2 questions :

1. Sum the difference
2. Sum of all subarrays

Q. Given an integer array of size N

↓

subsequences = 2^N

↓

Sum { For every subsequence, difference of largest no   smallest no }

Example: 3 5 10

	l	s	diff
	0	0	0
3	3	3	0
5	5	5	0
10	10	10	0
3 5	5	3	2
3 10	10	3	7
5 10	10	5	5
3 5 10	10	3	7
			Σ 21

(Sum of all max of each sub-sequence) − (Sum of all min of each sub seq)

$(0+3+5+10+5+10+10+10) - (0+3+5+10+3+3+5+3)$

$= 53 - 32 \longrightarrow 21$

Overview: How to solve this q

Example: 3 5 10

	l	s	diff
3	3	3	0
5	5	5	0
10	10	10	0
3 5	5	3	2
3 10	10	3	7
5 10	10	5	5
3 5 10	10	3	7
			$\sum$ 21

$$\underbrace{(3+5+10+5+10+10+10)}_{\sum \text{large}} - \underbrace{(3+5+10+3+3+5+3)}_{\sum \text{small}}$$

$$(53) - (32)$$

$$= 21$$

answer of this question =

$$\text{Sum}\left[\text{For every subsequence, largest no}\right] - \text{Sum}\left[\text{For every subsequence, smallest no}\right]$$

Code:

```
int    Sum_max_sub (int [] ar    ) {
           int N = ar.length
           int sum = 0
           Arrays.sort (ar )

           for ( i = 0    i < N    i++) {
                  sum = sum + ar[i] × $2^{i}$

           }
           return sum
}
```

Sum of min of subsequences

Code:

```
int    Sum_min_sub (int [] ar    ) {
           int N = ar.length
           int sum = 0
           Arrays.sort (ar )

           for ( i = 0    i < N    i++) {
                  sum = sum + ar[i] × $2^{N-i-1}$

           }
           return sum
}
```

Obs:

$$\begin{array}{cccccc} & & & i & & \\ & & & \downarrow & & \\ 0 & 1 & 2 & 3 & 4 & 5 \\ 2 & 4 & 5 & 7 & 8 & 10 \end{array}$$

$\downarrow$

min

$\boxed{7}$

2

7 ___

$\boxed{7}$ 7 8

7 10

7 8 10

8 10

1. $\underbrace{i+1 \quad i+2 \dots N-1}$ $\rightarrow$ elements

$(N-1) - (i+1) + 1$

$N-1 - i \cancel{-1} \cancel{+1}$

$N - i - 1$

$N = 6$

$i = 3$

$6 - 3 - 1$

$= 2$

$2^2 = 4$

2^{N-i-1} $\Big\}$ arr[i] min

Final Code :

```
int sum _ sub (int [] ar ) {
    int N = ar.length
    max_sum = 0
    Arrays. sort ( ar )
    int MOD = 1000000007

    for ( i = 0   i < N   i++) {
max_sum = (max_sum + (ar[i] x 2^i) % MOD) % MOD
min_sum = (min_sum + (ar[i] x 2^(N-i-1)) % MOD) % MOD
    }

    return max_sum - min_sum
}
```

Doubt :

2^i → Math. pow(2, i) ✓

→ (1 << i)

8 → 2^3 → 1000

1000 (1 << 3)

Q. Sum of all subarrays

```
long    Subsums (int[] arr    ){
long    int N = arr.length
        total = 0
        for (  i = 0      i < N     i++) {
            long occ   =   (i+1) * (N-i)
            long contri =   arr[i] * occ

            long total  =   total + contri
        }
        return total
}
```

TC: O(N)

SC: O(1)

Constraints: $1 <= N <= 10^5$

$1 <= arr[i] <= 10^4$

$(i+1) \times (\underset{100}{N}-i)$

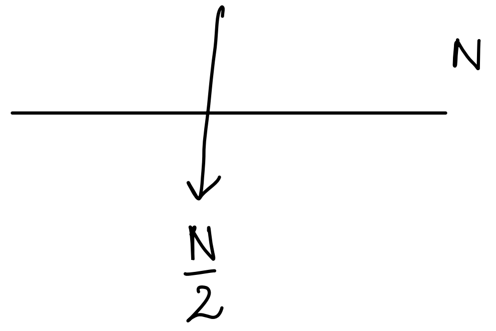
$\underline{N=100}$

$$\underline{(i+1) \times (N-i)}$$

$\underline{i=70}$ $(71) \times (30) \approx 2100$

$i=50$ $(51 \times 50) \approx 2500$

i → mid value



$$\left(\frac{N}{2}+1\right)\left(\underbrace{\underline{N} \; - }_{\to\,\frac{N}{2}}\frac{N}{2}\right)$$

$$\left(\frac{N}{2}+1\right)\left(\frac{N}{2}\right) \approx \underline{N^2}$$

$$\downarrow$$

$$\underline{10^{10}}$$

$$\downarrow$$

long

$N \longrightarrow 10^5$