Sorting Algorithms

Today's Checklist

- Time Complexity
- 2 Pointer approach
- Bubble sort
- Selection sort
- Insertion sort

Ques: Given an array of integers with 1 to n elements and the size of the array if n+1. One element is occurring more than once i.e duplicate number is present. Find the duplicate element.

	0	1	2	3	4	5	6	7	8	9	10
arr	6	1	7	3	2	5	4	8	9	9	10

M-I :

```
for(int i = 0 ; i < n-1 ; i++){
    for (int j = i+1 ; j < n ; j++){
        if (arr[i] == arr[j]){
            printf(   );
            break;
        }
    }
}
```

Efficient → In terms of space

	0	1	2	3	4	5	6	7	8	9 ✓	10
arr	6	1	7	3	2	5	4	8	9	9	10
	↓	↓	↓	↓	↓	↓	↓	↓	↓		

10 + 9 + 8 + 7 + 6 + 5 + 4 + 3 + 1 = 53 operations

M-2

↓

	0	1	2	3	4	5	6	7	8	9	10
brr	0	1	1	1	1	1	1	1	1	1	0

visited array

Efficient → In terms of time..

Not Efficient → In terms of space → O(n) Extra space

We are using extra space

M-3

	0	1	2	3	4	5	6	7	8	9 ✓	10
arr	6	1	7	3	2	5	4	8	9	9	10

sum = ~~0~~ ~~6~~ ~~7~~ ~~14~~ ~~17~~ ~~19~~ ~~24~~ ~~28~~ ~~36~~ ~~45~~ ~~54~~ 64

Sum of numbers from 1 to 10 → $\frac{10 \times 11}{2} = 55$

$$S_n = \frac{n(n+1)}{2}$$

$$64 - 55 = \boxed{9}$$

Efficient in terms of time & space both

3rd gen i3

M-3

3rd gen i3

M-1

Time Complexity

Space Complexity

TLE → time limit exceeded

Q/

```
for(int i=0; i<n; i++){
    printf("Hello");
}
```

$\downarrow$

n operations $\rightarrow$ $O(n)$

'Big O Notation'

$O(n+a) \cong O(n)$

$\downarrow$

constant

Q/

```
for(int i=-2; i<=n; i++){
    printf("Hello");
}
```

$n+3 \rightarrow O(n+3) \sim O(n)$

Q/

```
for(int i=1; i<=3*n; i++){
    printf("Hello");
}
```

$O(3*n) \sim O(n)$

$O(k*n) \approx O(n)$

$\downarrow$

$k \rightarrow$ constant

Q/

```
for(int i=1; i<=n*n; i++){
    printf("Hello");
}
```

$O(n*n) = O(n^2)$

Q.

```
for(int i=1; i≤n; i++){
    for(int j=10; j≤n; j++){
        printf("Hello");
    }
}
```

$i=1 \rightarrow j=1$ to n

$$\boxed{O(n^2)}$$

Q.

```
for(int i=1; i≤n; i++){
    for(int j=1; j≤i; j++){
        printf("*");
    }
    printf("\n");
}
```

$\frac{n(n+1)}{2}$ operations

$$O\left(\frac{n(n+1)}{2}\right) = O\left(\frac{n^2}{2}+\frac{n}{2}\right)$$

$$= O\left(\underbrace{\frac{1}{2}}n^2+\underbrace{\frac{1}{2}}n\right)$$

$$\approx O(n^2+n)$$

$$\approx \boxed{O(n^2)}$$

$$O(3n^3 + 2n^2 + 8n) \approx O(n^3 + n^2 + n) \approx O(n^3)$$

$$O(\sqrt{n} + 8) \approx O(\sqrt{n})$$

$$O(n^{3/2} + n + 1) \approx O(n^{3/2})$$

<u>Extra Space</u> : 'n' size array , n^2 size array , $\frac{n}{2}$ size

$\rightarrow$ 5 size array $\rightarrow$ X

$\downarrow$

$O(1)$