
5/2/21

Q1. $\vec{S} = (1, 1, 3, 2, 2, 4, 2, 1, 3)$

$h_1(x) = 2x \bmod 7$

$h_2(x) = 3x \bmod 7$

BF =

0	1	2	3	4	5	6
0	~~4~~ 1	~~5~~ 5	3	3	1	5

<key, value cell>

~~<4,1>~~ <4,1> <1,1> <1,1> <2,1> <4,1> <3,1>
<1,2> <1,2> <2,2> <2,2>
<3,3> <1,3> <2,3> <2,3>
~~<1,5>~~
<1,4>
<3,5>

Query (2) = min (BF[4], BF[6]) = min (3, 5) = 3

the solve is correct because key 2 has not conflicts on cell BF[4].

Q2.

10 → 3, 10, 11, 13, 15, 17
11 → 5, 10, 11, 13, 14
12 → 3, 10, 11, 13, 21, 25, 30

↖ copies from 10 because it copies 4 items

↖ copies from

node	Ostd	ref	copy list	~~copy~~ Extra nodes
11	5	1	0111100	5, 14
12	7	2	1111100	21, 25, 30

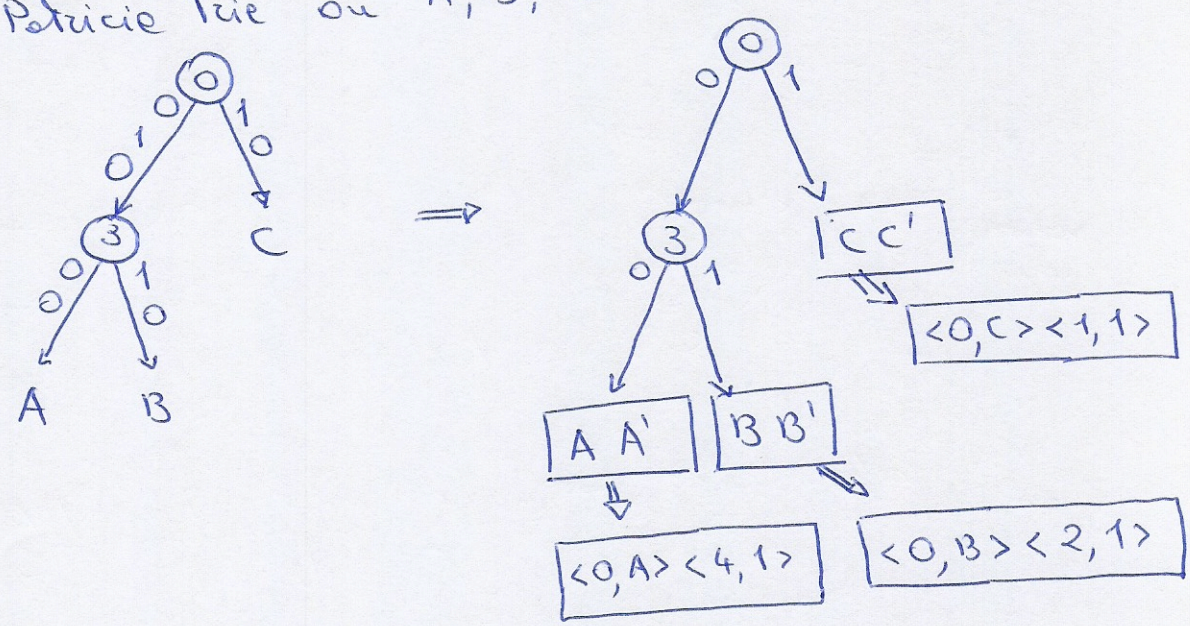
node	outd	ref.	#blocks	Copy blocks	Extra words
11	5	1	3	0, 2, 1	5, 14
12	7	2	2	1, 3, 1	21, 25, 30

first bit

Q3

- A = 01000
  A' = 01001 ⟶ FC(A') = <4, 1>
- B = 01010
  B' = 01101 ⟶ FC(B') = <2, 1>
- C = 10
  C' = 11 ⟶ FC(C') = <1, 1>

Build the Patricie Trie on A, B, C



Search S = 01011

Phase 1: Ends in the leaf of the Patricie Trie pointing to B

Phase 2: lcp(S, B) = 4 — 0 < 1 (B, S)

Phase 3: We arrive on the edge 

so the lexicographic position of S is between B and C

Then the search algorithm jumps in the disk page BB' and scans it serially for S. Then it discovers that S does not belong to the set of strings.

Q4.

F_new = x _ a b a b

h(c1 c2) = (c1 + c2) mod 7

a = 1, b = 2, _ = 3, x = 4

h: 4 3 3

F_old = a b x x a b

6 5

3 1 3

Client $\xrightarrow{011}$ Server, indicating that the second and third blocks of F_new have been identified in F_old and they are equal to "ab".

The server answers:

gap (abab | x_) = <0,0, x> <0,0, _>

Q5.

```
And_not (t1, t2)
    x = head (t1);
    y = head (t2);
    while (x ≠ NULL) do
        if (y == NULL) || (doc[x] < doc[y])
            { print doc[x]; x = next[x]; }
        else y = next[y]
```