



Agenda

- ① Employee and Manager
- ② Problem with a given diff
- ③ Array fair Divisible by K
- ④ Fair Sum Divisible by k
- ⑤ longest Subarray with zero sum
- ⑥ Equilibrium Index.

Employee And Manager.

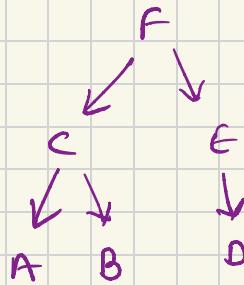
Q1

Map o

key (Emp) Value (Mngr)

A	C
B	C
C	F
D	E
E	F
F	F

Org chart.

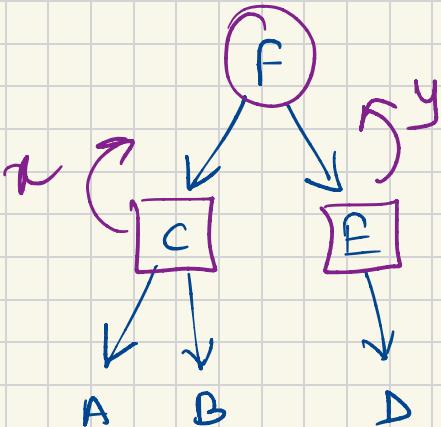


Q2

Map

Key	Value
A	0
B	0
C	2
D	0
E	1
F	5

}



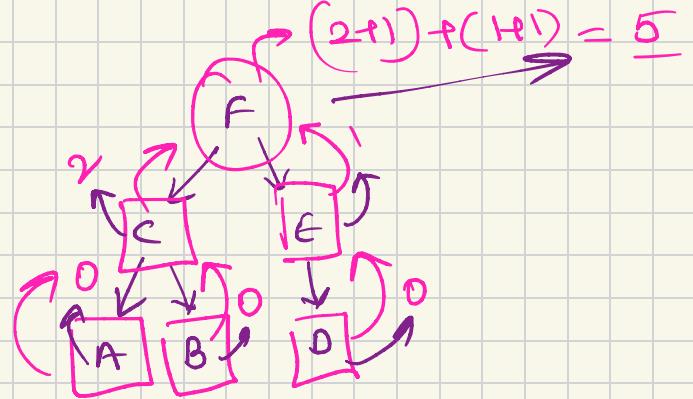
$$\sum i_F = (x+1) + (y+1)$$

No. of Emp Under C

No. of Emp Under E

find No. of Emp Under (string emp)

$$\text{No. of Emp Under} = \sum [(\text{No. of Emp Under two direct reportee}) + 1]$$



Map

Direct reportee

$$C \rightarrow \{A, B\}$$

$$E \rightarrow \{D\}$$

$$F \rightarrow \boxed{\{C, E\}}$$

F
↑

fun(string E)

```

    {
        Put total = 0;
        for (string Emp : Map.get(F))
        {
            total = fun(Emp) + 1;
        }
    }
  
```

Steps

- { ① Map of Direct Reportees
- ② Do recursion to calc. no. of Emp.
 → Store in Map.

key (Emp) Value (Mngr)

A	C
B	C
C	F
D	E
E	F
F	F



Key
Mngr

C

F

E

Value
List of Emp.

{A, B}

{C, E}

{D}

```

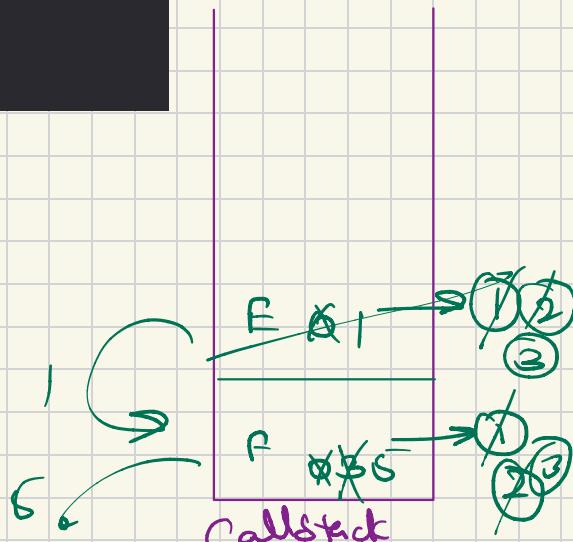
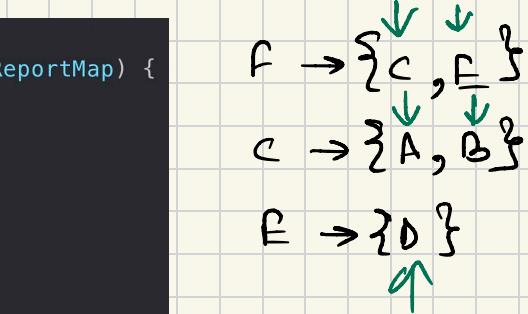
// faith: returns number of employees under this employee emp
public int getEmpUnder(String emp, HashMap<String, ArrayList<String>> directReportMap) {
    if (directReportMap.containsKey(emp) == false) {
        System.out.println(emp + " -> " + 0);
        return 0;
    }

    int totalCount = 0;
    for (String directs : directReportMap.get(emp)) {
        int count = getEmpUnder(directs, directReportMap);
        // number of employees under this direct and +1 for him
        totalCount += (count + 1);
    }

    System.out.println(emp + " -> " + totalCount);
    return totalCount;
}

```

A → 0
 B → 0
 C → 2
 D → 0
 E → 1
 F → 5

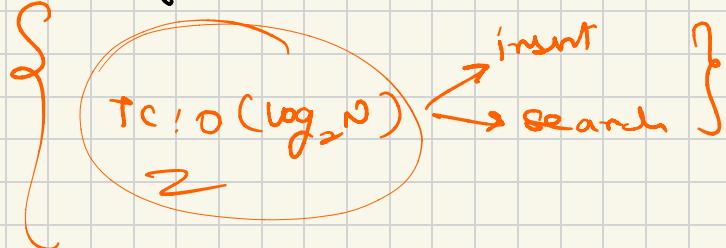


✓ Tree Map

✓ TreeSet

Method

- o Keys are always in a sorted Order



Problem with a given difference.

$$\text{int}[] \text{arr} = \{ 5, 10, 3, 2, 50, 80 \} \quad B = 78$$

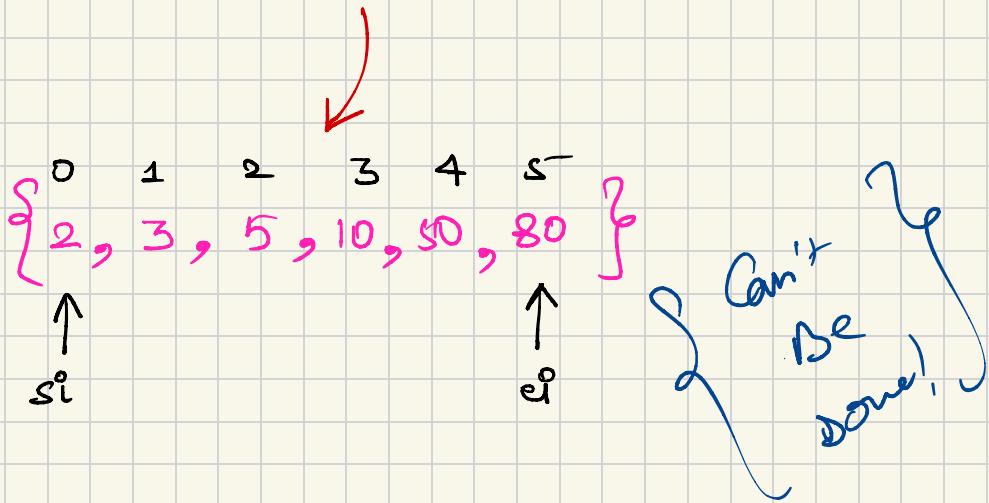
Brute force

o Nested Loops

$$\xrightarrow{i} \left. \begin{array}{l} \text{arr}[i] - \text{arr}[j] \leq B \\ \text{arr}[j] - \text{arr}[i] \geq B \end{array} \right\}$$

return true

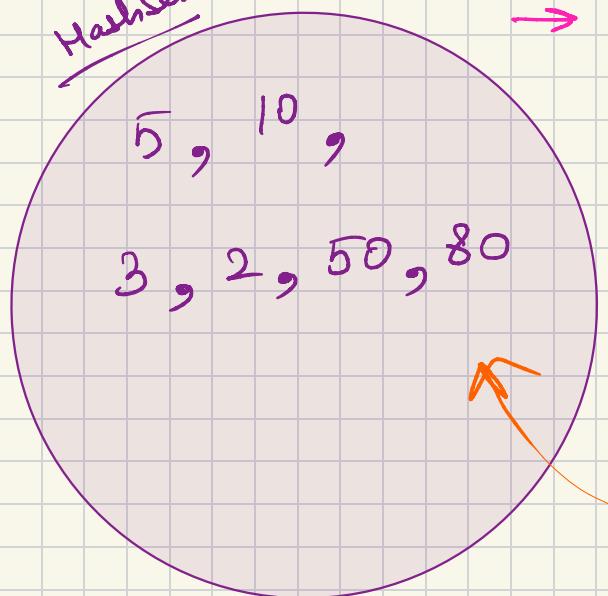
`int[] arr = { 5, 10, 3, 2, 50, 80 } B = 45`



$\text{int[]} arr = \{ 5, 10, 3, 2, 50, 80 \}$ $B = 78$

$$(x > y) \Rightarrow x - y = B \quad \text{or} \quad y - x = B$$

Handled



$$y = x - B \quad \text{or} \quad y = B + x$$

$$\begin{aligned} Tc(O(N)) + O(N) \\ = O(N) \\ SC(O(N)) \end{aligned}$$

$$\begin{cases} y = -73 \\ y = 82 \end{cases}$$

$$\begin{cases} y = -75 \\ y = 81 \end{cases}$$

$$\begin{cases} y = -76 \\ y = 80 \end{cases}$$

True

Array Pair Divisible by K.

int[] arr = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10} K = 5

even length

(N/2) pairs that, where sum of each pair is divisible K

* you can use each ele only once.

(9,1) (2,8) (7,3) (4,6) (10,5) }
↓ ↓ ↓ ↓ ↓
10 10 10 10 15

True | False
↓

Can it be alone
or not }

$\text{int[]} arr = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$ $K = 5$

pair (x, y)

$$x + y = \alpha K$$

$\left. \begin{array}{l} \\ \end{array} \right\} \text{sum should be multiple of } K$

$$\begin{array}{c} x = qK + r \\ \hline r \end{array}$$

$0 \leq r < K$

$$(q_1 \times k + r_1) + (q_2 \times k + r_2) = \alpha k$$

$$k(q_1 + q_2) + (r_1 + r_2) = \alpha k$$

$$0 \leq r_1, r_2 < k$$

divisible by k

make sure it is divisible by k

$$r_1 + r_2 = 0$$

$$\text{or } r_1 + r_2 = k$$

o Sum of any two no. can be divisible by K,
 if sum of these rem = 0 or K }
 {

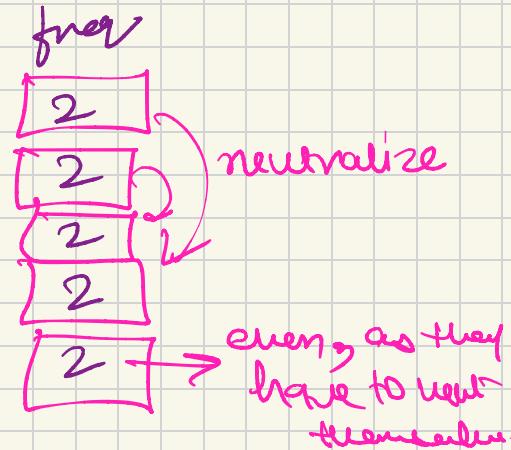
int[] arr = { 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 } K = 5
 rem 1 2 3 4 0 1 2 3 4 0

$$r_1 + r_2 = K$$

$$r_2 = K - r_1$$

rem

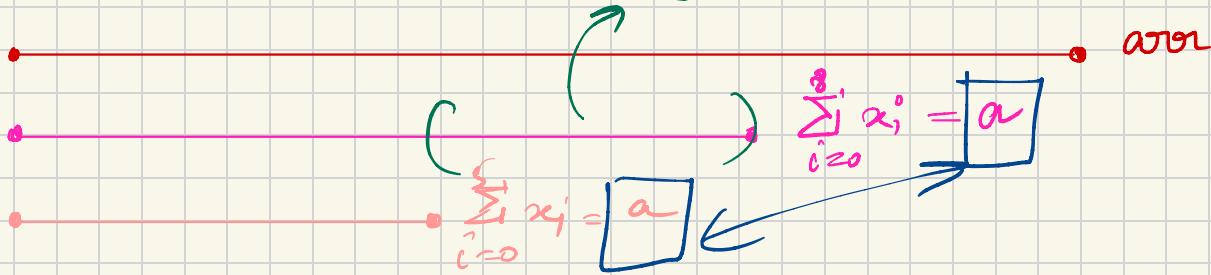
✓	1
✓	2
✓	3
✓	4
✓	0



Largest Subarray with Sum equal to zero

$$\text{int arr} = \{2, -2, 2, -8, 1, -1, 10, 23\}$$

Subarray with sum = 0



$$x_1 + x_2 + x_3 + x_4 + x_5 + x_6 + x_7 + x_8 = a$$

$$\downarrow a$$

$$\downarrow 0$$

`int[] arr = { 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 };`

0 2 0 2 -6 -5 2 12 35

ans = ~~X~~ ~~X~~ ~~X~~
5

HashMap

$$1 - (-1) = \boxed{2}$$

$$6' 0 = 5$$

$$2 - 0 = \cancel{\cancel{2}}$$

sym

0

2

-6

-5

12

35

incon

-1

0

3

2

6

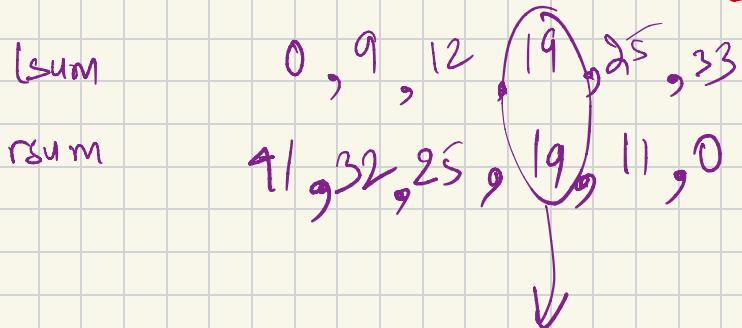
-7

Equilibrium Index

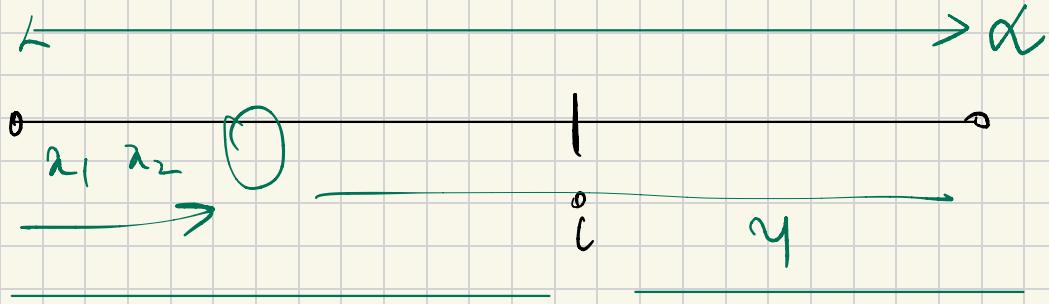
$\text{arr}[] = \{ 9, 3, 7, 6, 8, 11 \}$

Equilibrium Index: An index, where sum of elements on left = sum of elements on right

$\text{arr}[] = \{ 9, 3, 7, 6, 8, 11 \}$



eq^{index}



A diagram illustrating vector addition and subtraction. A parallelogram is formed by vectors α and γ . The top side of the parallelogram is labeled $\alpha = \beta + \gamma + \alpha_i \gamma_j$. The bottom side is labeled $\beta = \alpha - \beta - \alpha_i \gamma_j$.