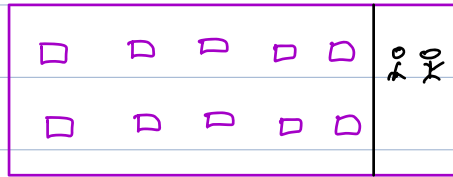


Hashmap Intro

Sachin Madhi

Hotel



5

Register

1

2

3

4

5

6

7

8

9

10

1000 rooms

Hotel



⊗

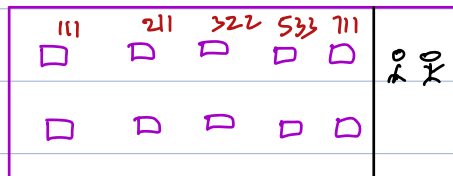
Array [1000] → 0 Vacant
→ 1 not Vacant

535 (check index)

Array [535]

lucky numbers

Hotel



$(1 - 10^6)$

rooms : 1000

range of room num: $(1 - 10^6)$

Array $[10^6]$

storage of array being wasted

111 →

3231 →

4746

6791

{ 111 : 1 ← not vacant

3231 : 1

4746 : 1

6791 : 1

HashMap

111

Key → value

Q1) Store population of every country

Country name \rightarrow key (String)
population of country \rightarrow value (long)

Q2) No. of states in each country

key \rightarrow country (String)
value \rightarrow no. of states (int)

Q3) For every country we want to know all states names

Advance

key \rightarrow country name (String)
value \rightarrow state names List<String>

(Implementation of HM)

key \rightarrow unique value
 \hookrightarrow primitive datatype (String, int, long, float, boolean)
char

value \rightarrow no constraint

HW

Q4) For every country store population of each state

HashMap

→ (K, V)

→ Key has to be unique

→ Key (primitive)

HashSet

(K)

Key has to be unique

Key (primitive)

HS: $\{1, 2, 3\}$
insert 2 ↗

Hashmap functionality

TC: $O(1)$

Advance

HM

HS (Implementation of HM)

size: No. of Keys in HM

insert (key, value)

search (key) \rightarrow T/F

get (key) \rightarrow value

delete (key) \rightarrow deletes (K,V) pair

update (key, new value)

size: No. of Keys in HS

insert(key) \rightarrow insert only if key is not present else ignores

search(key) \rightarrow T/F

delete (key) \rightarrow delete the key

update : Not present

"how to update in dictionary in python" Google

{# Hashing library names in different languages }

pseudocode

HashMap

HashSet

Java

HashMap

HashSet

python

dictionary

set

Q1) Find Frequency of Numbers

Given N array elements & Q queries for each query find frequency of each element in array

$$1 \leq N \leq 10^5$$

$$1 \leq Q \leq 10^5$$

$A[10] = \{ \begin{matrix} 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \\ 2 & 6 & 3 & 8 & 2 & 8 & 2 & 3 & 8 & 10 & 6 \end{matrix} \}$

$Q:$

findFrequency(A, Q)

2 : 3

3 : 2

4 : 0

10 : 1

Brute force :

Iterate on queries array

→ for each query count no. of times an ele is occurring

TC: $O(NQ)$

SC: $O(1)$

Element → freq of ele

{ 0 1 2 3 4 5 6 7 8 9 10 }

2 6 3 8 2 8 2 3 8 10 6 }

↑

hm = {}

for (i=0; i<N; i++) {

if (hm.search(A[i]) == True) {

value = hm.get(A[i])

hm.update(A[i], value+1)

}

else {

hm.insert(A[i], 1)

}

}

TC: $O(N+Q)$

SC: $O(N)$

for (i=0; i<len(Q); i++) {

query = Q[i]

if (hm.search(query) == True) {

print(hm.get(query))

}

else {

print(0)

}

}

hm

2: 3

6: 2

3: 2

8: 3

10: 1

Microsoft

Q2) Find the first non-repeating element

$$A[6] = \{ 1 \quad 2 \quad 3 \quad 1 \quad 2 \quad 5 \}$$

$$A[8] = \{ \overset{R}{4} \quad \overset{R}{3} \quad \overset{L}{3} \quad \overset{NR}{2} \quad 5 \quad 6 \quad 4 \quad 5 \} : 2$$

$$A[7] = \{ \overset{R}{2} \quad \overset{NR}{6} \quad 8 \quad 4 \quad 7 \quad 2 \quad 9 \} : 6$$

$\{ 1, 1, 1, 2, 2 \} \rightarrow -1$ (because all elements are repeating)

Create a HM Not correct

Ans \rightarrow Iterate the HM and find first key whose value = 1

$$\{ \overset{R}{1} \quad \overset{R}{2} \quad \overset{NR}{3} \quad 1 \quad 2 \quad 5 \}$$

\longrightarrow

HM

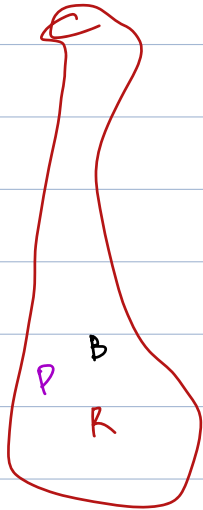
1: 2

2: 2

3: 1

5: 1

HM is unordered



Sack is unordered

HM

B: 2

A: 1

C: 3

D: 4

Create the HM

```
for (i=0; i<N; i++) {  
    if (hm.get(A[i]) == 1) {  
        return (A[i])  
    }  
    else {  
        # skip  
    }  
}  
return -1
```

Break (10:35 - 10:45)

Q4) Given $arr[N]$ elements find no. of distinct elements after removing duplicates.

$$A[5] = \{3, 5, 6, 5, 4\} \rightarrow \{3, 5, 6, 4\} : 4$$

$$A[5] = \{1, 1, 1, 2, 2\} \rightarrow \{1, 2\} : 2$$

$$A[3] = \{3, 3, 3, 3\} \rightarrow \{3\} : 1$$

$$\begin{array}{c} \downarrow \quad \downarrow \\ \{3\} : \text{len}(ns) = 1 \end{array}$$

```
for (i=0; i<N; i++) {  
    ns.insert(A[i])  
}
```

$\text{len}(ns)$

Google / Amazon / Uber

Q4) Given an N elements, check if there exists a subarray with sum = 0

$A[0] = \{ \overset{0}{2}, \overset{1}{2}, \overset{2}{1}, \overset{3}{-3}, \overset{4}{4}, \overset{5}{3}, \overset{6}{1}, \overset{7}{-2}, \overset{8}{-3}, \overset{9}{2} \}$

True

$A[0] = \{ \overset{0}{2}, \overset{1}{2}, \overset{2}{1}, \overset{3}{-3}, \overset{4}{4}, \overset{5}{3}, \overset{6}{1}, \overset{7}{-2}, \overset{8}{-3}, \overset{9}{2} \}$

Find all subarrays

TC: $O(N^3)$

Using prefix sum

Find all subarray sum

TC: $O(N^2)$

$A[0] = \{ \overset{0}{2}, \overset{1}{2}, \overset{2}{1}, \overset{3}{-3}, \overset{4}{4}, \overset{5}{3}, \overset{6}{1}, \overset{7}{-2}, \overset{8}{-3}, \overset{9}{2} \}$

$PF[0] = \{ 2, 4, 5, 2, 6, 9, 10, 8, 5, 7 \}$

$pf[2] = 5$

$sum(A[0:2])$

$pf[8] = 5$

$sum(A[0:8])$

$$\underline{\text{sum}(A[0:8])} = \text{sum}(A[0:2]) + \text{sum}(A[3:8])$$

$$5 = 5 + \text{sum}(A[3:8])$$

$$\text{sum}(A[3:8]) = 0$$

$$A[10] = \left\{ \begin{array}{c} 0 \\ 2 \end{array} \right. \left[\begin{array}{ccc} 1 & 2 & 3 \\ 2 & 1 & -3 \end{array} \right] \begin{array}{ccccccccc} 4 & 5 & 6 & 7 & 8 & 9 \\ 4 & 3 & 1 & -2 & -3 & 2 \end{array} \right\}$$

$$PF[10] = \left\{ \begin{array}{cccccccccc} 0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \\ 2 & 4 & 5 & 2 & 6 & 9 & 10 & 8 & 5 & 7 \end{array} \right\}$$

$$pf[0] = 2$$

$$\text{sum}(A[0:0]) = 2$$

$$pf[3] = 2$$

$$\text{sum}(A[0:3]) = 2$$

$$\text{sum}(A[0:3]) = \text{sum}(A[0:0]) + \text{sum}(A[1:3])$$

$$2 = 2 + \underline{\quad}$$

$$A = \left[\begin{array}{cccccc} 2 & 2 & 1 & -3 & 1 & 1 \end{array} \right]$$

$$pf = \left[\begin{array}{cccccc} 2 & 4 & 5 & 2 & 3 & 4 \end{array} \right]$$

correct

TC: $O(N)$ 1) Find pf array \rightarrow if $pf[0] = 0$ return true

TC: $O(N)$ 2) Check for duplicates in the pf array

TC: $O(N)$

SC: $O(N)$

$\{1, 1, 2, 3\}$ len: 4

\downarrow HS

$\{1, 2, 3\}$ len: 3

Put everything in a hs

if $(\text{len}(\text{hs}) < \text{len}(A))$

there is subarray with sum 0

else:

No subarray with sum 0

$$A = \begin{bmatrix} 2 & 1 & 3 & -6 & 7 \end{bmatrix}$$

$$1f: \begin{bmatrix} 2 & 3 & 6 & 0 & 7 \end{bmatrix}$$

$$\text{sum}(A[0:3]) = 0$$