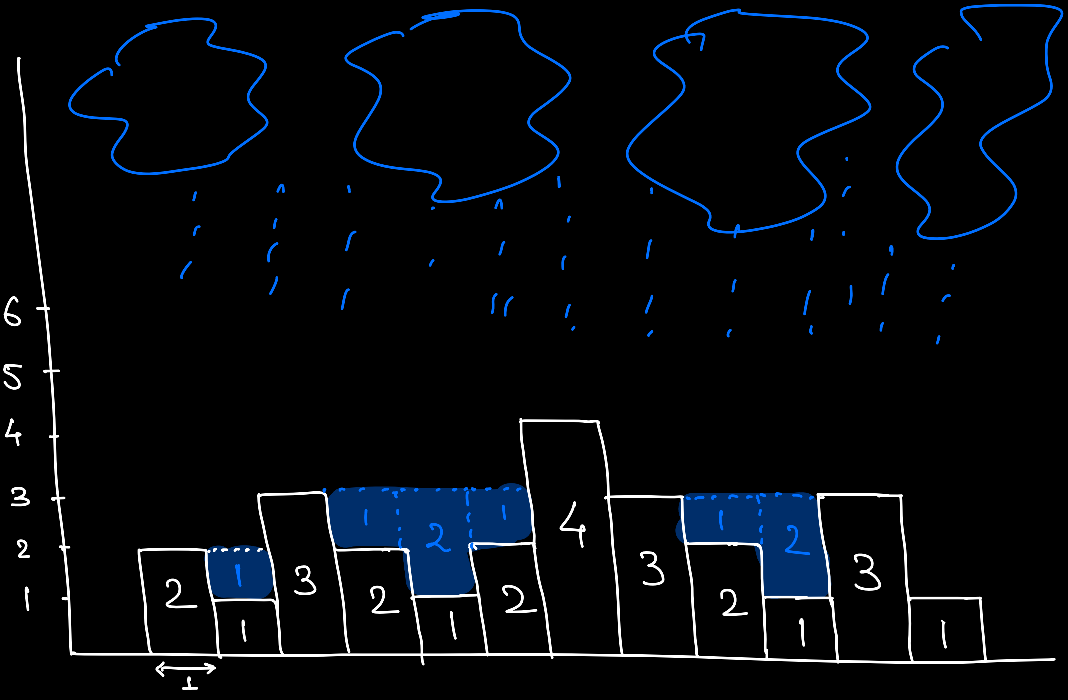
Q: Rain Water Trapping

MS | Amazon | Apple | GS | JPM | Bloomberg | Paytm | - -

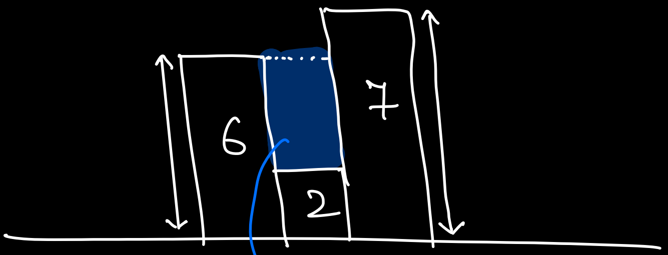
Given N array elements, A[i] denotes the height of ith Building. Return the units of water trapper b/w the buildings

Ex:- $\{ \overset{0}{2}, \overset{1}{1}, \overset{2}{3}, \overset{3}{2}, \overset{4}{1}, \overset{5}{2}, \overset{6}{4}, \overset{7}{3}, \overset{8}{2}, \overset{9}{1}, \overset{10}{3}, \overset{11}{1} \}$

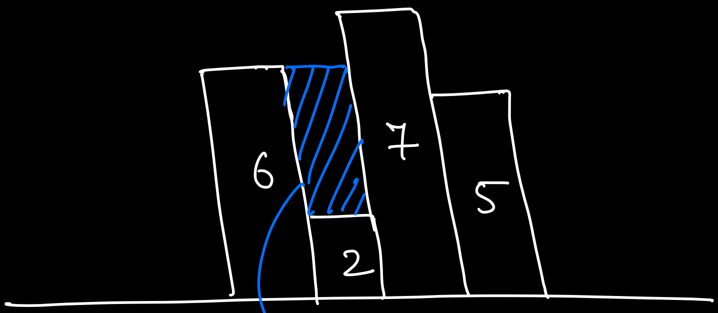


$\Rightarrow 8$

Ans :- Sum of units of water trapped on top of each building.

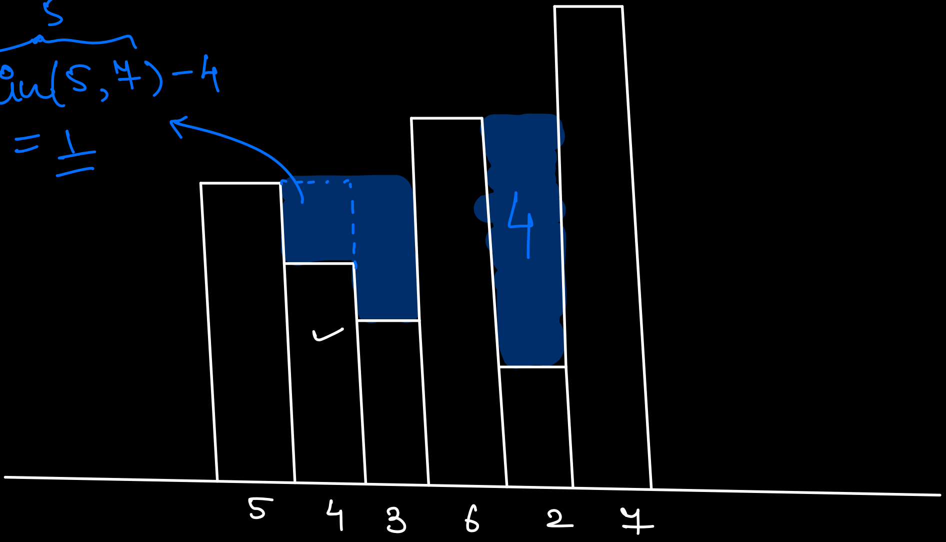


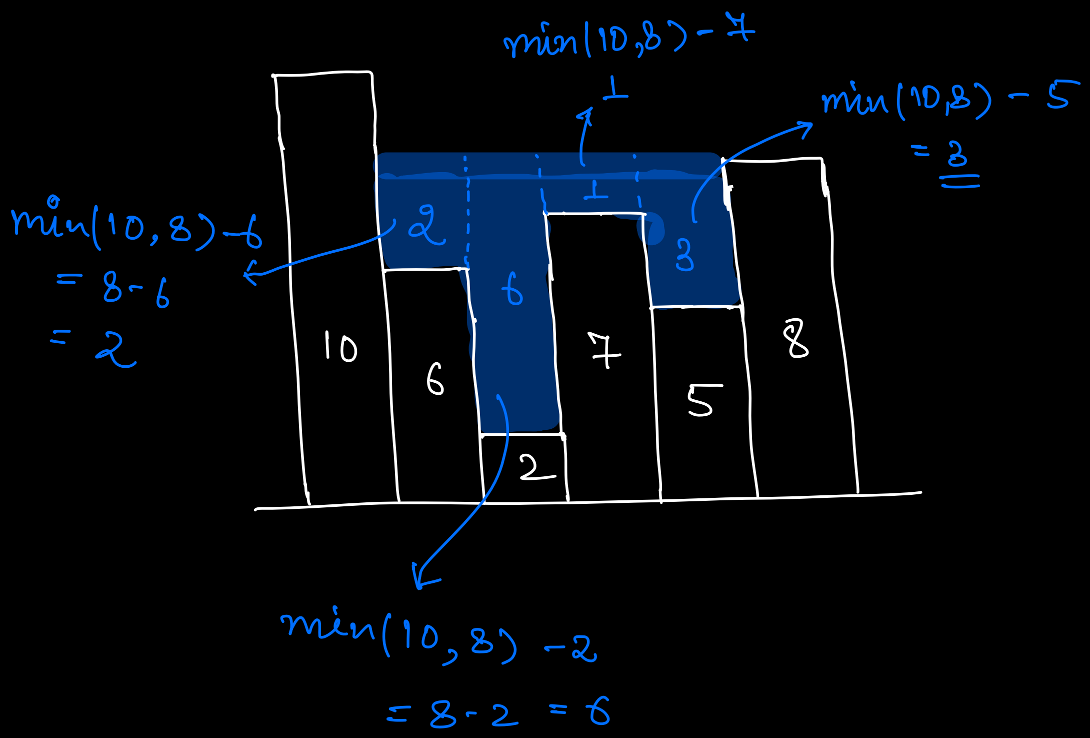
$$\min(6,7)-2 = 6-2 = \underline{\underline{4}}$$



$$\min(6,7)-2 = 4$$

$$\overbrace{\min(5,7)}^{5}-4$$
$$= \underline{1}$$





min(10,8) - 7
1

min(10,8) - 5
= 3

min(10,8) - 6
= 8 - 6
= 2

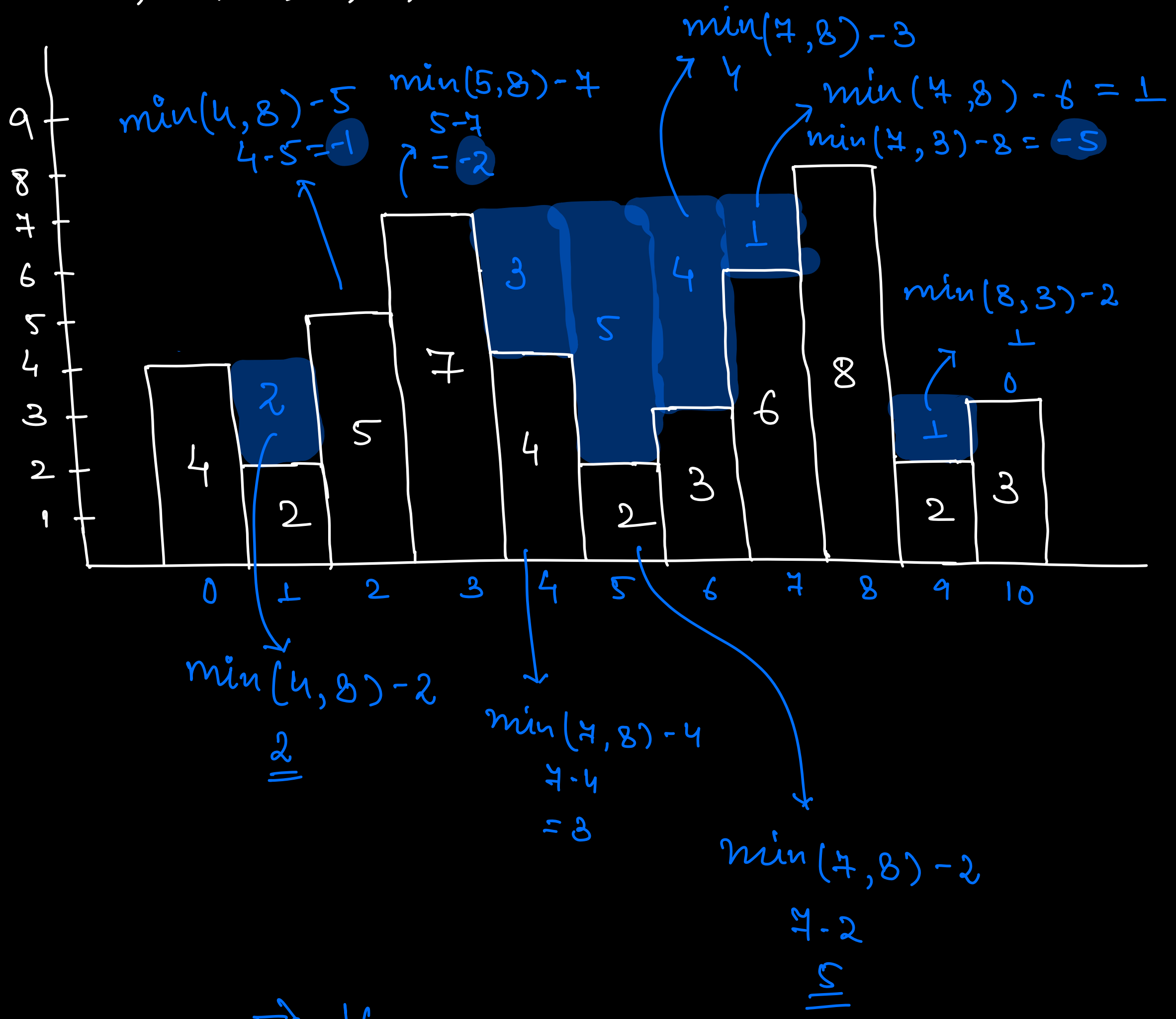
min(10,8) - 2
= 8 - 2 = 6

left boundary → left Max

right boundary → right Max

water = min(leftMax, rightMax) - A[i]

Ex:- {4, 2, 5, 7, 4, 2, 3, 6, 8, 2, 3}



⇒ 16

```
ans = 0;
for( i = 1; i < N-1; i++){
    leftMax → [0, i-1]
    rightMax → [i+1, N-1]
    Val = min(leftMax, rightMax) - A[i]
    if(Val > 0)
        ans += Val;
```

3

→ Precalculate leftMax & rightMax

	0	1	2	3	4	5	6	7	8	9	10
A :	{4	2	5	7	4	2	3	6	8	2	3}
LM :	{0	4	4	5	7	7	7	7	7	8	8}
RM	{8	8	8	8	8	8	8	8	3	3	0}

$LM[i] = max(LM[i-1], A[i-1])$

1) Create LM[] → O(N)
2) Create RM[] → O(N)
3) Find the val for each index → O(N)

TC : O(N)
SC : O(N)

\# $(a/b) \% M \rightarrow ((a \% M)/(b \% M)) \% M$ ×

$a = 16$
$b = 4$
$M = 5$

$(16/4) \% 5 \Rightarrow ((16 \% 5)/(4 \% 5)) \% 5$

$4 \% 5$
$\Rightarrow 4$

$(1/4) \% 5$
$\Rightarrow 0$

$\Rightarrow (a/b) \% M \Rightarrow (a \times b^{-1}) \% M$

$\Rightarrow (a \% M * \underbrace{b^{-1} \% M}_{\text{inverse Modulo.}}) \% M$

Inverse Modulo. :-

Given b & M

$b^{-1} \% M$ exists if $\gcd(b, M) = 1$ ↳ b & M are co-prime no's.

$$\Rightarrow b \times \frac{1}{b} = 1$$

$$\downarrow \% M$$

$$\left(b \times \frac{1}{b}\right) \% M = 1$$

$$(b \% M \times \underbrace{b^{-1} \% M}_{x}) \% M = 1$$

$$(b \% M * \underbrace{x}_{[\cancel{0}, M-1]}) \% M = 1$$

1

```
for( x = 1; x <= M-1 ; x++ ) {
        if ((b%M * x) % M == 1)
                return x;
}
```

↳ $b^{-1} \% M$

TC : O(M)

$b = 10, \; M = 7$

$x \in [1, 6]$

$x = 1 \quad (10 \% 7 * 1) \% 7 \neq 1$

✗ $\quad (3 \times 1) \% 7 \neq 1$

⋮

$x = 5 \quad (10 \% 7 \times 5) \% 7 = (3 \times 5) \% 7$

$= 15 \% 7 = ①$

$$\boxed{b^{-1} \% M = 5}$$

Fermat's Little Theorem :

Given b & M

$b^{-1} \% M$ exists if $\gcd(b, M) = 1$

↳ b & M are co-prime no's.

if M is prime :

$$(b^{M-1}) \% M = 1$$

↓ % $b^{-1} \% M$

$$b^{M-1} \,\%\, M \;*\; b^{-1} \,\%\, M \;=\; b^{-1} \,\%\, M$$

$$\Big\downarrow \%\, M$$

$$\left(b^{M-1} \,\%\, M \;*\; b^{-1} \,\%\, M\right) \%\, M = b^{-1} \,\%\, M$$

$$\Downarrow$$

$$\underbrace{\left(b^{M-1} \times b^{-1}\right)} \%\, M = b^{-1} \,\%\, M$$

$$\Downarrow$$

$$\boxed{b^{M-2} \,\%\, M = b^{-1} \,\%\, M}$$

$$\downarrow$$

Pow(a, N, P)

↳ $a^{N} \,\%\, P$

Pow(b, M-2, M)

↳ prime no.

$$\boxed{\begin{array}{l} b^{-1} \,\%\, M = b^{M-2} \,\%\, M \\ \qquad\qquad \Rightarrow \text{Pow(b, M-2, M)} \end{array}}$$

Q. Given N, r, P (prime no.)

Calculate $N_{C_r} \% P$ $[N, r < P]$

$$\left(\frac{N!}{r!\,(N-r)!}\right) \% P$$

$$\boxed{\begin{aligned}&\left(\frac{a}{b}\right) \% P\\ &\rightarrow (a \% P \times b^{-1} \% P) \% P\end{aligned}}$$

$$\downarrow$$

$$(\underset{\Uparrow}{N! \% P} * (r! * (N-r)!)^{-1} \% P) \% P$$

$$((r!)^{-1} \% P * ((N-r)!)^{-1} \% P) \% P$$

$b^{-1} \% P \rightarrow b^{P-2} \% P$

1) P is Prime no.

2) $\gcd(b, P) = 1$

$(r!)^{-1} \% P =$

$\gcd(r!, P) = 1$ ✓

$((N-r)!^{-1}) \% P$

$N < P$

$N - r < P \Rightarrow \gcd((N-r)!, P) = 1$

$$(\underbrace{r!})^{-1} \% P \;=\; (\underbrace{r!})^{P-2} \% P$$

$$\Rightarrow (\underbrace{r! \% P}_{x})^{P-2} \% P$$

$$\underbrace{x^{P-2} \% P}_{Pow(x,\, P-2,\, P)}$$

$$((N-r)!)^{-1} \% P \Rightarrow ((N-r)!^{P-2}) \% P$$

$$\Rightarrow ((\underbrace{N-r)! \% P}_{y})^{P-2} \% P$$

$$\Downarrow$$

$$Pow(y,\, P-2,\, P)$$

Q: Calculate $^{N}C_r \% P$

↳ Not a prime no.

$$^{N}C_r \% P = \left(\frac{N!}{r!(N-r)!}\right) \% P$$

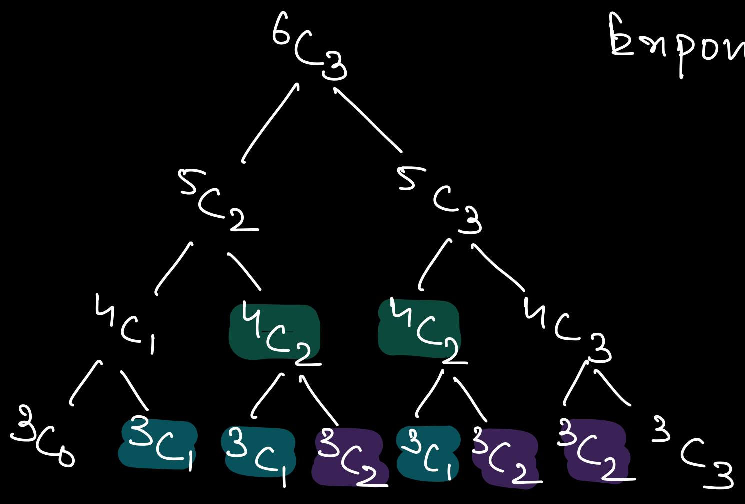
$$(a+b) \% P = (a \% P + b \% P) \% P$$

$$^{N}C_r \% P = \left(^{N-1}C_{r-1} + {}^{N-1}C_r\right) \% P$$

$$= \left(^{N-1}C_{r-1} \% P + {}^{N-1}C_r \% P\right) \% P$$

```
int   nCr( N, r, P) {
      // nCr%P
      if(r==1) return  n%P
      if(r==0 || r==n) return 1;
      return (nCr(N-1, r-1, P) +
                         nCr(N-1, r, P) ) %P
}
```

$^{n}C_0 = 1$, $^{n}C_n = 1$, $^{n}C_1 = n$



Exponential
<u>TC</u>

Q. flip

Given a Binary Array, find the subarray [s,e] when flipped (1 ⇄ 0) can give the max no. of 1's in the entire array.

i) flip operation can be done atmost once.

ii) If there's NO subarray we want to flip return [].

$$
\begin{array}{lcccccc}
 & & 0 & 1 & 2 & 3 & 4 \\
A[5] : & \{ & 1 & 0 & 1 & 0 & 0 \ \}
\end{array}
$$

[1-4] ⇒ { 1 1 0 1 1 } ⇒ 4 1's

[0-2] ⇒ { 0 1 0 0 0 } ⇒ 1 1's

[1-1] ⇒ { 1 1 1 0 0 } ⇒ 3 1's

[3-4] ⇒ { 1 0 1 1 1 } ⇒ 4 1's

$S_1\ [\overset{s}{1} - \overset{e}{4}]$

$S_2\ [3 - 4]$

$\Downarrow$

$\underline{\underline{S_1}}$

$\overset{X}{\boxed{s_1\ e_1}} \qquad \overset{Y}{\boxed{s_2\ e_2}}$

$s_1 < s_2$

$\hookrightarrow X$ else $\Rightarrow Y$

$s_1 \geq s_2 \Rightarrow e_1 \quad e_2$

$$\left.\begin{array}{l} S_1 : [2 - \textcircled{4}] \\ S_2 : [2 - \textcircled{8}] \end{array}\right\} \Rightarrow S_1$$

$$\left.\begin{array}{l} S_1 : [4 - 7] \\ S_2 : [4 - 5] \end{array}\right\} \Rightarrow S_2$$

$$\begin{array}{llllllllll}
 & & 0 & 1 & 2 & 3 & 4 & 5 & 6 & \\
A : & \{ & 0 & 0 & 1 & 0 & 0 & 1 & 0 & \} \\
{[0-1]} : & \{ & 1 & 1 & 1 & 0 & 0 & 1 & 0 & \} \Rightarrow 4 \\
S_1\ [0-4] : & \{ & 1 & 1 & 0 & 1 & 1 & 1 & 0 & \} \Rightarrow 5 \\
S_2\ [0-6] : & \{ & 1 & 1 & 0 & 1 & 1 & 0 & 1 & \} \Rightarrow 5 \\
{[3-6]} : & \{ & 0 & 0 & 1 & 1 & 1 & 0 & 1 & \} \Rightarrow 4
\end{array}$$

$\Rightarrow S_1 : [0 - 4]$

$$A : \{ \overset{0}{1} \quad \overset{1}{1} \quad \overset{2}{1} \quad \overset{3}{1} \quad \overset{4}{1} \}$$

$$\rightarrow \underline{\underline{[\ ]}}$$

Brute force

→ Iterate over every subarray & check which subarray flip is giving the max. no. of 1's.

HW   # of subarrays $= \frac{N(N+1)}{2} \approx O(N^2)$

$$O(N^2 * N) \rightarrow \boxed{O(N^3)}$$

Observations

	1's	0's		Net gain in 1's.
1)	5	3	$\xrightarrow{\text{flip}}$	−2
2)	3	6	$\xrightarrow{\text{flip}}$	+3
3)	2	7	$\xrightarrow{\text{flip}}$	+5

→ flip the subarray that gives us max gain.

Max gain ⇒ No. of 0's > No. of 1's. X

⇒

Sub 1	
1's	0's
3	7

net gain = +4

Sub 2	
1's	0's
5	8

net gain = +3

→ flip subarray 1

→ flip the subarray that gives us max gain.

gain

$$0 \xrightarrow{\text{flip}} 1 \Rightarrow +1$$

$$1 \xrightarrow{\text{flip}} 0 \Rightarrow -1$$

In array :- Replace $0 \longrightarrow +1$

$1 \longrightarrow -1$

Max gain ≡ Max Sum.

$$\begin{array}{cccccccccc} & & 0 & 1 & 2 & 3 & 4 & 5 & 6 & \\ A: & \{ & 0 & 0 & 1 & 0 & 0 & 1 & 0 & \} \end{array}$$

$$A : \{ \underbrace{\underbrace{1 \quad 1 \quad -1 \quad 1 \quad 1}_{} \quad -1 \quad 1}_{} \}$$

⇒ After replacing 0 with 1 & 1 with -1, find the subarray with max gain
[s, e]

⇒ KADANE'S ALGO

* * *