

* Hoisting:

→ It makes variables accessible / usable in the code even before they are actually declared. 'variables are lifted to the top of their scope'.

↓ Behind the scenes

Before execution, code is scanned for variable declarations and it decides whether to lift or not.

functions → Yes initial value

var → Yes, undefined

let, const → NO, <uninitialized>, TDZ

(Temporal dead zone)

(zone where the variable has no access)

Q: Maximum element and its Index :

eg: ^{0 1 2 3 4}
[1, 2, 7, 4, 10]

op: 10 4

~~-∞~~ 1

1 > -∞ ✓

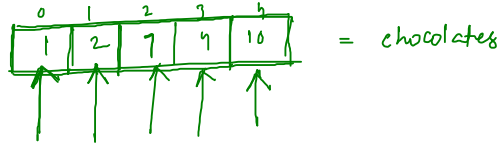
2 > 1 ✓

7 > 2 ✓

4 > 7 ✗

10 > 7 ✓✓

max_ele = ~~-∞~~ ✗ ✗ ✗ 10



if (arr[i] > max_ele) {

max_ele = arr[i];

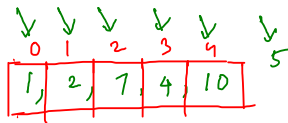
max_idx = i;

}

```

718 function ArrayProblem1(n, arr) {
719   let max_ele = -Infinity;
720   let max_idx = -1;
721
722   for (let i = 0; i < n; i++) {
723     if (arr[i] > max_ele) {
724       max_ele = arr[i];
725       max_idx = i;
726     }
727   }
728
729   return max_idx;
730 }

```



$\max_ele = -\infty \neq 10$
 $\max_idx = -1 \neq 4$

① $arr[0] > -\infty$
 $1 > -\infty$

⑤ $arr[4] > 7$
 $10 > 7 \checkmark$

② $arr[1] > 1$
 $2 > 1 \checkmark$

⑥ $i = 5 \ (5 < 5) \times$

③ $arr[2] > 2$
 $7 > 2 \checkmark$

④ $arr[3] > 7$
 $4 > 7 \times$

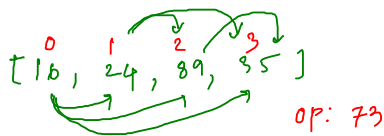
$\star \max_ele = 0 \ ?$
 $\max_idx = 0$

$[-1, -2, -3, -4, -5]$

$\max_ele = -1 \quad -1 > 0 \times$
 $\max_idx = 0 \quad -2 > 0 \times$
 $\quad \quad \quad -3 > 0 \times$
 $\quad \quad \quad -4 > 0 \times$
 $\quad \quad \quad -5 > 0 \times$

\star initialize \max_ele
 with as minimum as
 possible
 $\Rightarrow -\infty$

Q: max diff b/w any two elements:



$$\begin{aligned} (16, 24) &= 8 & (24, 89) &= 65 \\ (16, 89) &= 73 & (24, 35) &= 11 \\ (16, 35) &= 19 \end{aligned}$$

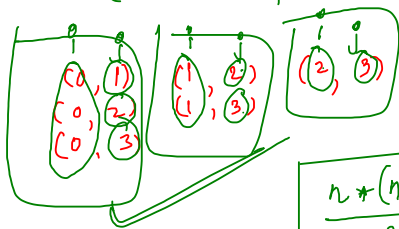
abs(diff)
 $16 - 24 = |-8| = 8$

$$(89, 35) = 54$$

```

for (let i = 0; i < n; i++) {
  for (let j = i + 1; j < n; j++) {
    console.log(arr[i], arr[j])
  }
}

```



$$\frac{n * (n - 1)}{2}$$

iterations

TLE

n = 4 ① i = 0 ② i = 1

→ j = 1 → j = 2
 → j = 2 → j = 3
 → j = 3 ~~→ j = 4~~
~~→ j = 4~~

③ i = 2

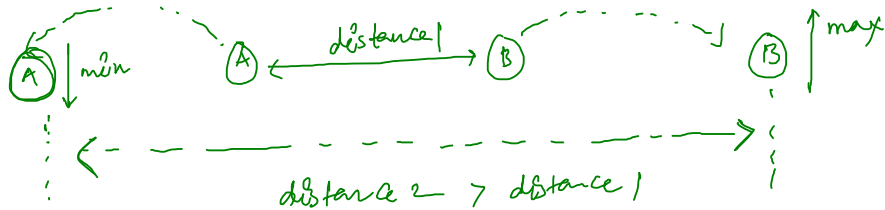
→ j = 3
~~→ j = 4~~

⑤ i = 3

~~→ j = 4~~

⑥ ~~→ j = 4~~

Improve :



$$\text{max_diff} = \text{max_ele} - \text{min_ele}$$

⇒ (n iterations)