Algorithm Engineering
20/1/21

Q1.

(bus, bath, abacus, aargh, cat)

pivot = bus, i=0

bus
bath
abacus
aargh
cat

$\xrightarrow{i=0}$

abacus
aargh
---
bus
bath
---
cat

abacus, aargh $\xrightarrow{i=0}$ $S_= \{$ abacus, aargh $\xrightarrow{i=1}$ aargh $S_<$ --- abacus $S_=$

bus, bath $\xrightarrow{i=1}$ $S_<$ bath --- $S_=$ bus

cat → STOP, one string

Q2.

$S = (1, 2, 3, 4, 6, 8, 9)$

$l=1$, $r=7$

$low = 1$, $hi = 9$

$m = \frac{l+r}{2} = 4$; $S[4] = 4$

range: $[low + m - l, hi + m - r] = [1 + 4 - 1, 9 + 4 - 7] = [4, 6]$

$Encode_2(\underset{0}{4-4})$ in $\lceil \log_2 6-4+1 \rceil = 2$ bit $\longrightarrow$ 00

On the left half:

$(\underset{l}{1}, \underset{r}{3}, \underset{low}{1}, \underset{hi}{3}) \longrightarrow$ no bit emitted

On the right half

$(\underset{l}{5}, \underset{r}{7}, \underset{low}{5}, \underset{hi}{9})$

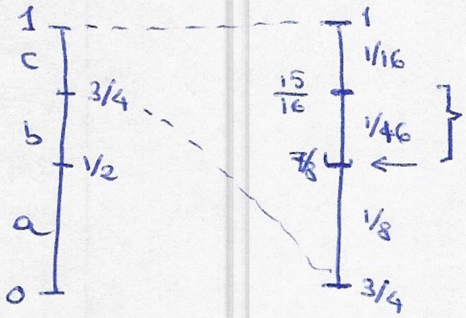
$m = 6 \longrightarrow S[6] = 8$

range $= [5 + 6 - 5, 9 + 6 - 7] = [6, 8]$ $\longrightarrow$ 10

$Encode_2(8-6) = 2$ in $\lceil \log_2 8-6+1 \rceil = 2$

Q3. $p(a) = 1/2$, $p(b) = p(c) = 1/4$

compressed sequence $= (111)_2 = \frac{1}{2} + \frac{1}{4} + \frac{1}{8} = \frac{7}{8}$



sequ = cb

$s_0 = 1$ $s_1 = 1/4$
$l_0 = 0$ $l_1 = 3/4$

Q4.

$S_1 = 1, 8$

$\updownarrow =$

$S_2 = 1, 2, 5, 7, 10, 15, 20$

$=$

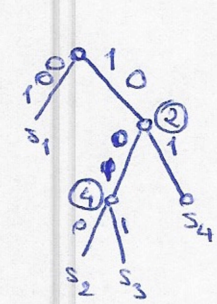
$S_1 = 8$

$S_2 = 2, 5, 7, 10, 15, 20$

$\uparrow 2^0$, $\uparrow 2^1$, $\uparrow 2^2$

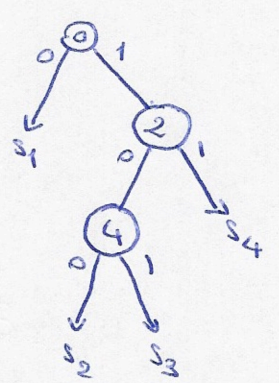
binary search here

Q5.



compacted Trie

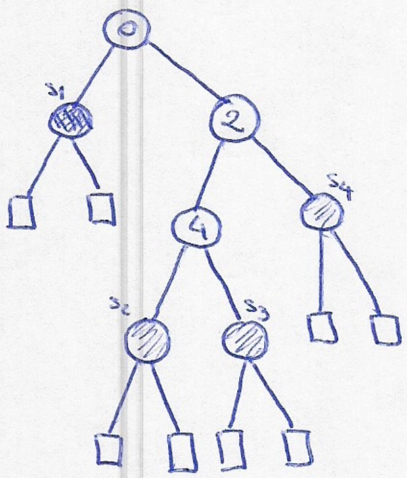
$\longrightarrow$



Patricia Trie

- Given $P = 110 \longrightarrow$ downward search stops at 4 and thus any descending string can be taken.
- Let it be $s_2 \rightarrow lcp(P, s_2) = 1$ and $s_2 < P$
- Going upward we stop at the edge 0–2 and go right $\Rightarrow P > s_4$

For the succinct encoding of a binary tree we proceed as seen in class:



⊙ we do not need to store the first bit of every edge because they are surely {0, 1}.

• We need to store the integers of the internal nodes, and use an array, using the BFS order as the succinct encoding.

$$I = \begin{array}{ccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ \end{array}$$
$$I = [0, s_1, 2, 4, s_4, s_2, s_3]$$

We create the binary sequence that encodes the tree structure

$$\begin{array}{c|ccccccccccccc} & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 \\ B = & 1 & 1 & 1 & 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ & 1 & 2 & 3 & & & 4 & 5 & 6 & 7 & & & & \end{array}$$

there is a bijective mapping between these numbers and the entries of I.

• We assume to build a Rank/Select data structure on B.

We know that: $B[i] = 1 \Rightarrow$ node of the patricia tree
$= 0 \Rightarrow$ dummy leaf

$$\left.\begin{array}{l} \text{left\_child}(\hat{x}) = \text{Rank}_1(2 \cdot \hat{x}) \\ \text{right\_child}(\hat{x}) = \text{Rank}_1(2 \cdot \hat{x} + 1) \end{array}\right\} \text{ where } \hat{x} \text{ is the numbering indicated below B and thus the one of the 1s.}$$

1) We start with $\hat{x} = 1 \longrightarrow I[\hat{x}] = I[1] = 0$

2) $P[0] = 1 \longrightarrow \text{right\_child}(\hat{x}) = \text{rank}_1\underbrace{(2 \cdot 1 + 1)}_{B[3] = 1}$ over $B = 3 = \hat{x}$

Since $B[3] = 1$ the right child does exist, and $I[3] = 2$ gives us the value stored in it. Notice that checking the left and right children we discover that it is an internal node.

3) $P[2] = 0 \longrightarrow \text{left_child}(\hat{x}) = $
$B[2 \cdot 3] = 1$
$\text{rank}_1(2 \cdot 3) = 4 = \hat{x}$