

①

ADS Exam-3

Priya lakshmi  
Theagarajan

UID: 9298 0120

①(a) Suffix tree is used to arrange strings as prefix list with # in the last place. It is used in applications like finding out Gene sequence & in longest-repeating substring & Longest Common substring.

Tries store edge information and also has a field to figure out the skipped over characters.

But suffix tree stores the common part as a string in the node. Tries ~~also~~ ~~have~~ is used in processing social security number while suffix tree is used in string matching.

1(b). Suffix tree GATAGACA #

GATAGACA # → 1

ATAGACA # → 2

TAGACA # → 3

AGACA # → 4

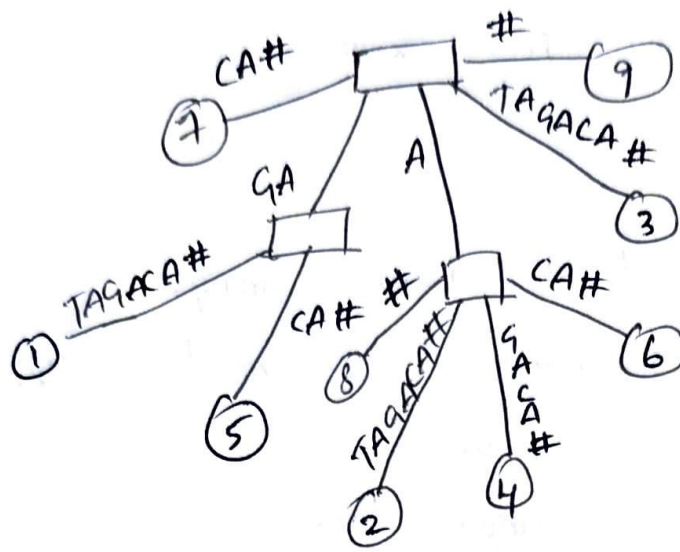
GACA # → 5

ACA # → 6

CA # → 7

A # → 8

# → 9



Q2) (a) Segment trees are used in

- Computational Geometry
- VLSI mask verification
- Sentry location.

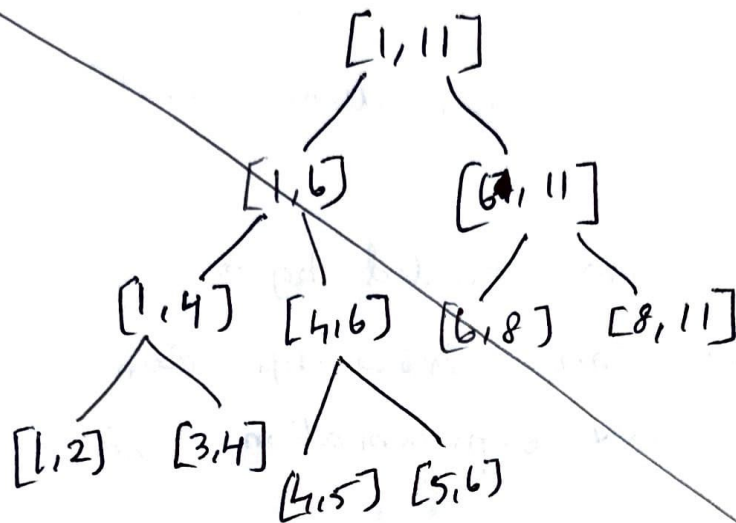
(b). SEARCH operation in segment tree.

Search  $[a, a+1]$  → Complexity is  $O(\log n + s)$

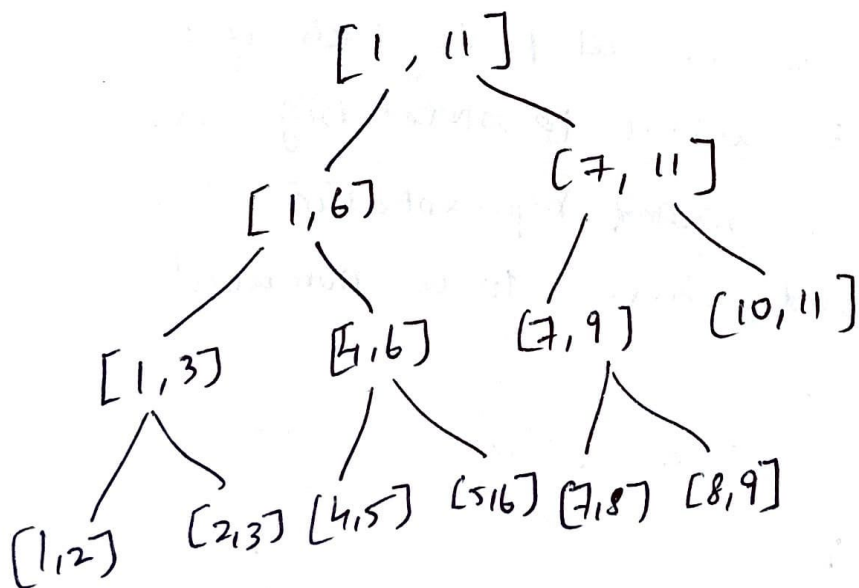
where  $s$  is the number of segments.

- ① Follow unique path from root to leaf node for interval  $[a, a+1]$
- ② Report all segments stored in the nodes on this unique path.
- ③ No segment is reported twice since one segment cannot be stored in both a node and its ancestor

2(c) Segment tree with root range  $[1, 11]$  ②



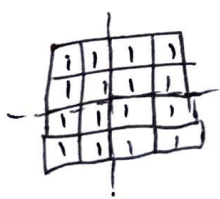
2(c) Segment tree with root range  $[1, 11]$



Q(3) (a) The Quad tree is used in image processing area and is used for region data. Example. Roads & rivers in state / country. It is a binary image, where image is divided into pixels & Each pixel is black (0) or white (1).

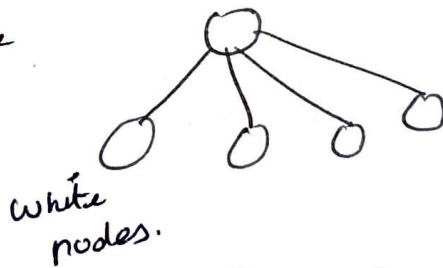
The space complexity of a quad tree is  $O(n^2)$   
 The time complexity of scaling operation is  $O(n^2)$   
 The time complexity of intersection operation is  $O(n^2)$

Q3(b) Yes, the space needed by the quad tree representation of a black & white image is always substantially less than the matrix representation. This is because, in quad trees, only when the node is gray it is expanded further to get the 4 children. For white & black image the nodes would be either all 0 or all 1 in each of the 4 quadrants. This will be useful to store big white & black images, in matrix representation however all the pixels would have to be numbered as either 0 or 1.



Matrix representation

All are white



Quad tree

size difference

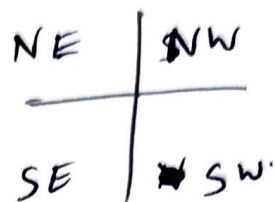


Q(3)(c)

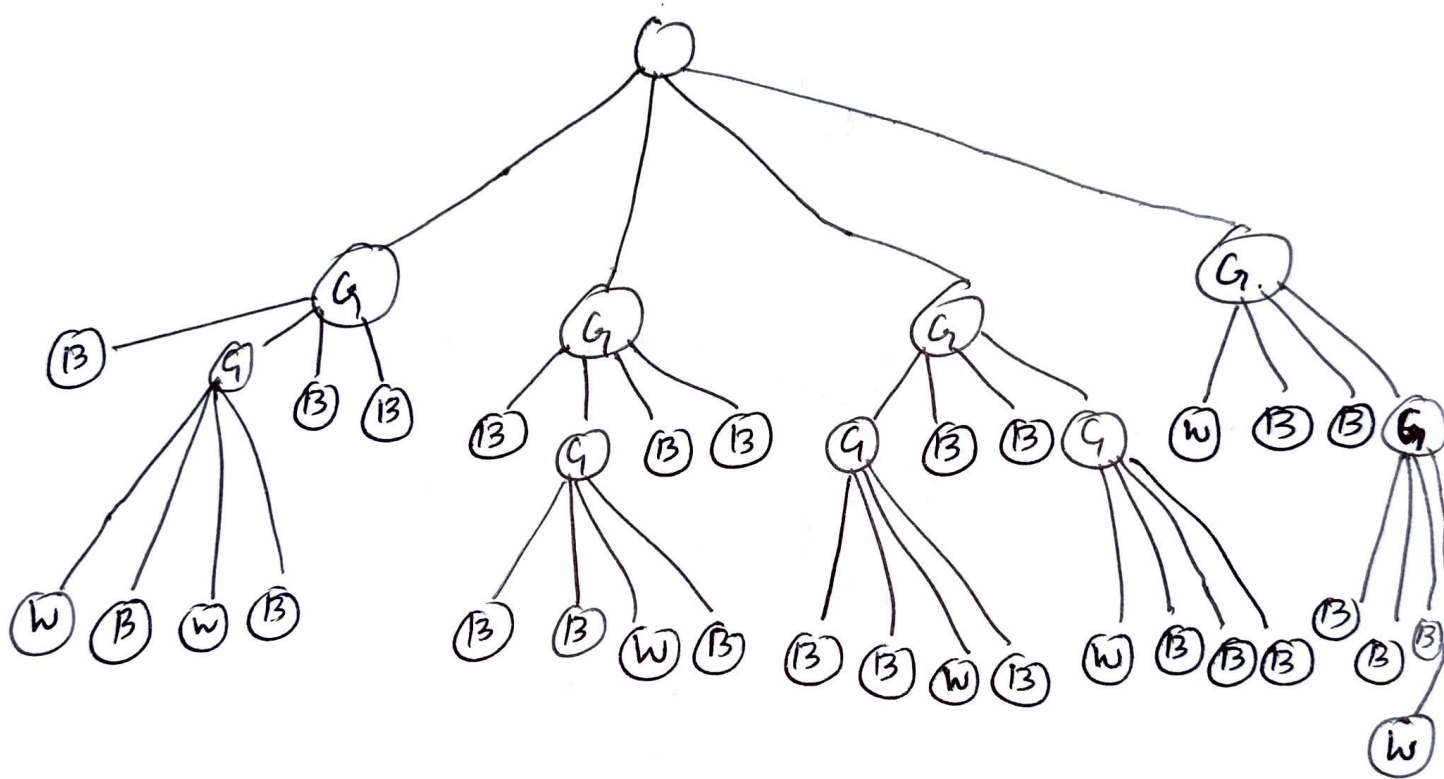
# QUAD TREE.

3

- (W) → white
- (B) → Black
- (G) → Gray.



Ordering of nodes are done like above  
NE, NW, SE, SW.



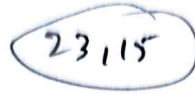
Q(4)

# Radix Priority search.

(K)

insert (23, 15)

[0, 32)

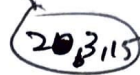


insert (20, 8)

[0, 32)

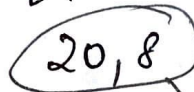


[16, 32)

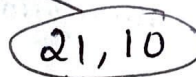


insert (21, 10)

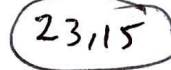
[0, 32)



[16, 32)

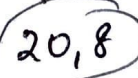
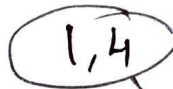


[16, 24)

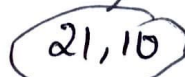


[24, 32)

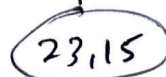
insert (1, 4)



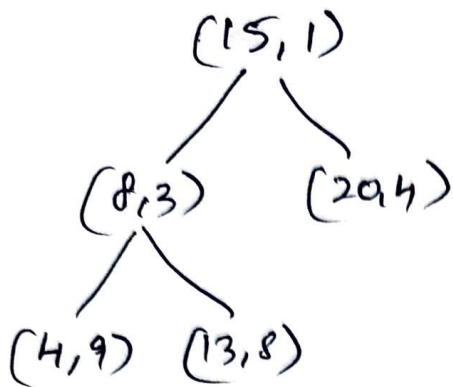
[16, 24)



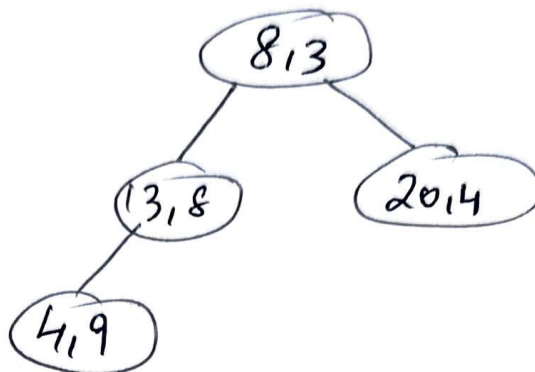
[20, 24)



(b)



Remove 15



After removing 15,

the since 4 is a min heap

find the next minimum 4 & place it as the root.

When there is a deficit, recursively promote the node with minimum 4 as the parent.