# Q) GSS1 (SPOJ)
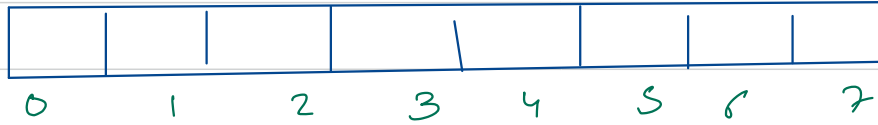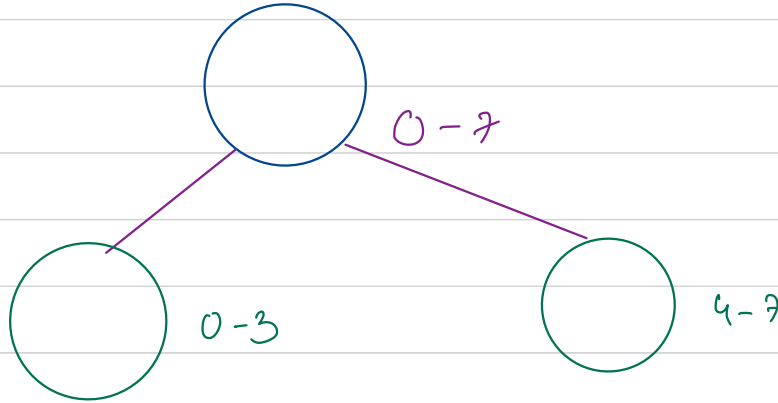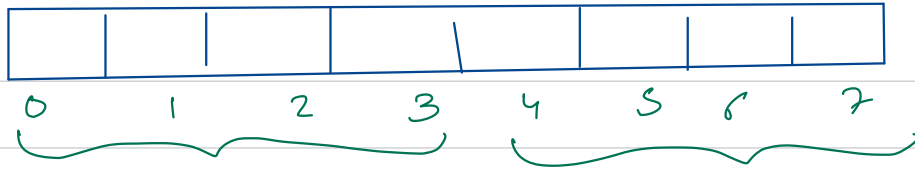
Segment tree

$q \rightarrow (x, y)$
$\downarrow$
maximum
sum
subarray
in this
range
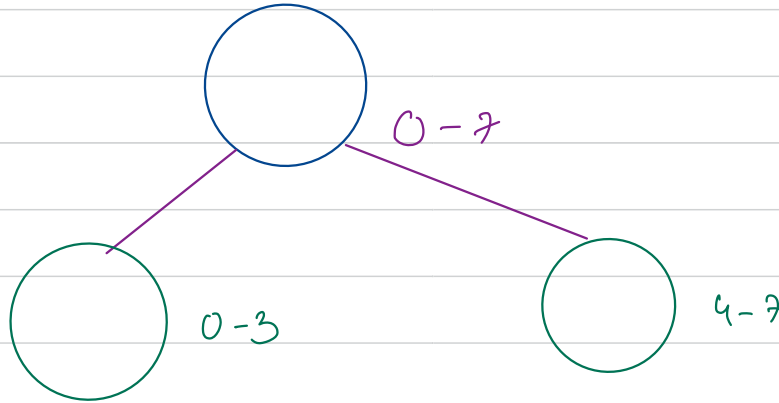
To get ans of just LST or just RST, we need max sum sub of LST & RST

0 - 7

0 - 3

4 - 7

$$\left\{ \begin{array}{l} \text{maxsum} \\ \text{maxprefix} \\ \text{maxsuffix sum} \\ \text{Sum of segment} \end{array} \right.$$

$$\text{maxsum} = \left\{ \begin{array}{l} \text{maxsum} - \text{lst}, \quad \text{maxsum} - \text{RST}, \\ \text{maxsum suffix} - \text{lst} + \text{maxsum prefix} - \text{RST} \end{array} \right\}$$

1 → for any parent segment, our max sum sub. can be coming either from just the LST or just the RST or both.

# Q → 2v3 (Hacker earth)

Type 0 → l, r → decimal value of string $[l..r]$ %3

Type 1 → make 0 → 1
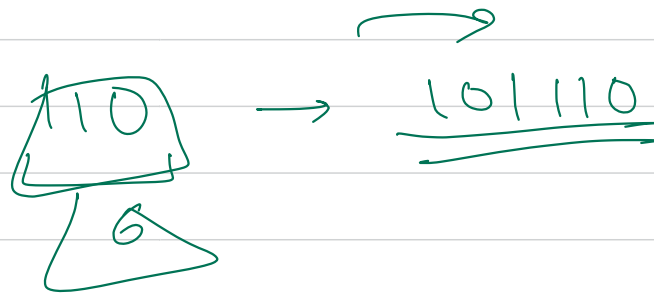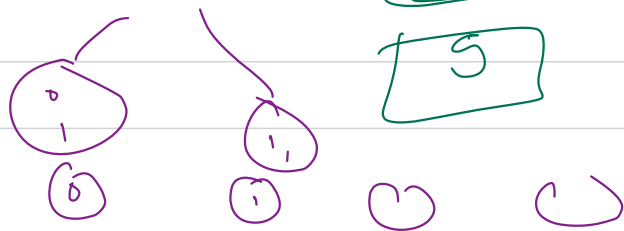
| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | | | | | | | |

$(a+b) \% c \rightarrow (a\%c + b\%c) \%c$

$(a-b) \%c \rightarrow (a\%c - b\%c + c) \%c$

$(a*b) \%c \rightarrow (a\%c * b\%c) \%c$
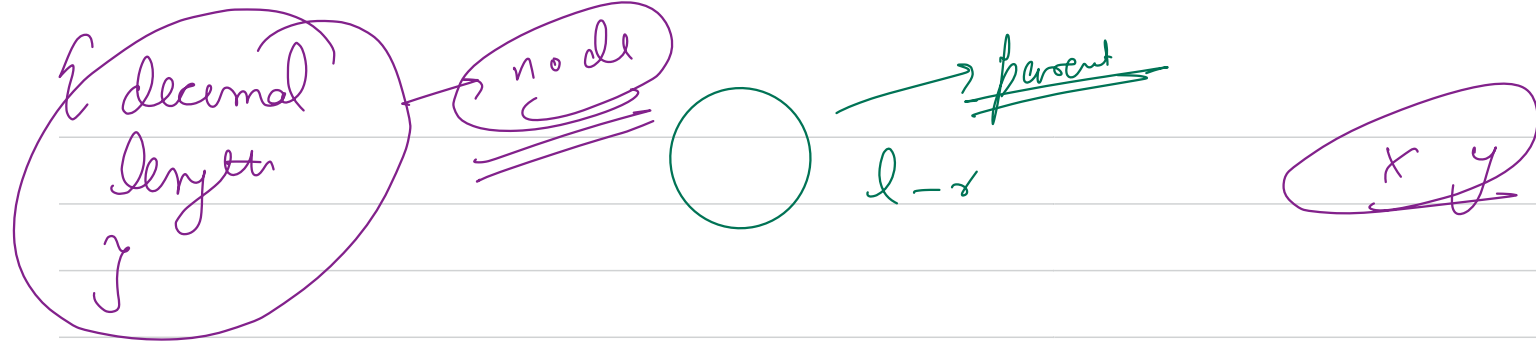
{ decimal length } → node

→ parent

$l - r$

x y

lST

KST

101

S

110 → 101 110

0

1

0

1

$110 \rightarrow 1 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 \longrightarrow x$

$101 \rightarrow 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 \longrightarrow y$

$101110 \rightarrow 1 \times 2^5 + 0 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 0 \times 2^0 \rightarrow z$

$z = \underset{\text{(elen-bin x)}}{2} \; x \; y \; + x$

$= 2^3 \left( 1 \times 2^2 + 0 \times 2^1 \times 1 \times 2^0 \right) + \qquad 1 \times 2^2 + 1 \times 2^1 + 0 \times 2^0$

$z \Rightarrow 1 \times 2^5 + 0 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 1 \times 2^1 + 0 \times 2^0$

**Q:→** You are given an integer array. We define a term, inversion **pair** : $\left(\begin{array}{c} a[i] > a[j] \text{ and} \\ i < j \end{array}\right)$

Given this array, for each element of the array, calculate the no. of inversion.

| 40 | 10 | 30 | 50 | 20 |
|----|----|----|----|----|
| 0  | 1  | 1  | 0  | 3  |

result →

$$\overset{2}{7} \quad 5 \quad 6 \quad 12 \quad 9 \nwarrow \longrightarrow \text{array}$$

| | 0 | 1 | 1 | 0 | 4 | → inversion |
|---|---|---|---|---|---|---|

(tree)

(~ve)

arr

| 0 | 0 | 1 | 0 | 0 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 |

$0 \longrightarrow$ value not visited

$1 \longrightarrow$ value visited

$O(n \times \log max)$

$\longrightarrow$ Type 1 $\longrightarrow$ update arr from $0 \rightarrow 1$

Type 2 $\longrightarrow$ get sum of elements after d

**Q.** Given an array, you have to apply 2 queries-

a) Set $(i, v)$      $arr[i] = v$

b) first_above $(x) \rightarrow$ find the first item greater than or equal to $x$. Return the value.

what if we are given
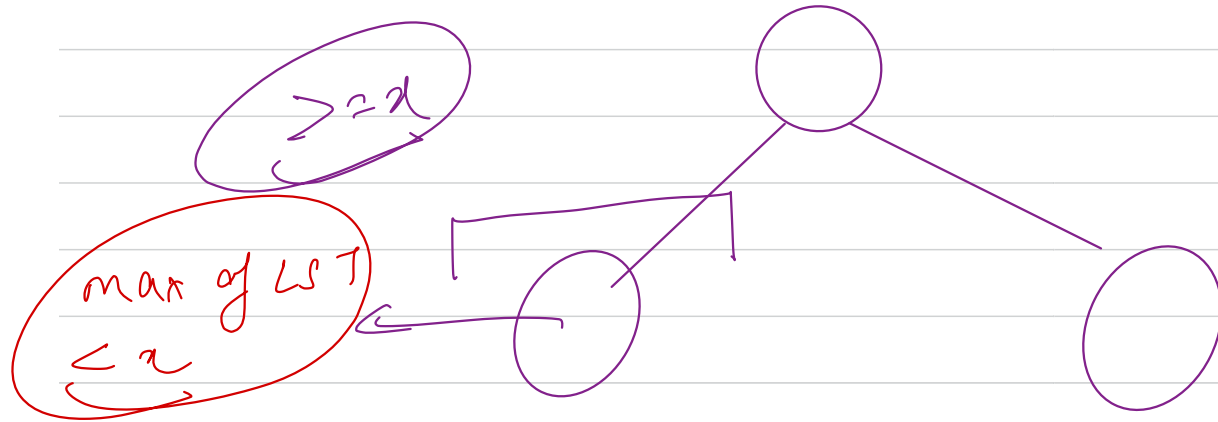a range to perform
query b

3   2   1   3   2 6 4 8

$x = 5$

$\rightarrow$ find first value atleast $x$.



$>= x$

max of LST $< x$

Can we eliminate any subtree $\geq x$.

if max of segment is $< \alpha$, then

that segment is of _no use._

**First Values**

| 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|
| 3, | 2, | 1, | 2, | 2, | 6, | 4, | 8 |

**Max Seg Tree**

$x = 3$

$x = 5$ 6

$O(\log N)$

$O(N)$

```
                              8   0-7
                     /                    \
                 3  0,3                  8  4,7
                /       \               /        \
           3 0-1      2 2-3        6 4,5        8 6,7
           /    \      /    \       /    \       /    \
        3      2    1      2     2      6     4      8
       0,0    1,1  2,2    3,3   4,4    5,5   6,6    7,7
```