Quad Trees

google maps

barber shops neer me

x

rest near me

X

A B

D C

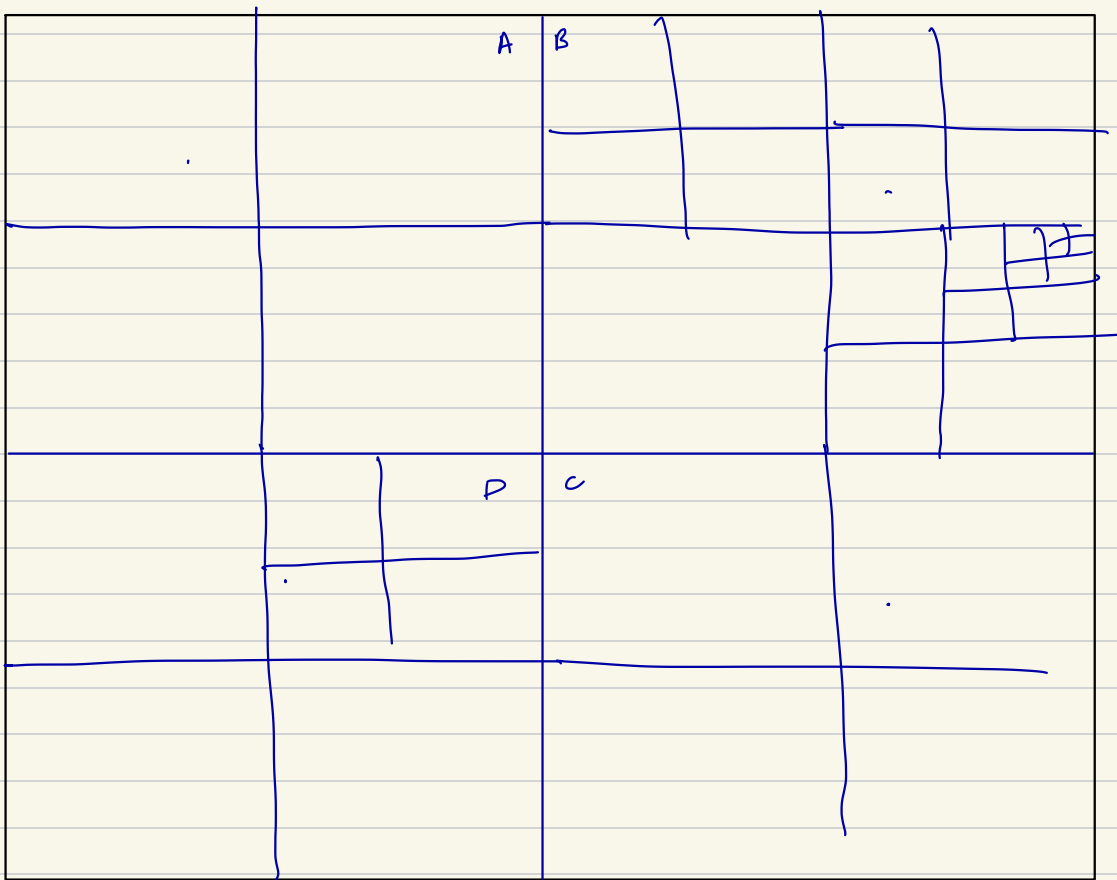A   B.   Tree   (A+B)/2

$\left(\dfrac{A+C}{2}\right)$

C   D

A 107-8

(I)
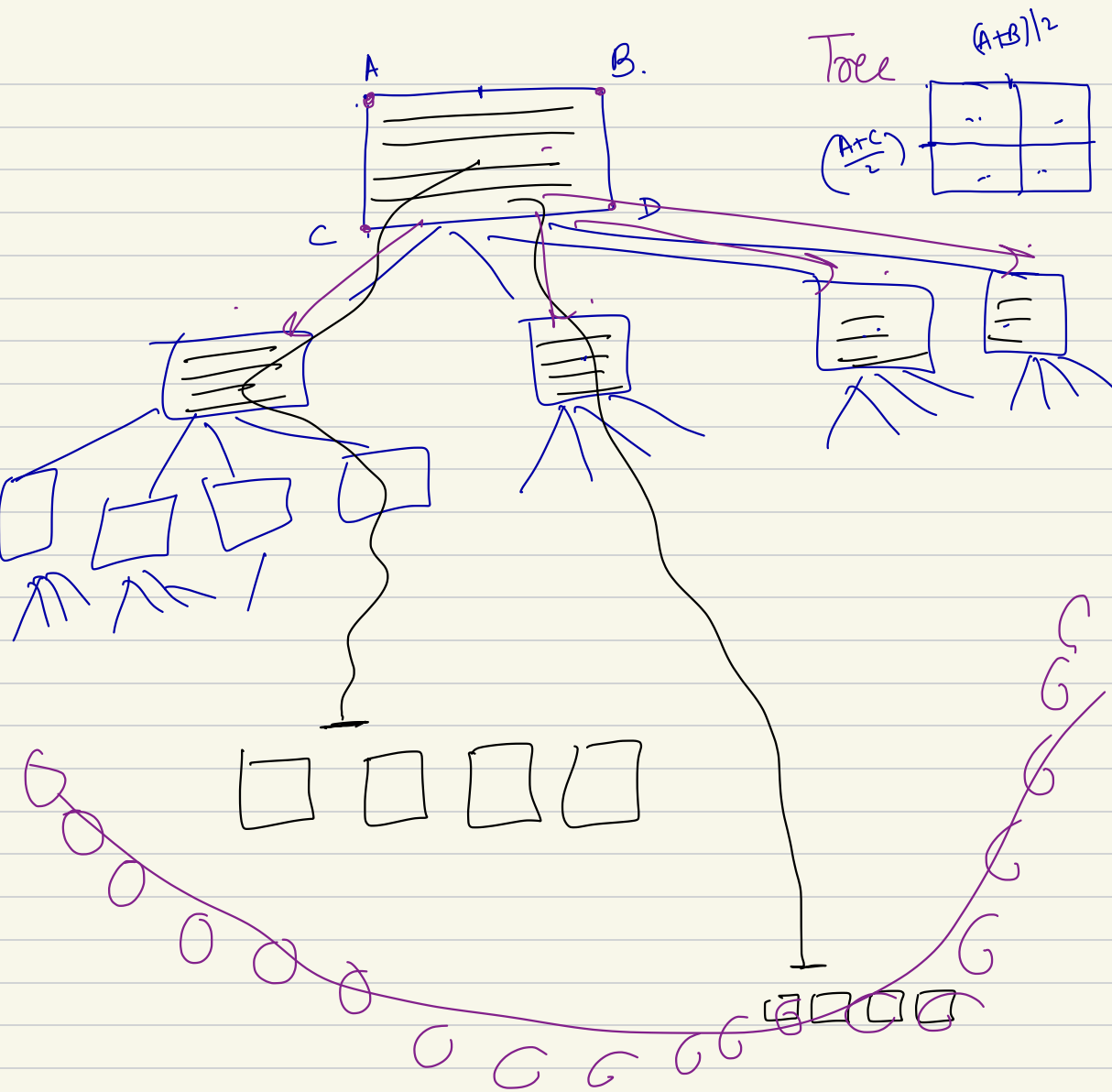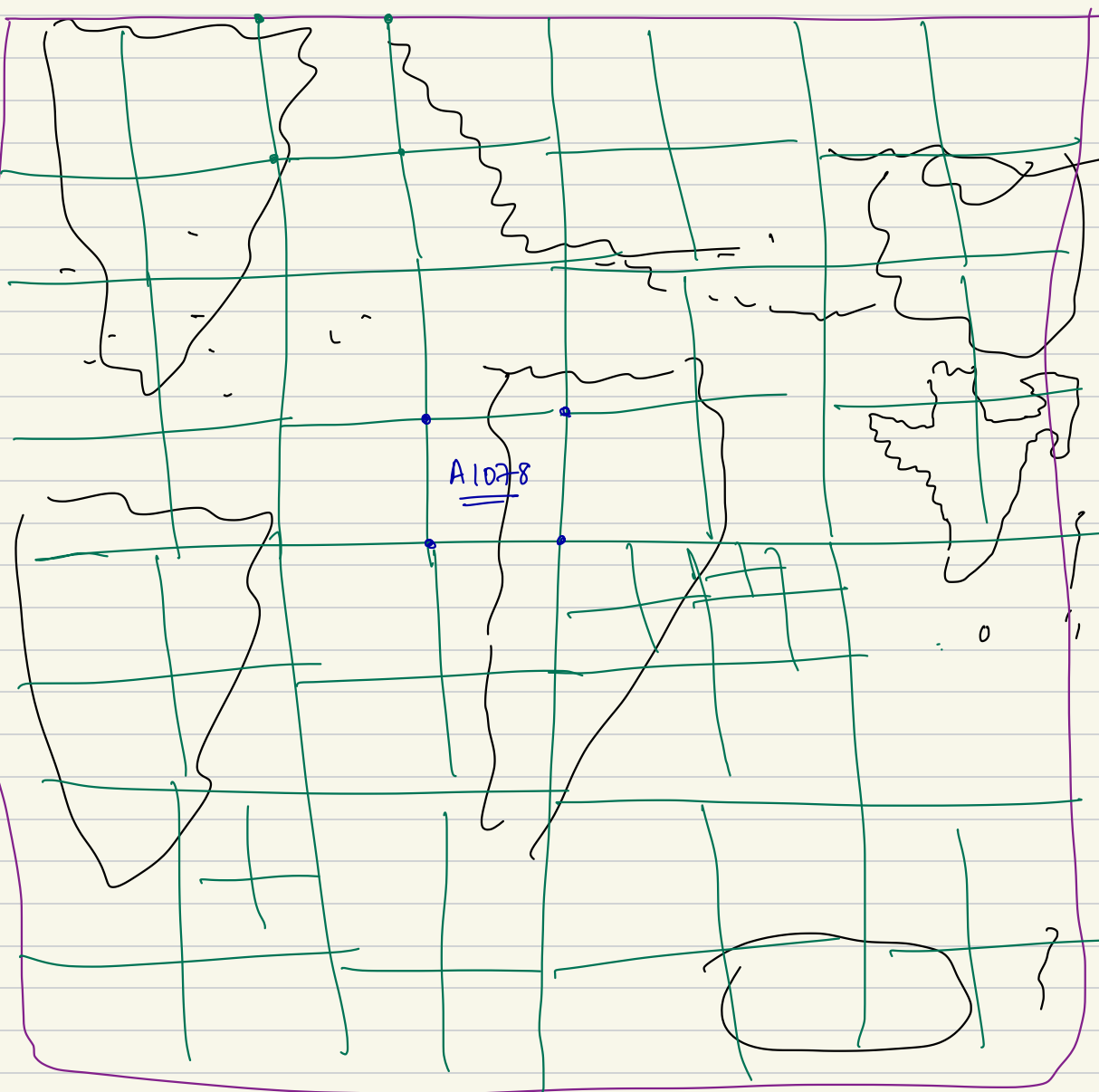
leaf-cell-id	T_L	T_R	B_L	B_R
A 1078	x_1, y_1	x_2, y_2	$x_3 y_3$	$x_4 y_4$
	-	-	-	.

(II)

100 M places

PlaceId	Lat	long	description	Title	leaf-cell-id

Finding Grid Id for a point

(x, y)

y

$x_1 y_1$

$\left[\dfrac{x_1+x_2}{2}, y_1\right]$

$x_2 y_1$

A

B

$\left(\dfrac{x_1+x_2}{2}, \dfrac{y_1+y_2}{2}\right)$

$x_1, \dfrac{y_1+y_2}{2}$

$x_2, \dfrac{y_1+y_2}{2}$

D

C

x_1, y_2

$\left[\dfrac{x_1+x_2}{2}, y_2\right]$

$x_2 \quad y_2$

x

(0,100)   (50,100)   (75,100)   (100,100)

110

109   108

(50,75)   (75,75)   (100,75)

107

(50,50)106

(75,50)   (100,50)

(0,50)

103

x

104

(0,25)   (25,75)   (50,75)

105

101   102

(0,0)   (25,0)   (50,0)   (100,0)

(0,100)   (100,100)

(0,0)   (100,0)

(0,50)   (50,50)

(50,50)   (50,50)   (100,50)   (50,100)   (100,100)   (0,100)   (50,100)

110

105

(0,0)   (50,0)   (50,0)   (100,0)   (50,50)   (100,50)   (0,50)   (50,50)

101   102   103   104
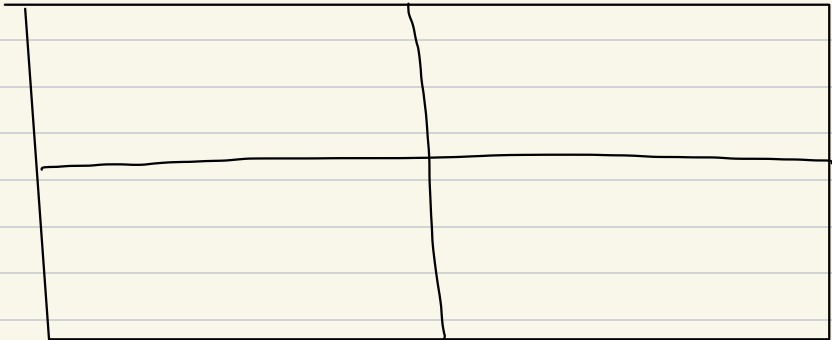
106   107   108   109
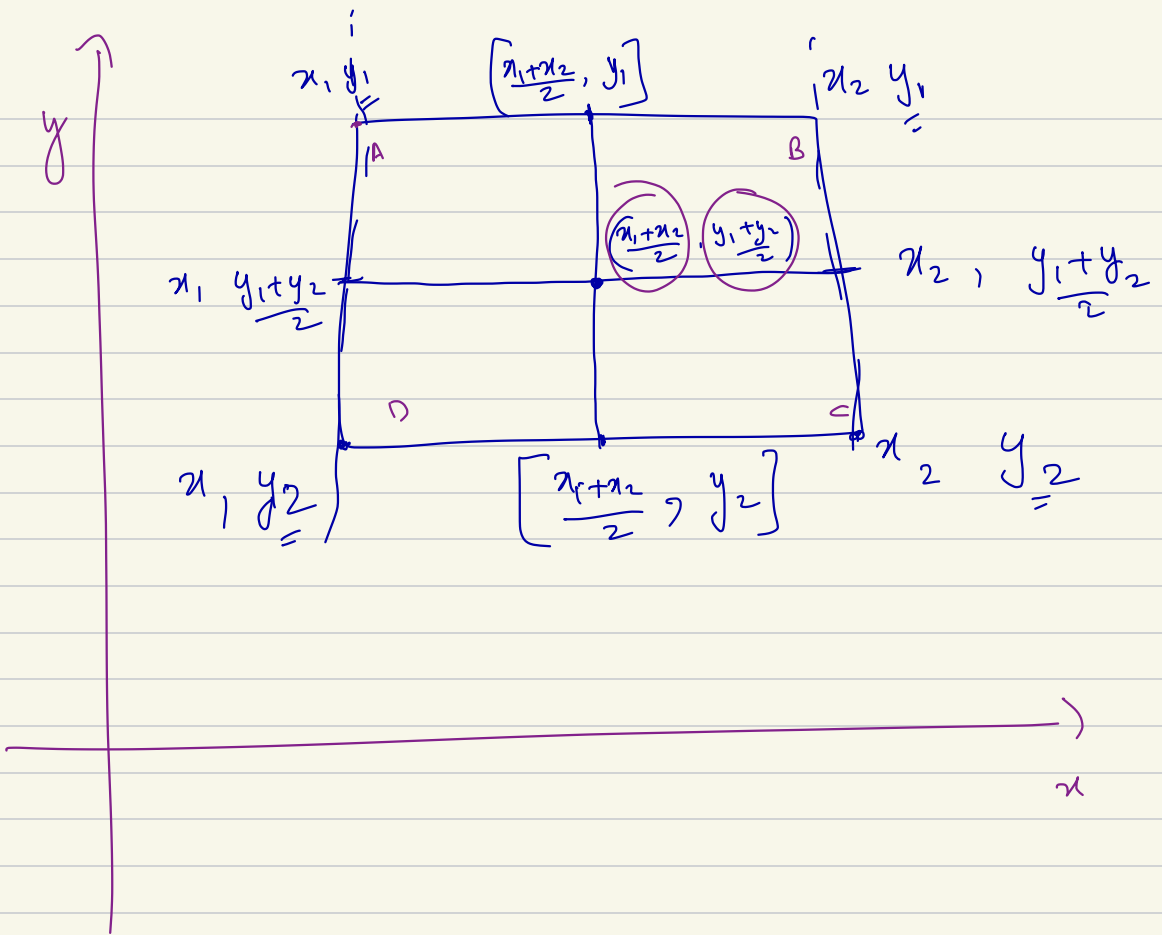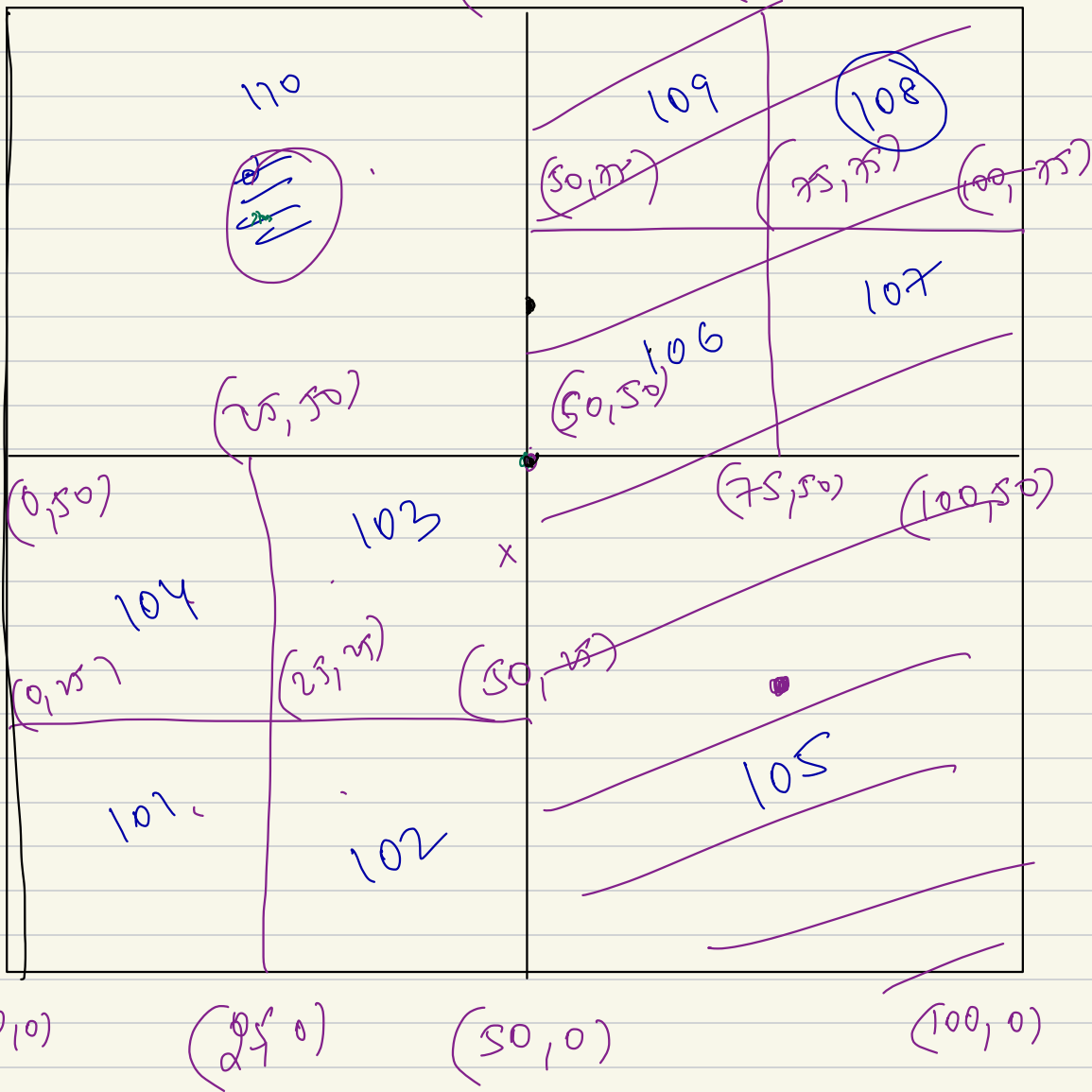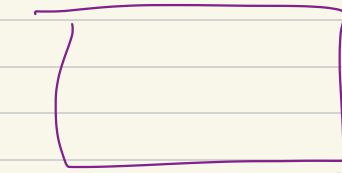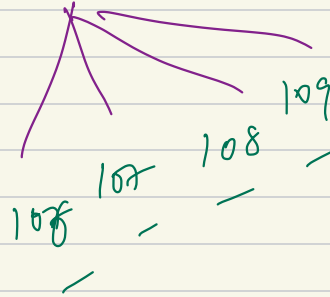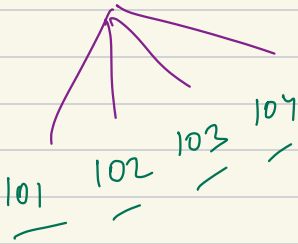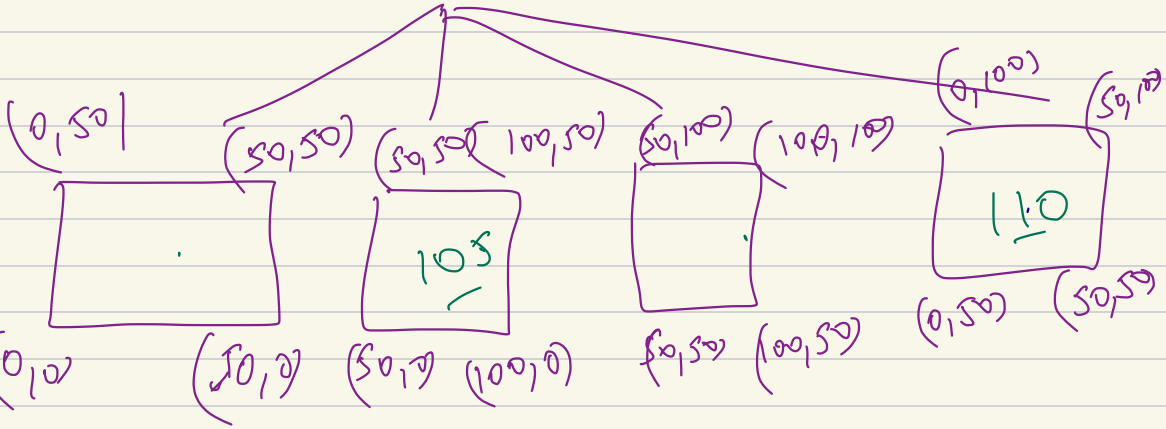
Find (20, 60) ? ? !

↓

(110) ☆

Find (45, 90) → 110

Find (66, 20) → 105
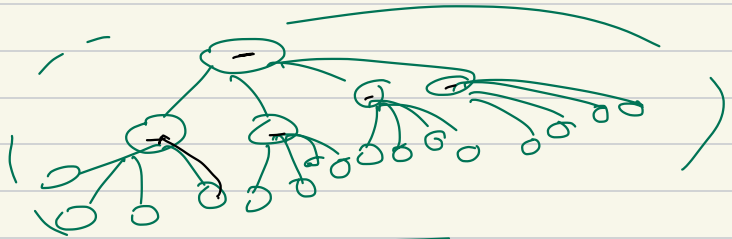
Find (80, 78) → 108 ☺

Quad Tree

BUILD   GOOGLE MAPS   API

Step 1:   Tabulate all places of interest

Place Id	Place Type	Title	Description	Lat	Long	leaf cell-id

Too →

Step 2:   Initialize a Quad Tree ☺

Leaf Cell Id	T_L	T_R	B_L	B_R

hm 1

grid cell id   →   4 Pts

hm 2:

    cell   →   4 children cell ids

✓

hm 3

   cell → parent cell id

Step 3: Now support Queries

☆ point (x,y) → Find nearby barber shops

① Find Cell Id for (x,y)

→ A 1080

② Places Collection

→ All ≤ 100 places for that cell.

Ⅰ → Filter ⇒ Barber shop

Rank Them

User 😊

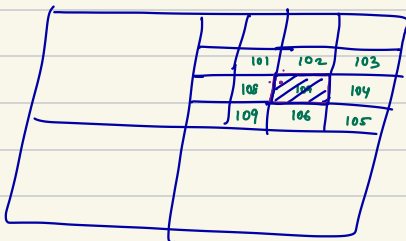Ⅱ Give you a few neighbouring
😊 cell ids as well.

① All places inside those cells
   ep. 6 neighbours ≡ 600 places
② Filteration
③ Ranking
   ↳ User 😊

101	102	103
108	100	104
109	106	105

100M places of interest   10L X 100

10Cr   Place of interest

7B people

Step4: How to handle a new establishment being
created

890 → Place (x, y)   Indian
                       Restaurant   Saga Rata — _____

① Find leaf- cell-id [x, y] ⟶ A 2078

② Add 890 = Place Id in the Places Collection

③ Check if cellid = A2078 > 100 places

              ≤100   > 100

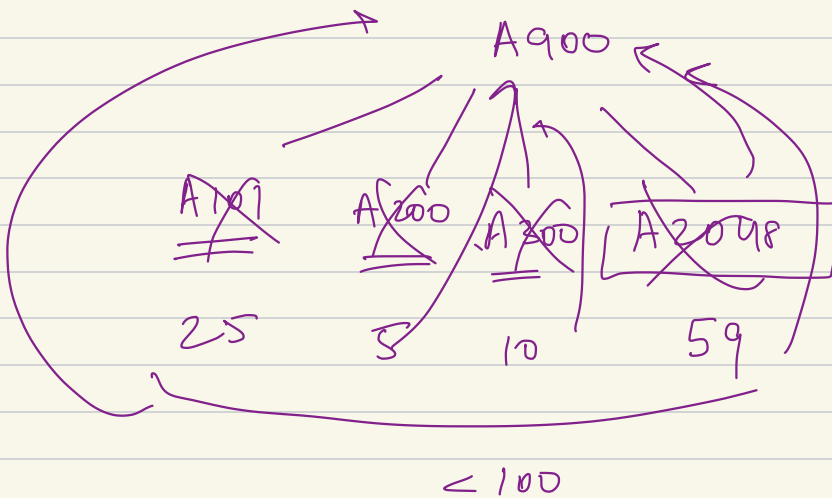   No   A 2078 / divided into
              ☺   4 new cells

                      A 909   A 9092 A9093 A9094

                                  divided →

                        101 places

Step 5: Remove an establishment. ✓

107 — A20 48 ✗

(A2048) ⟶ 60 − 1 = 59

A900

A107   A200   A300   A2048

25   5   10   59

< 100

Calculation

7B people

700M places   1 place: 10 people

70M places   1 place : 100 people

100M places of interest / world   :)

(100 M)

$$\text{\# Leaf Nodes} = \frac{100 \text{ M}}{20}$$

$$= 5 \text{ M} \quad ☺$$

leaf nodes

$$\frac{5M}{64}$$

$$\frac{5M}{16}$$

$$\frac{5M}{4}$$
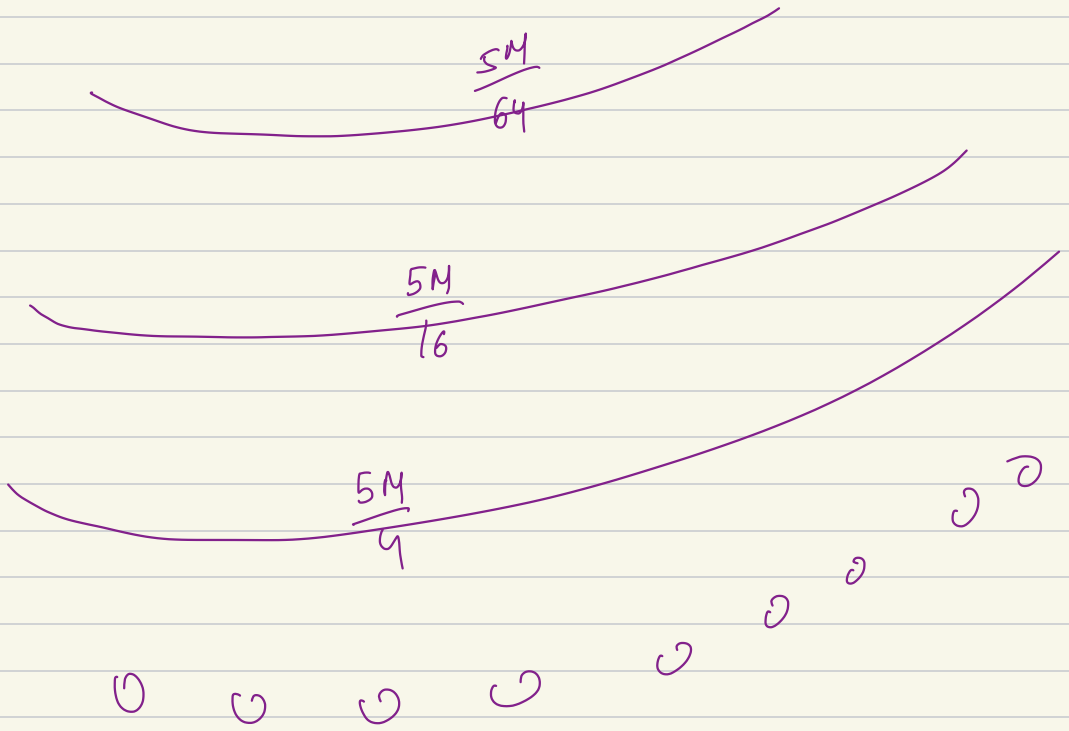
Total # Nodes in whole quad Tree

$$= \quad 5M \quad + \frac{5M}{4} \quad + \frac{5M}{16} \quad + \frac{5M}{64} \quad + \frac{5M}{256} \quad \ldots \ 1$$

$$= \quad 5M \left(1 + \frac{1}{4} + \frac{1}{16} + \frac{1}{64} \ - - - - - \right)$$

$$\frac{1}{3/4} \quad = \frac{4}{3}$$

$$= \quad 5M \times 1.33$$

6.5 M nodes

STORAGE SPACE

Quad Tree - Nodes

100 M places

- ✓ Place id → 8B
- ✓ Title
- ✓ Descrn → 50B
- ✓ Place Type → 8 B
- ✓ Cell id → 8B

200B ~~100B~~.

$100 M \times 100 B$

$10 \times 10^3 \times MB$

$= \cancel{100 GB}$

20GB

6.5M nodes

6.5 M nodes
$\times$ 200 Bytes

$13M \times 100 B$
$= \quad = \quad =$

$1.3 \times 10^3 \times MB$

$= 1.3 GB$ 🙂🙂

6.5M nodes
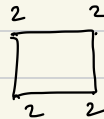
$8 \times 8 = 64 Bytes$

8 Bytes

40 Bytes

112 B

200 Bytes

21.3 GB 🙂🙂🙂

Uber

100 M   Places of Interest

Drivers ≡ Places of Interest

7 B
700 M   drivers   1 driver : 10 people

70 M   driv   1 driver : 100 people

7 M   drivers   1 driver : 1000 people

10 M   Uber drivers / world.

Uber

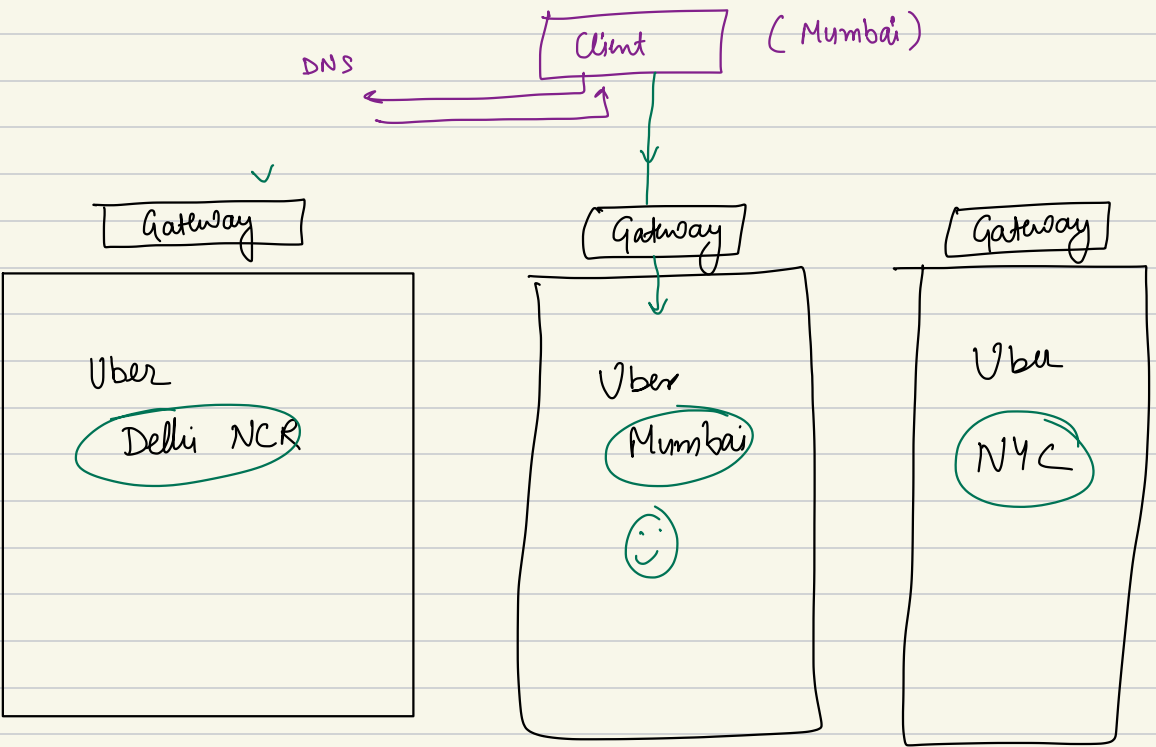To be a collection of independent

small Uber systems

| Uber NYC | Uber Delhi NCR | Uber Mumbai + nearby |

DNS

Client (Mumbai)

Gateway ✓

Gateway

Gateway
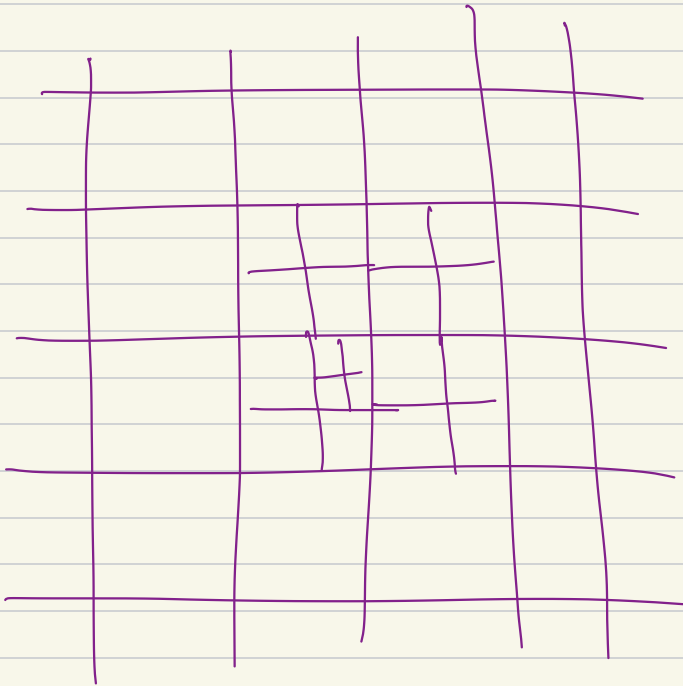
Uber

Delhi NCR

Uber

Mumbai

:)

Uber

NYC

Zoom into any 1 city cluster

ex. Mumbai
city cluster

500k drivers in Mumbai City
cluster

How to initialize a quad Tree
given when all drivers keep on moving

☆ Based on some parameter which
can be a good approximation

Ⓘ   Initialized Quad Tree ☺

Ⓘ   Every Uber driver based on
     their current location can be
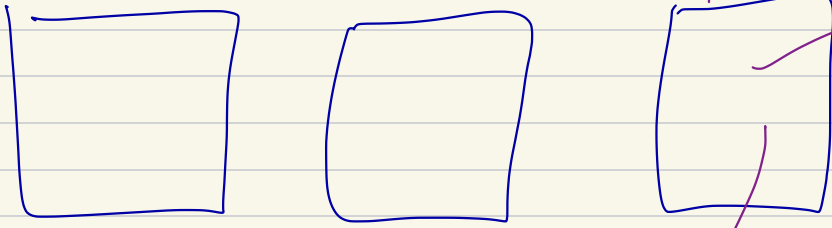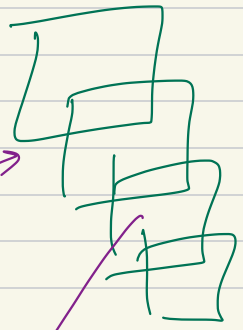   allocated to 1 quad Tree leaf
     node.

Client   driver

→ (driver-id, location)

Gateway

Global Cache

Quad Tree

+ update driver cell ☺

driver-id , latest live location

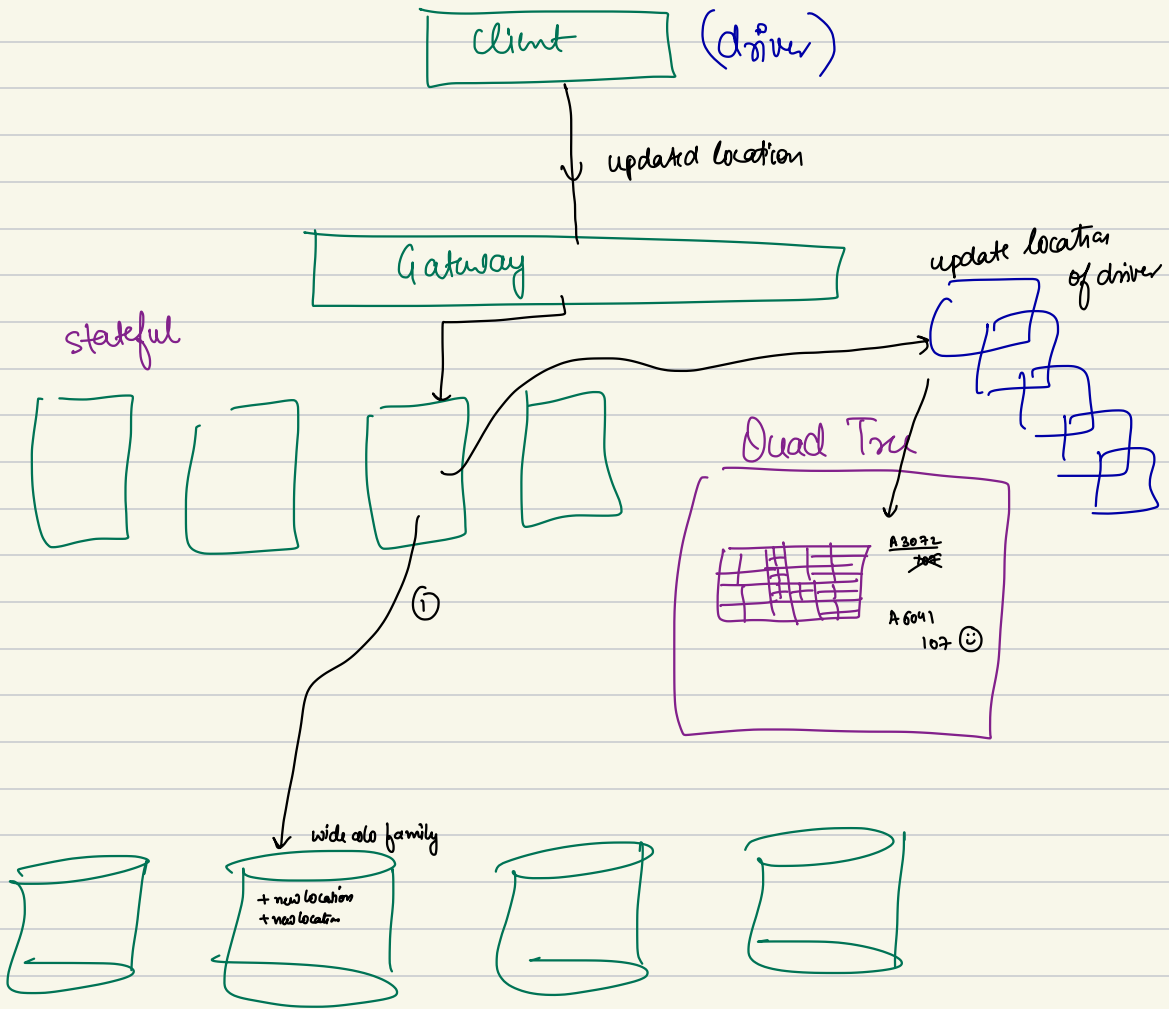| 107 | x_1, y_1 |
| 108 | x_2, y_2 |
| = | |

[we don't split / merge with
every driver cell update]

Update my quad Tree cell distribution

[cell merging + cell splitting]

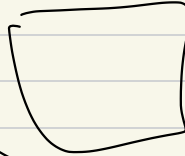once every 1-4 hours.

Uber - Mumbai Cluster

Client (driver)

| updated location
↓

Gateway → update location of driver

stateful

Quad Try

A 3072 ~~104~~

A 6041
107 ☺

①

wide colo family

+ new location
+ new location

Client   (user)

book
(location
    destination)

Gateway

Stateless

(x,y)
call id

Quad Tree

Find Drivers
call id

Question

How to find neighbourhood cells

strategy 1

(0,60)   (40,60)

v
(10, 40)

(0, 20)   (40, 20)
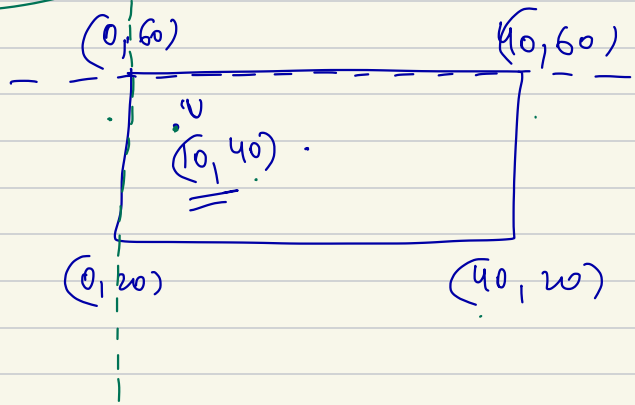
[10, 40]

                          10, (60-1) ⇒ (10, 59)

0(-1, 40)

Find (-1, 40)   →   B 1079 ☺

                          Find (10, 59) → B7091
                                          ☺

Strategy 2

Strategy 3

G 1079

B 1070
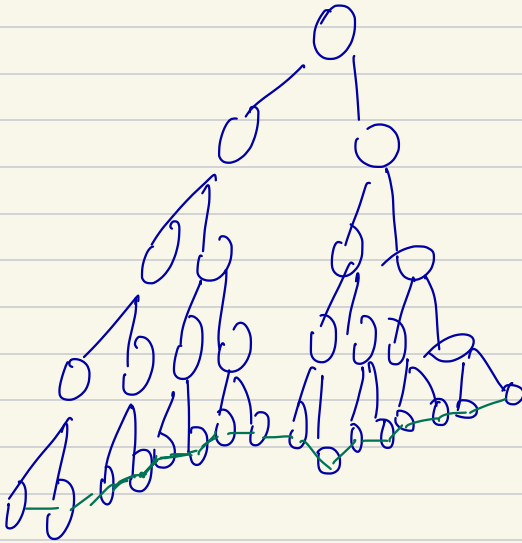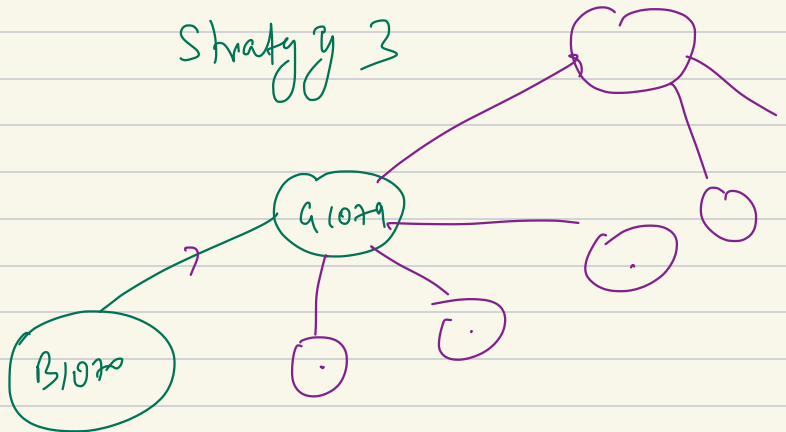
hm 1

cellid → location coordinates

hm 2

cell-id → children cells

hm 3

cell-id → parent-cell-id

hm 4

cell-id → [places]

Question

Drivers → update his everytime I get a new
driver location

driver-id   details   location   leaf-cell-id

107   ~~B1074~~ C7079
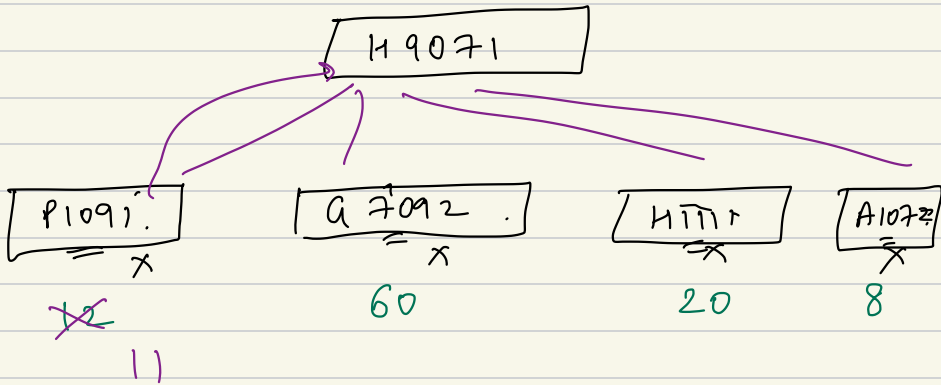
location

[leaf-cell-id   [.   ]]

[cell-id   →   Parent cell ids]

[

→ once every 1-4 hours.

Question — how to merge leaf nodes

esta semord (-)

| H 9 0 7 1 |

| P 1 0 9 1 | | G 7 0 9 2 | | H 1 1 1 r | | A 1 0 7 z |
= ✗   = ✗   ✗   ✗

✗   60   20   8

1)

evaluate the possibility of merging??

11 + 60 + 20 + 8 = 99

merge :)

| H 9 0 7 1 :✓ | (new leaf node)

99

mid x =

mid y =

x, y => user

if (x > midx)

{ if (y > mid y)

I st child

II nd child

.
;
;

}

}