Key - Value

int

String

List

Ky

MP

UP

P

100K

Virat ⇒ [80, 0, 101...]

BatsmenId Scr

1

2

3

1003

2136

[ ]

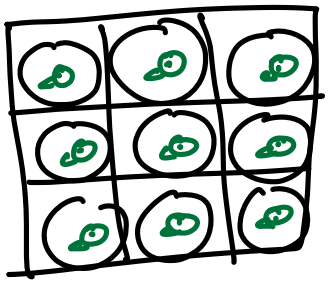
[ ]

2136

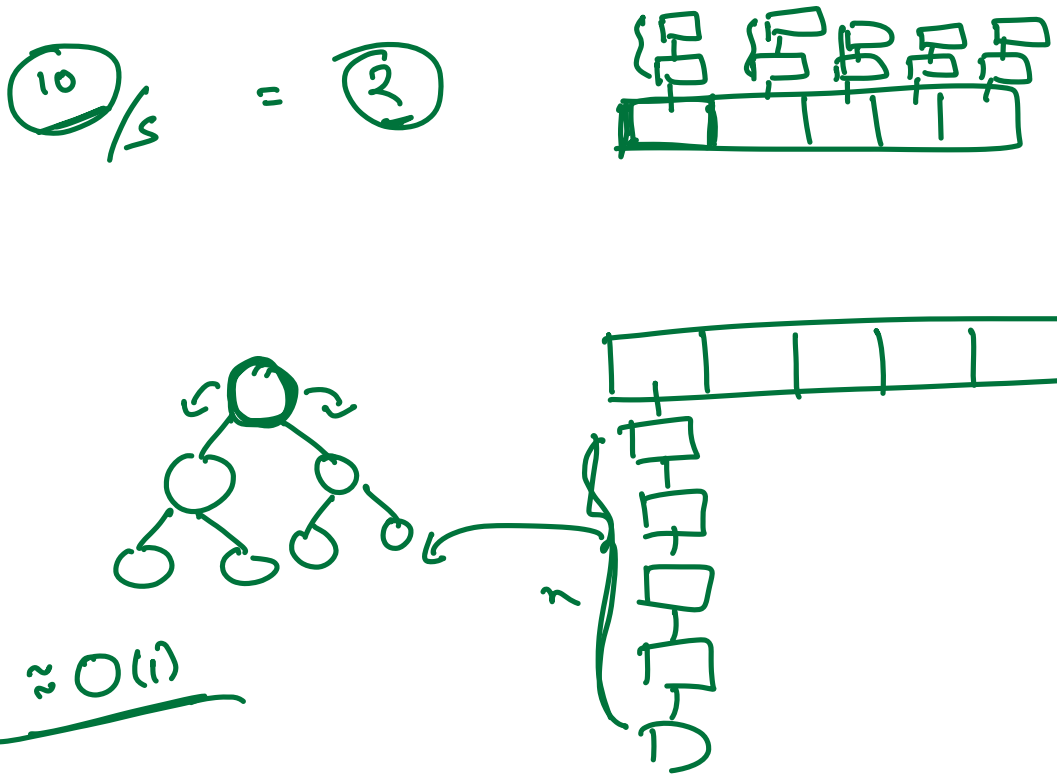
0 1 2 3 4 5 6 7

hf(2136) ⟶ 6

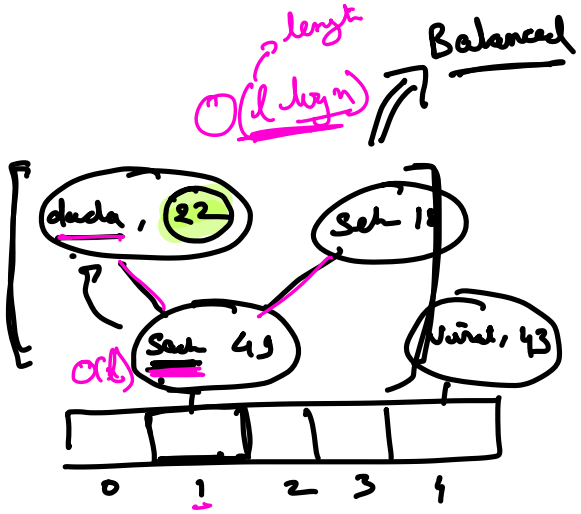
Pegion Hole



10



String	Val
Sachin	49
Virat	43
Shikhar	18
Dada	22
Sehwag	18



hf(Sachin) = 1

hf(Virat) = 4

hf(dada) = 1

hf(Sehw) = 1 ⇒ O(1)

a aaaa a
1 1 1 1 1
a a a a a

Sachin
1 1
Sehwag

HashCode of Strings

a ⇒ 97
A ⇒ 65
B ⇒ 66
C ⇒ 67
⋮

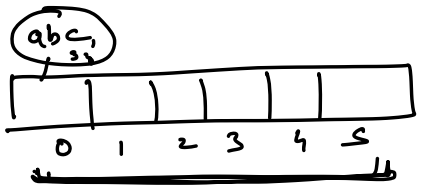
} ASCII

ABC → [65, 66, 67]

A + B + C = (N) % size

Assume

a : 1
b : 2
c : 3
d : 4
e : 5
⋮
z : 26



size: 6

abc : 38

hf(abc) = (a + b + c) % 6
= 6 % 6
= 0

$hf(abc) == hf(bca) == hf(cab) == hf(bbb)$

$hf(abc) = a \times p^0 + b \times p^1 + c \times p^2$ % Size (31)

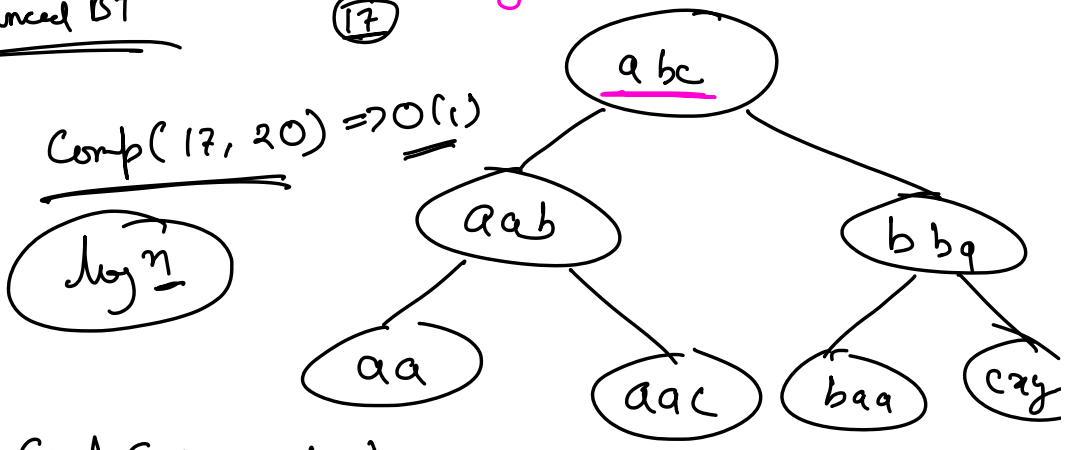
bac
cba
acb

$hf(bca) = b \times p^0 + c \times p^1 + a \times p^2$

$hf(S) = \sum_{i=0}^{n} S[i] \times p^i$ → Rabin Karp ★

Balanced BT (17)

xyz



Comp(17, 20) ⇒ O(1)

$\log n$

Comp(xyz, abc) ⇒ $O(l)$ ← length of strings

$\approx O(l \log n) + O(l)$ → Hashcode

Q Given an array.

Return true if there is a pair whose sum is also present in the array.

[3, 8, 2, 1, 9, 16, 12, 20]

= True

[5, 100, 20]

= False

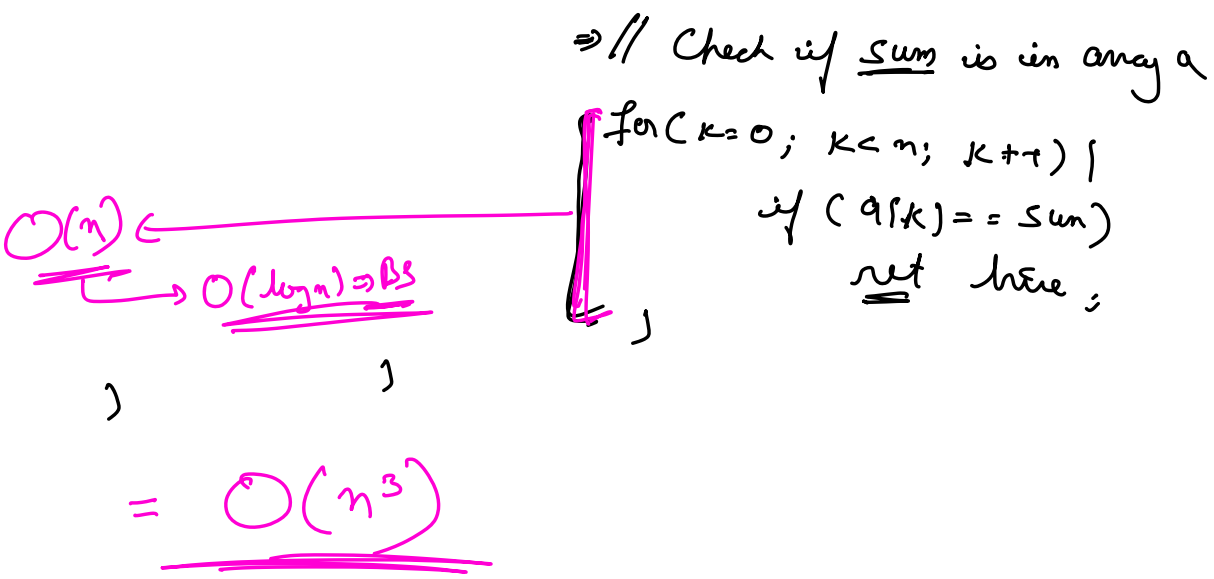
[2, 3, 5, 6]

2, 3 3, 5 5, 6

2, 5 3, 6

2, 6

```
O(n) ← for (i=0; i<n; i++){
    O(n) ← for (j=i+1, j<n; j++) {
            sum = a[i] + a[j];
```



⇒ add all val in a Set / Map

⇒ loop over pairs & check if the sum is present in set / Map.

```
for (i=0; i<n; i++) {
    set.add(a[i])
}
```
⇒ O(n)

```
for (i=0; i<n; i++) {
    for (j=i+1; j<n; j++) {
        sum = a[i] + a[j];
    }
}
```
→ O(1) addition

⇒ O(N²)

```
        if ( set. contains (sum))
            ret true;
    }
}
```

$O(N) + O(N^2)$

$= O(N^2)$

Q

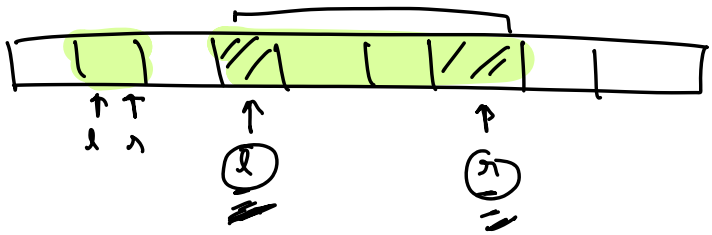
Amazon
MS
Adobe
Ola
Myntra

Given an array.
Return true if there exists a subarray whose sum = 0.

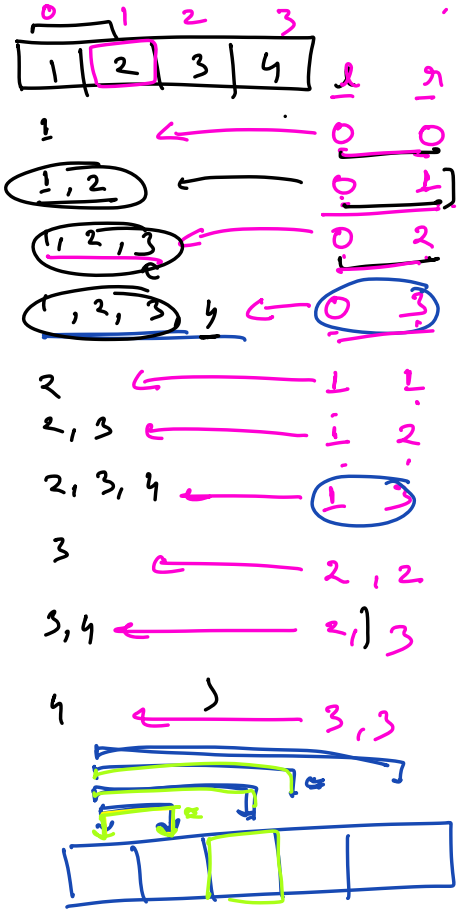
[2, 1, 3, -4, 0, -2, 10, -7]

[1, 8, 3, 2, -5, 6, -17]

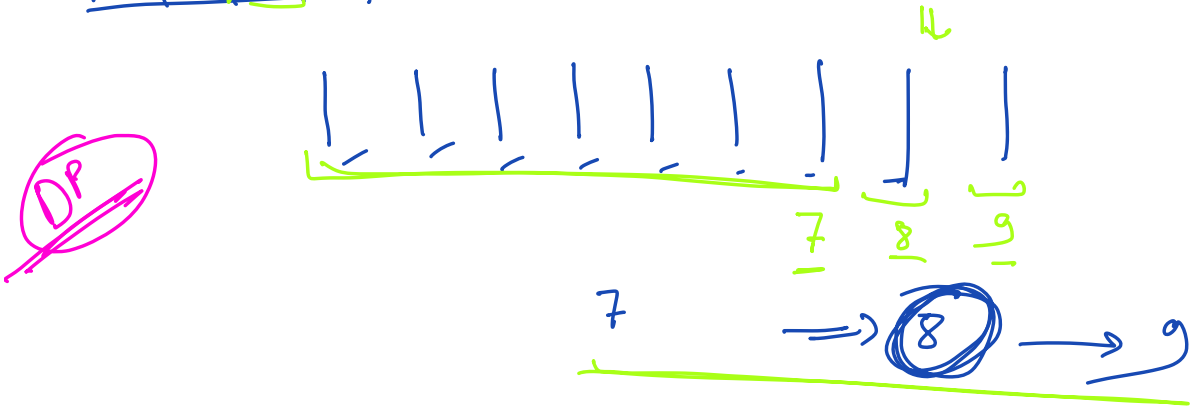
[-17, 100, 200, 17] ×

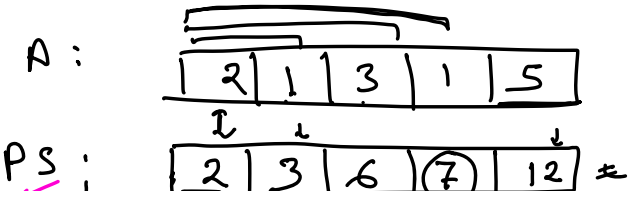


```
for (l=0; l<n; l++) {
    for (r=l; r<n; r++) {
        // l, r
        sum = 0;
        for (i=l; i<=r; i++) {
            sum = sum + a[i];
        }
        if (sum == 0) {
            ret true;
        }
```



TC: $O(N^3)$



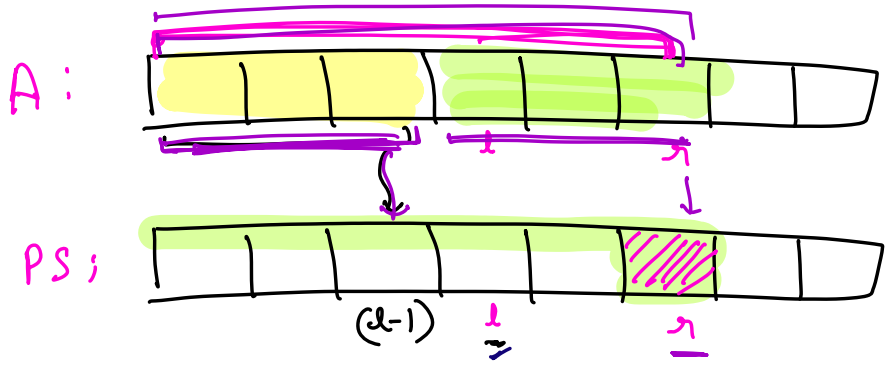


$PS[i] \Rightarrow$ Sum of elements till index i

$$\sum_{x=0}^{i} A[x]$$

```
PS[0] = A[0];
for (i = 1; i < n; i++) {
    PS[i] = PS[i-1] + A[i];
}
```

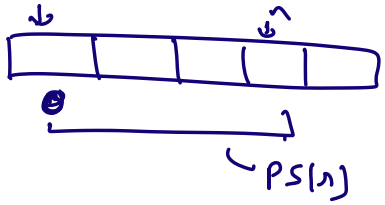
O(N)

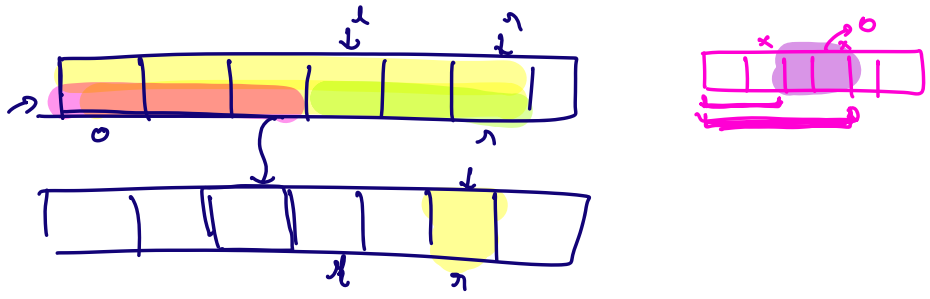


$PS[r] = PS[l-1] + Sum(l-r)$

$\Rightarrow$ $Sum(l-r) = PS[r] - PS[l-1]$. if $l \neq 0$

What if $l = 0$?

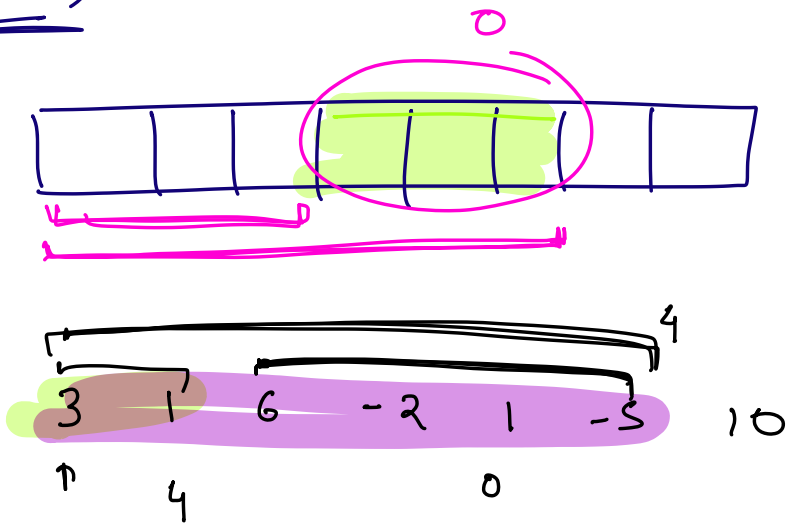




$$\Rightarrow \quad PS[r] = PS[l-1] + sum[l-r]$$

$$\Rightarrow \quad \boxed{PS[r] - PS[l-1] = sum[l-r]} \qquad O(1)$$

$O(N^2)$



$$\boxed{sum[l-r]} = PS[r] - PS[l-1]$$

$$0 = PS[r] - PS[l-1]$$

$$\boxed{PS[r] = PS[l-1]}$$

⇒ Find if a number repeats in the PS

Set / Map

① Create PS

② Insert all PS[i] in Set/Map

③ While inserting check if already present.

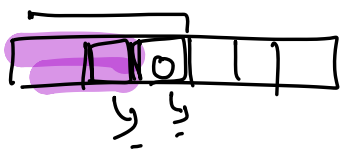
```
PS[0] = A[0] ; Set.add(A[0]);
for(i = 1; i < n ; i++) {
    PS[i] = PS[i-1] + A[i];
    if(PS[i]==0 || Set.contain(PS[i]))
        ret true;
    }
    Set.add(PS[i]);
}
ret false
```

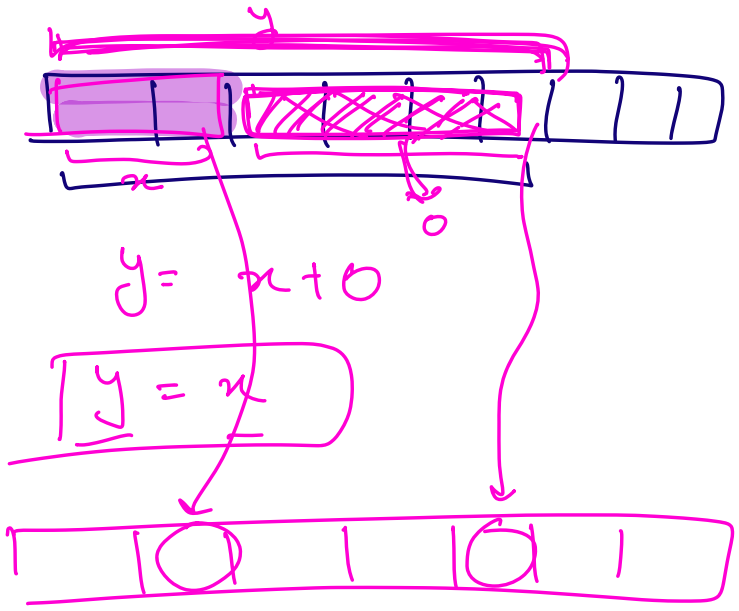
O(n)

A: 2, -1, -1, 8, 9, 3

PS: 2, 1, 0, 8, 17, 20

2, 1, 0, 8, 17, 20



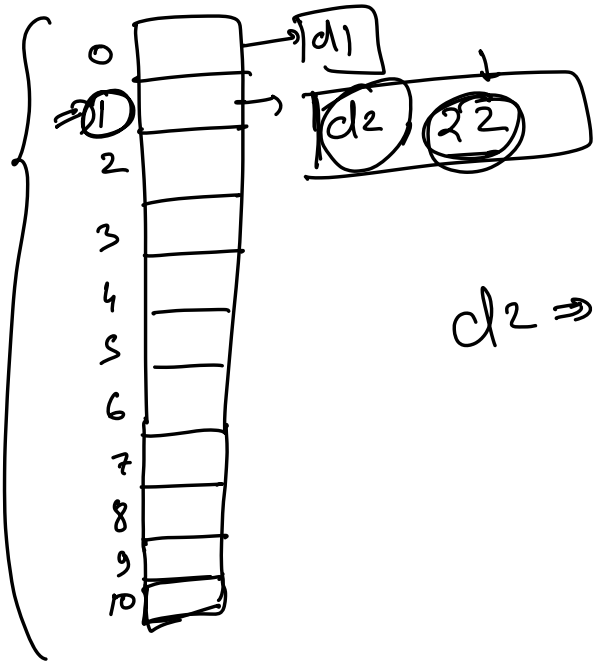


---

21 % 7 ⇒ 0

22 % 7 ⇒ 1

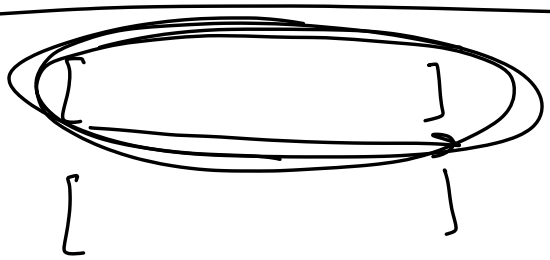
22 % 11 ⇒ 0



d2 ⇒ 22

Consistt Hashy

(HD2)



: x

: y

[2, 2, 1, 1, 0] : z

HashMap

Key	Cnt.
2	2
1	2
0	1

HashMap <Integer, Integer>

if