MAX ABSOLUTE DIFF

Given an Array.
Find the MAX value of :

$$f(i,j) = |A_i - A_j| + |i - j|$$

$i \neq 0$

$f(i,j)$

$|x| = abs(x)$

$$|x| = x \; ; x \geq 0$$
$$= -x \; : x < 0$$

$|4| \rightarrow 4$

$|-4| \rightarrow -(-4) = 4$

A : [1, 3, -1]
0 1 2

$f(0,0) = |A_0 - A_0| + |0 - 0|$

$= |1 - 1| + |0 - 0|$

$= 0$

$f(0,1) = |A_0 - A_1| + |0 - 1|$

$= |1 - 3| + |0 - 1|$

$= |-2| + |-1|$

$2 + 1 = 3$

I BF

$0 <= i <= N-1$
$0 <= j <= N-1$

```
ANS = 0;
for (i=0; i<N; i++) {
    for (j=0; j<N; j++) {
        ANS = MAX(ANS, abs(A[i] - A[j] +
                           abs(i-j));
    }
}
cout ANS;
```

TC : $O(N^2)$

SC : $O(1)$

II

$A = [1, 3, -1]$ (indices 0, 1, 2)

$f(0,0) = |1-1| + |0-0| = 0$

$f(0,1) = |1-3| + |0-1| = 3$

$f(0,2) = |1-(-1)| + |0-2| = 4$

$f(1,0) = |3-1| + |1-0| = 3$

$f(1,1) = |3-3| + |1-1| = 0$

$f(1,2) = |3-(-1)| + |1-2| = 5$

$f(2,0) = |-1-1| + |2-0| = 4$

$f(2,1) = |-1-3| + |2-1| = 5$

$f(2,2) = |-1-(-1)| + |2-2| = 0$

Obs

$$f(i,i) = 0$$

$$f(i,j) = f(j,i)$$

→ $i < j$

$$f(i,j) = |A_i - A_j| + |i - j|$$

$$f(i,j) = |A_i - A_j| + (j - i)$$

Case I → $A_i \geq A_j$

$$f(i,j) = (A_i - A_j) + (j - i)$$

$$= (A_i - i) - (A_j - j)$$

$$X_{[K]} = A_{[K]} - K$$

$$f(i,j) = X_i - X_j$$

$$f(i,j) = X_{MAX} - X_{MIN}$$

	0	1	2 (i)	3	4
A	1	2	5	2	6
X	1	1	3	-1	2

CASE : 2

$A_i < A_j$

$$f(i,j) = |A_i - A_j| + (j-i)$$

$$= (A_j - A_i) + (j-i)$$

$$= (A_j + j) - (A_i + i)$$

$$Y_k = A_k + k$$

$$f(i,j) = Y_j - Y_i$$

$$f(i,j) = Y_{MAX} - Y_{MIN}$$

$$ANS = MAX(X_{MAX} - X_{MIN},\ Y_{MAX} - Y_{MIN})$$

A

X

Y

X_{MAX}, X_{MIN}

Y_{MAX}, Y_{MIN}

$O(N) + O(N)$

$O(N) + O(N)$

$TC = O(N)$

$SC = O(N)$

$X_{MAX} = -\infty$, $X_{MIN} = \infty$

$Y_{MAX} = -\infty$, $Y_{MIN} = \infty$

```
for (i = 0; i < N; i++) {
    X = A[i] - i;
    Y = A[i] + i;
    XMAX = MAX(XMAX, X);
    XMIN = MIN(XMIN, X);
    YMAX = MAX(YMAX, Y);
    YMIN = MIN(YMIN, Y);
}
ret MAX(XMAX - XMIN, YMAX - YMIN);
```

$TC = O(N)$

$SC = O(1)$

FIND the first MISSING INTEGER

Given an array. Find the first (+ve) int. which is missing from the array!

[1, 2, 3 .. ∞]

A = [8, 10, 1, -3, 2, -5] → 3 ✔

I) BF

TC = O(N²)

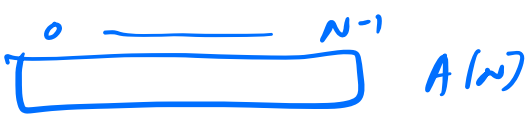
SL = O(1)

f (i = 1 → ∞) → N+1

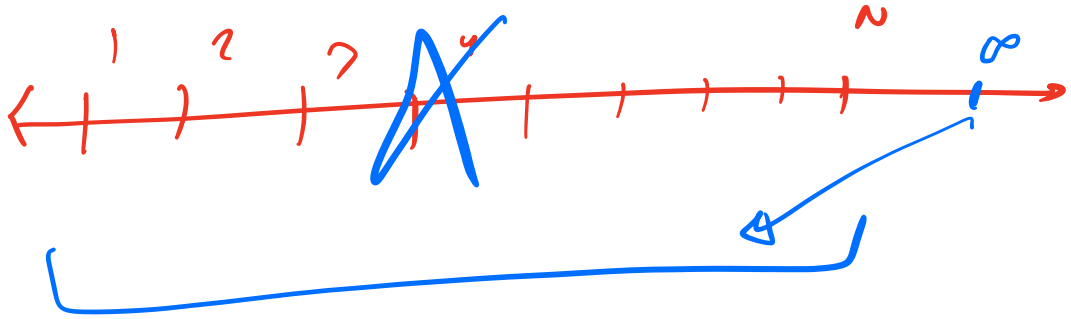
check if i is present in A[] → O(N)

Range of ANS

[1, N+1]



N+1



II Hash Set

1 → Put all elements in a HS → O(N)

2 → for (i = 1 → N+1) → N+1

check if i is in HS → O(1)

↳ NOT : ANS = i

TC = O(N)

SC = O(N)

III Sort

A : [8, 10, 1, -3, 2, -5]

DISTINCT

Sort → N log N

A : [-5, -3, 1, 2, 8, 10]

+ve

x

-ve

→ O(N)

cnt = 1 → 2 → 3 → ANS

TC = O(N lg N)

SC = O(1)

DUPLICATE

Sort A

-ve

1 2 2 3 3 3 4 7

x

ANS = 5

cnt 1 2 2 3 3 3 4 5

IV

N = 6

A: [8, 10, 1, -3, 2, 5]

1 2 -3 8 5 10

ANS [1, 7]

NOT interested in <= 0 , > N

N = 6

[1, 6]

0 1 2 3 4 5

A: [~~8~~, ~~10~~, ~~1~~, -3, ~~2~~, ~~5~~]

1 2 8 10 5

0 1 2 3 4 5 6 7 8

A: [~~8~~, ~~10~~, ~~2~~, ~~1~~, -1, 2, ~~7~~, ~~8~~, 5]

~~7~~ 2 10 8 7 8

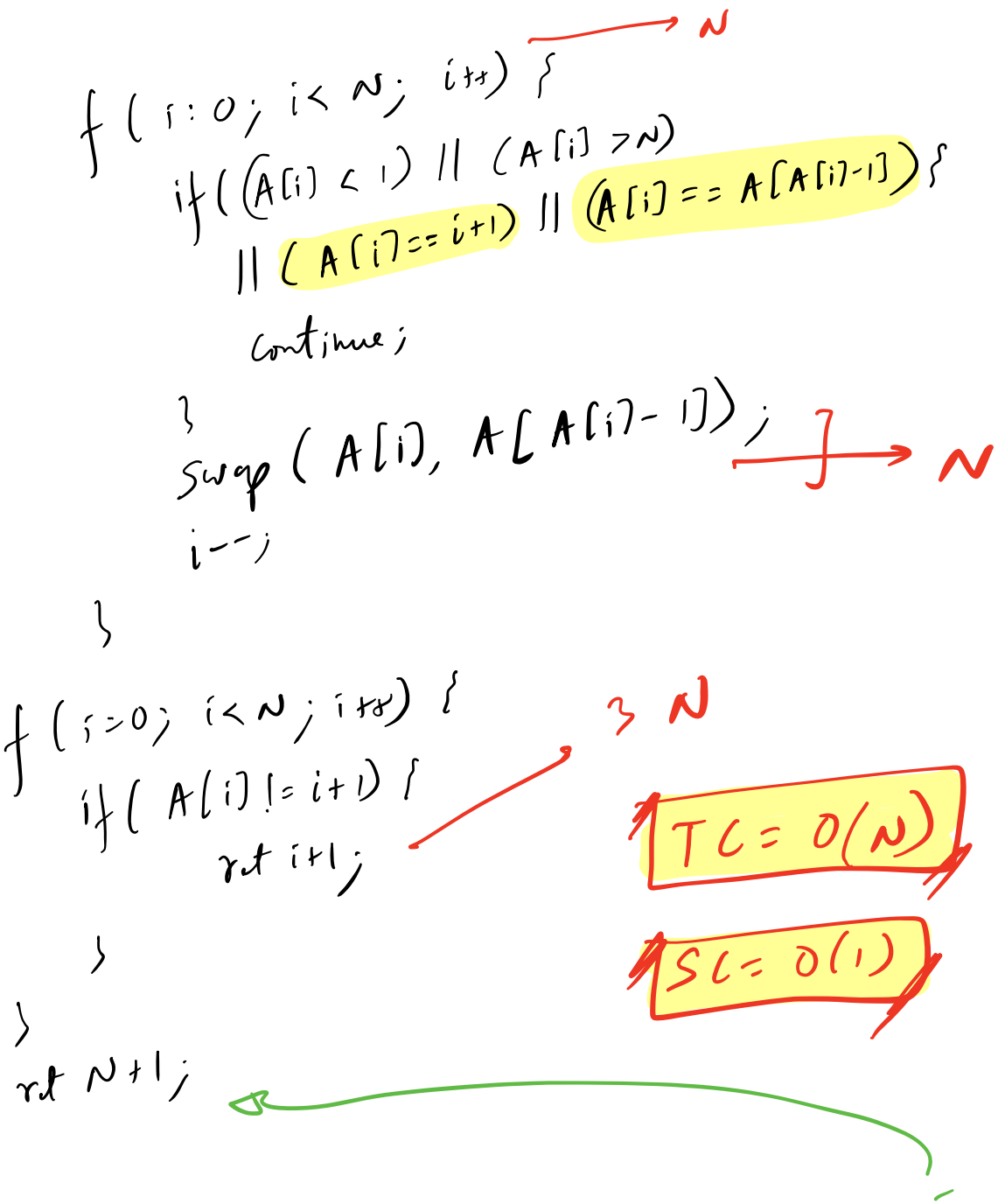
~~8~~ 3

1

[1, 5]

ANS

A[i] → None A[i]-1 index

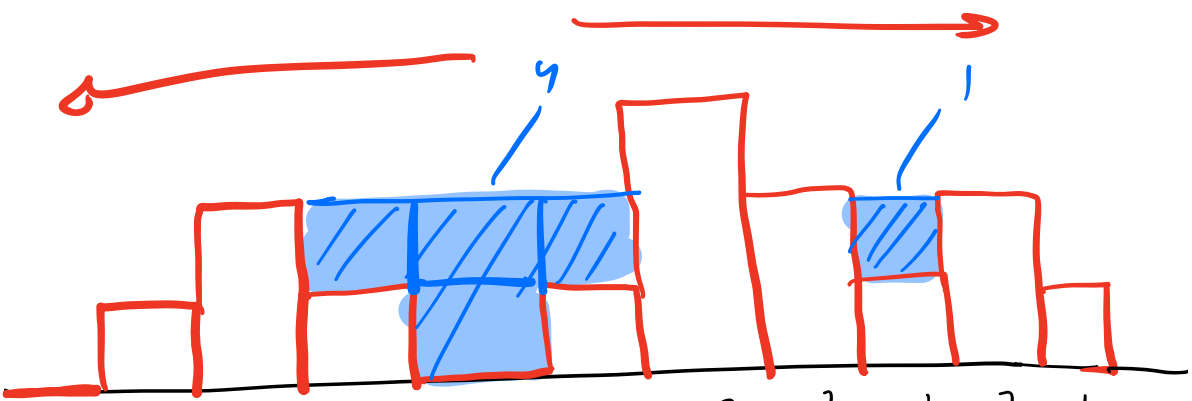

for (i:0; i<N; i++) {
if((A[i] < 1) || (A[i] > N)
|| (A[i]==i+1) || (A[i]==A[A[i]-1])){
continue;
}
swap(A[i], A[A[i]-1]);
i--;
}
N
for(i=0; i<N; i++) {
if(A[i] != i+1) {
ret i+1;
}
}
ret N+1;
3 N
TC = O(N)
SC = O(1)

RAIN WATER TRAPPING

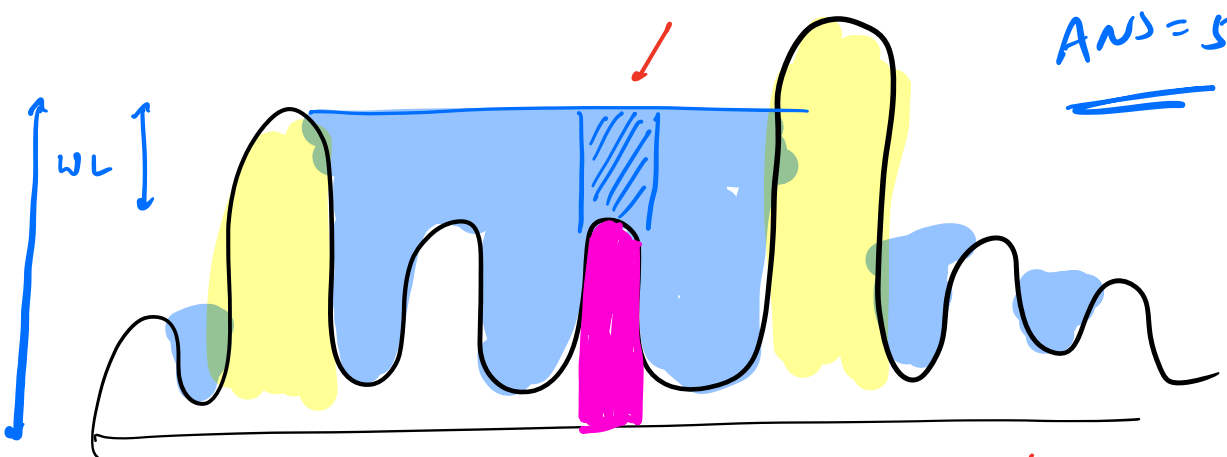
Given an array A. A[i] → height of the wall at ith index!

Imagine that it rains.

Find the amount of water collected!



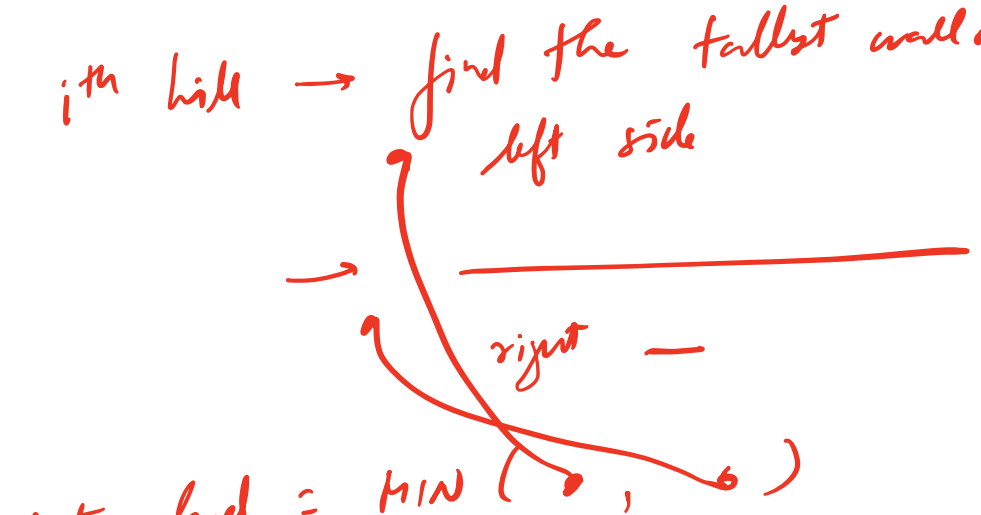
A: 0 1 2 1 0 1 3 2 1 2 1



ANS = 5

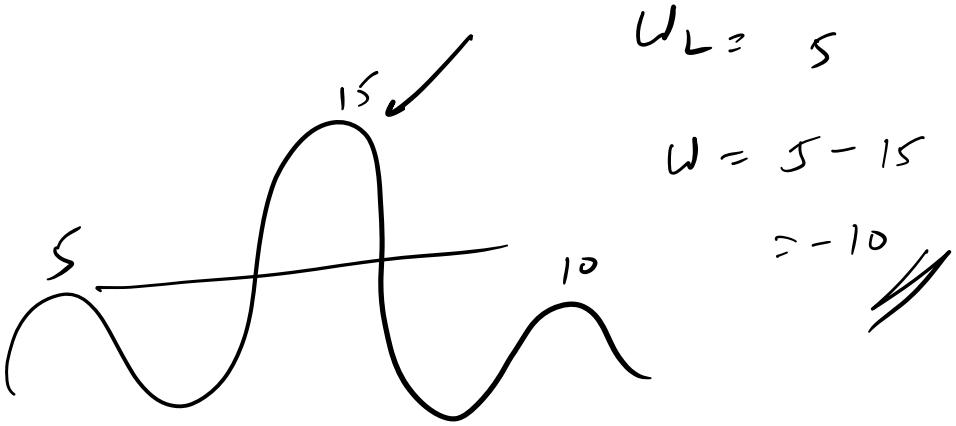
Idea: find water on top of every hill

⇒ Contribution technique!



Water level = MIN (,)

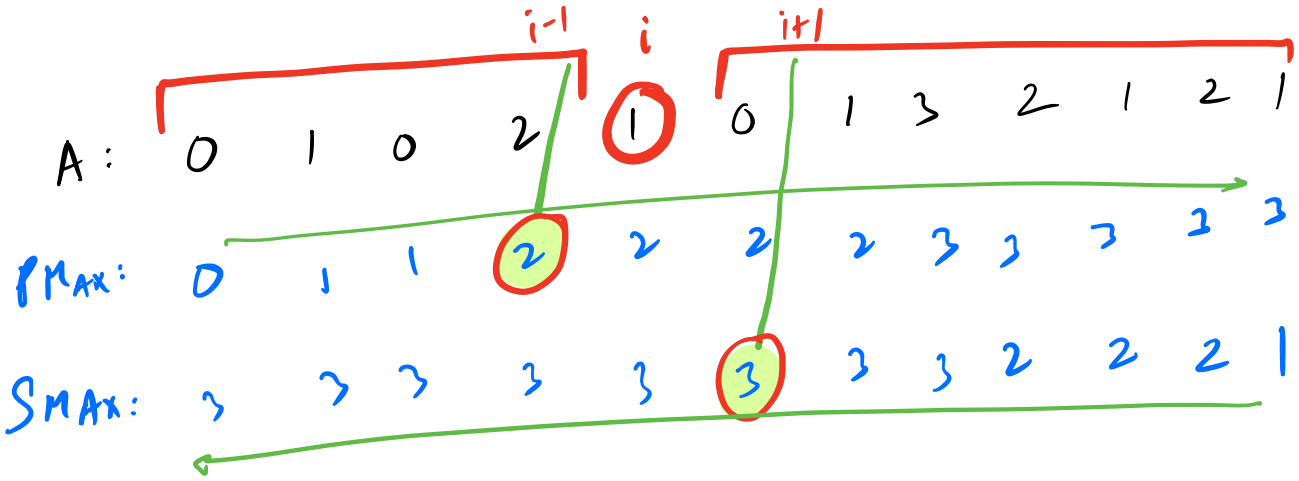
Water = MAX(0, Water level − A[i])



$W_L = 5$

$W = 5 - 15$

$= -10$



$$WL_i = MIN(PM[i-1], SM[i+1])$$

$$W_i = MAX(0, WL_i - A_i)$$

$$ANS = \sum_{i=0}^{N-1} W_i$$

$$TC = O(N)$$

$$SC = O(N)$$