Hashmap → ① usage → [Demo, question solve
② creation

Heap ① usage → Demo → question
② creation

① Demo of Hashmap :

key vs. value] → pair D.S.

D.S → get → O(1)
set → O(1)
retrieve. → O(1)

Functions in hashmap →

① put (key, value) → Key absent, add key-value pair
Key present, value update

② get (key): → Return value associated to key, if present, otherwise return null

③ containskey (key) → True / False

④ size () → no of keys

⑤ keySet () → Iterator, helps to iterate on keys of hmap.

key → If we want to use our userdefined class as key, then we have to override Hashcode function, so that uniqueness of key can be maintained.

Highest Frequency character :

String → "a b c b d e a z a a r a t a b t t a"

Character vs. Integer

a → ~~1~~ ~~2~~ ~~3~~ ~~4~~ ~~5~~ ~~6~~ 7
b → ~~1~~ 2
c → 1
d → 1
e → 1
z → 1
r → 1
t → ~~1~~ ~~2~~ 3
b → 1

top frequency character → a
→ 7

map.getOrDefault(key, defaultvalue);

Initially -
- map.getOrDefault(a, 0); → 0
- map.put(a, 0+1);
- map.getOrDefault(a, 0); → 1

a → 1

get common Element - 1 :

HashSet

↳ key

Value X

arr1 → 10 20 30 10 4 7 9 15 8

HashSet → [10 30 7 15 / 20 4 9 8]

put key)

add → O(1)

remove → O(1)

contains → O(1)

arr2 → 4 10 9 13 12 15 8 20 30 10

4 ✓ 10 ✓ 9 ✓ 13 ✗ 12 ✗

4 ✓
10 ✓
9 ✓
15 ✓
8 ✓
20 ✓
30 ✓

4
10
9
15
8
20
30

if present in HashSet

① print

② Remove from HashSet

10 → already printed ✗

get common Elements 2 :

arr1 → 10 7 3 10 4 3 10 9 6 9 4 1

arr2 → 4 10 8 10 10 9 5 13 11 17 10 10 10

X X X X

all 10's are Exhausted

output → 4, 10, 3, 10, 10, 9

} Intersection of arr1 and arr2

from arr1 ↓ map1

Frequency map from arr1

1 → 1

10 → ~~X~~ ~~2~~ ~~3~~ ~~2~~ ~~1~~ 0

7 → 1

3 → ~~X~~ ~~2~~ 1

4 → ~~X~~ ~~2~~ 1

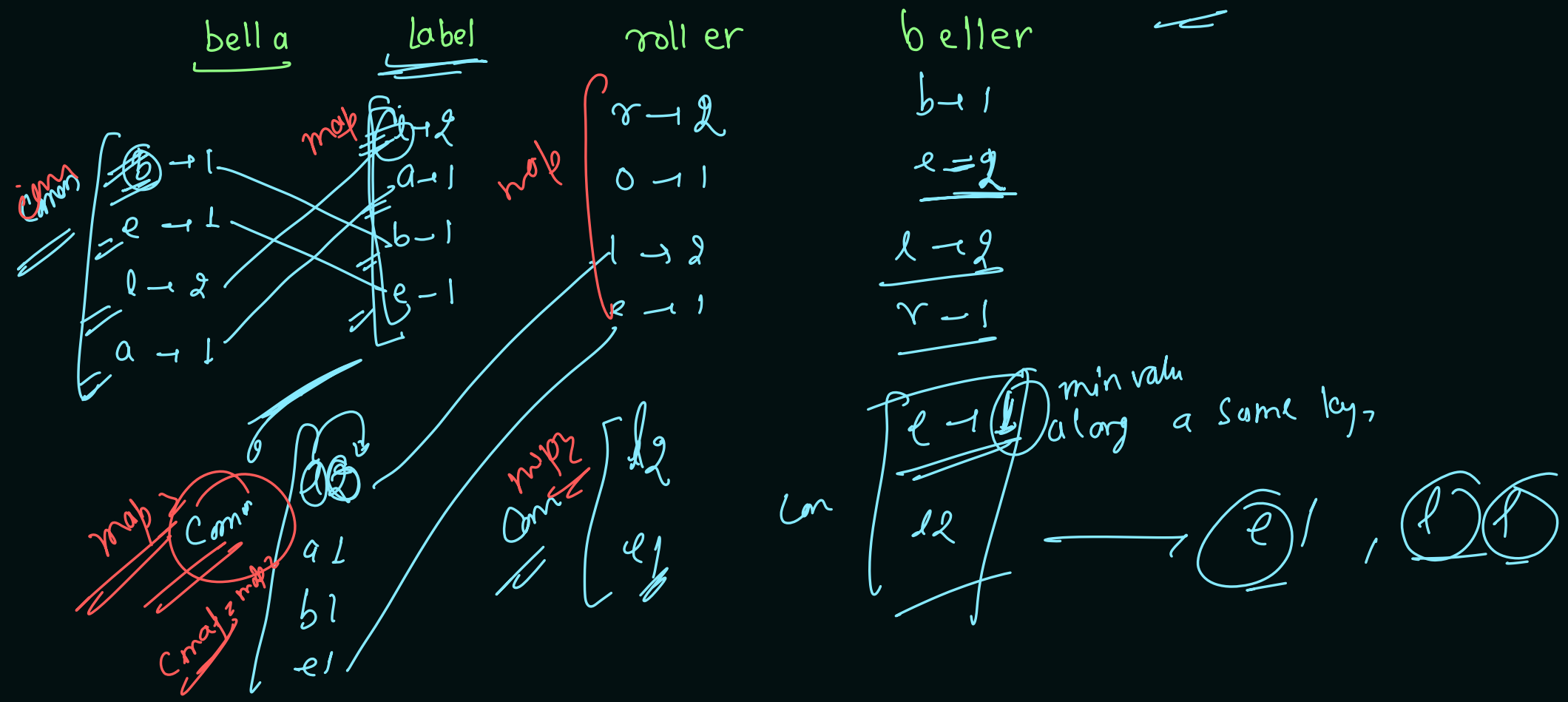
9 → ~~X~~ ~~2~~ 1

6 → 1

4, 10, 3, 10, 10, 9 } 2

if freq is more than 0 (equal to 1) → print val & decrease freq from map 1.

leetcode 1002



Longest consecutive Seq:

① figure out starting point of every consecutive seq.

12
5
1
2
10
2
13
7
11
8
9
11
8
9
5
6
11

key — Value

Element — Boolean

map

12 — F
5 — F T
1 — F T
2 — F
10 — F
13 — F
7 — F
11 — F
6 — F
9 — F
4 — F

5 → 6

1 → 2

5
6
7
8
9
10
11
12
13

:# How to figure out starting of seq.?

if element -1 is not exist in map then it is starting point