Day -7   DSA

* Find out the missing number from 1 to N-1 size of array.

$[\ 2, 4\ , 1\ , 5\ ]$ element start from 1 to N

N-1

↓   ↓

Ans :- 3   one of the ele is missing

⟹ $[\ 2, 4, 1, 5\ ] \xrightarrow{sort} [\ 1, 2\ , 4, 5\ ]$ **Brute Force**

→ sort → $O(n^2) / O(n\log n)$ ↓ complex logic

↓   ↳ Time Comp ↑

$for(\text{int } i = 1 ; i \le N ; i++)$ for loop ⟹ i = 1 to N

{   i

    if $(i ! = arr[i])$ Somewhere $1 = arr[i] = 1$ ✓

    return i; $i \neq arr[i]$ $2 = arr[2] = 2$ ✓   missing

} $3 = arr[3] = 4$ × ⟹ i = 3   term

                            ↘ $O(n)$

Approach - 2) :-

sum of first = 1+2+3+4+5 = A.P. = $\dfrac{n(n+1)}{2}$ = $\dfrac{5(5+1)}{2}$ = $\dfrac{5 \times 6}{2}$
N numbers

1 to 5 N = 5   = 15   total_sum   = 15

$N = 5$

$[\underline{2}, \underline{4}, \underline{1}, \underline{5}]$ ⟹
arr

$i = 0$; S_array = 0 + 2 = 2

$i = 1$; S_array = 2 + 4 = 6

$i = 2$; S_array = 6 + 1 = 7

$i = 3$; S_array = 7 + 5 = ⑫

int sum_array = 0 ⟹

for (int i = 0 ; i < arr. length ; i++)
{

　sum_array += arr[i];

}

⟶ | Sum_array = 12 |

return　total_sum － sum_array ;
　　　　　　　　15　　　　　　12

　　　　　　　　　　③

- Time
Complexity

⎱ n times

$O(n)$

$N = 10$　array.size = N+1 = 9

$[6, 7, 2, 3, 5, ①, ⑧, ⑨, ⑩]$ ⟹

⇓　int sum_array = 0

for (___)
{

___

}

⎱ 51

Total_Sum =
Sum of first
10 numbers

$= \dfrac{n \times (n+1)}{2} = \dfrac{10 \times 11}{2}$

$= \dfrac{110}{2}$

$= 55$

return $\dfrac{55 - 51}{4}$;

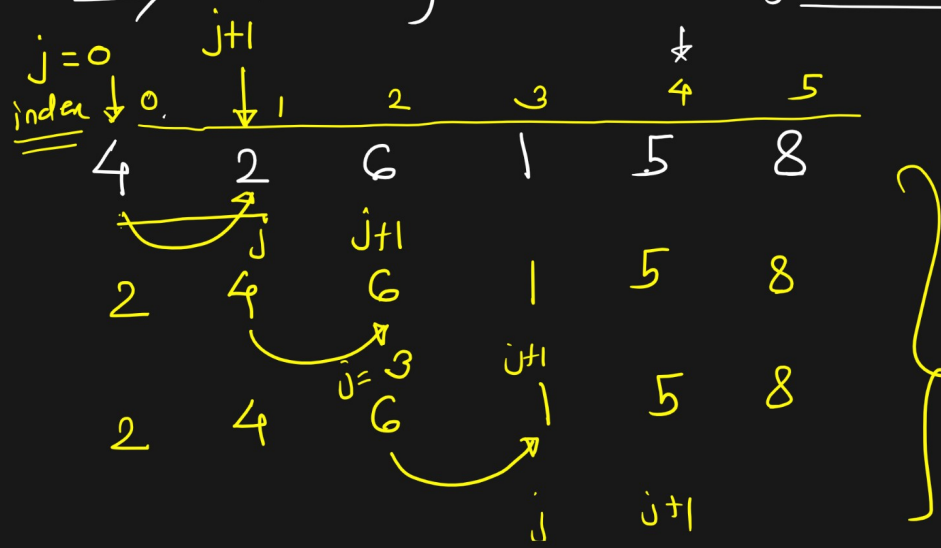* Sorting :- Sort the elements of array either in ascending or descending order.

Algorithm :-
1) Bubble Sort
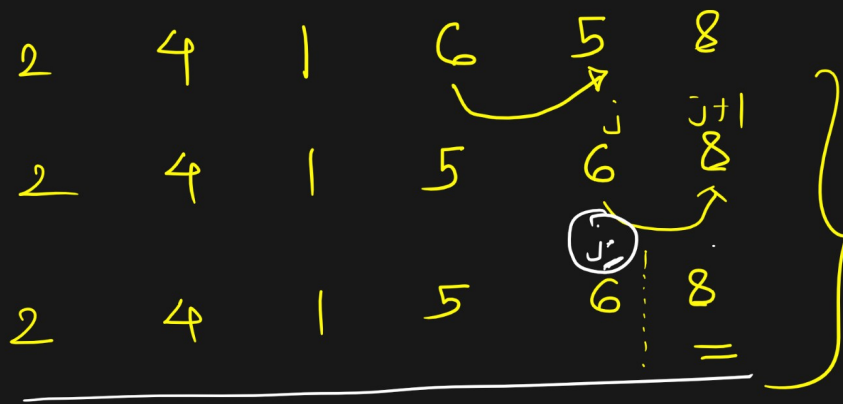2) Insertion Sort
3) Selection Sort
4) Merge Sort
5) Quick Sort

① Bubble Sort :-

⇒ Comparison based algorithm
⇒ adjacent pair comparison
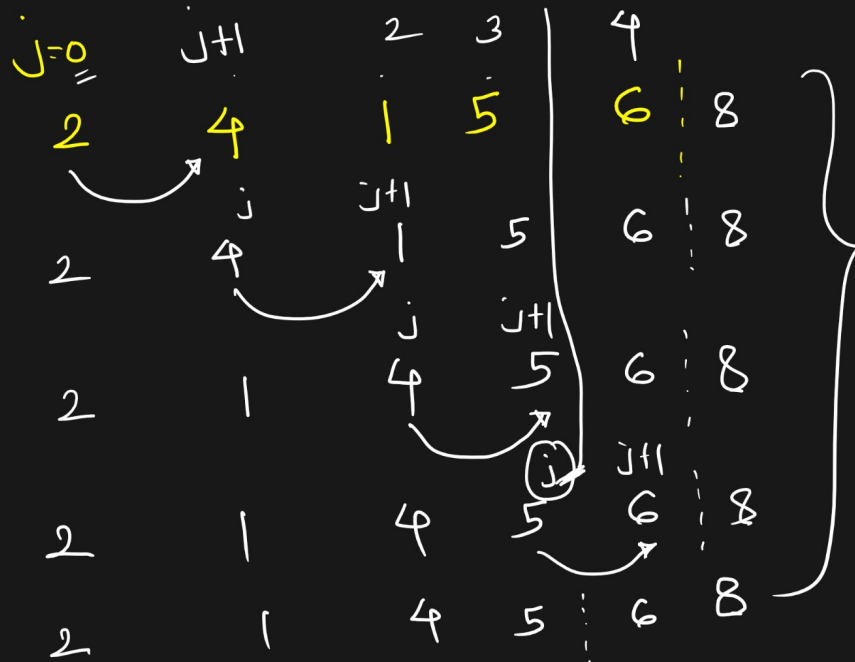⇒ If they are not in proper seq^n then we need swap.

$j=0$　　$j+1$

index →　0.　　1　　2　　3　　4　　5

4　　2　　6　　1　　5　　8

　　　　j　　$j+1$
2　　4　　6　　1　　5　　8

　　　$j=3$　　$j+1$
2　　4　　6　　1　　5　　8

　　　　j　　$j+1$

| 2 | 4 | 1 | 6 | 5 | 8 |

1st Pass = 1st largest ele at the end

| 2 | 4 | 1 | 5 | 6 | 8 |
(j, j+1)

| 2 | 4 | 1 | 5 | 6 | 8 |
(j)

| 2 | 4 | 1 | 5 | 6 | 8 |

$6 \Rightarrow j = 0 \text{ to } 5 = 0 \text{ to } N-1-$

$0 \text{ to } < N-1$

$0 \text{ to } < \dfrac{6-1}{5}$

$0 \text{ to } 4$

---

j=0   j+1   2   3   4

| 2 | 4 | 1 | 5 | 6 | 8 |

| 2 | 4 | 1 | 5 | 6 | 8 |
    j   j+1

| 2 | 4 | 1 | 5 | 6 | 8 |
        j   j+1

| 2 | 1 | 4 | 5 | 6 | 8 |
            j   j+1

| 2 | 1 | 4 | 5 | 6 | 8 |

| 2 | 1 | 4 | 5 | 6 | 8 |

2nd pass $\Rightarrow$ 2nd largest ele

---

j=0   j+1

| 2 | 1 | 4 | 5 | 6 | 8 |

| 1 | 2 | 4 | 5 | 6 | 8 |
    j   j+1

| 1 | 2 | 4 | 5 | 6 | 8 |
        j   j+1

| 1 | 2 | 4 | 5 | 6 | 8 |

3rd pass

$j=0$ $j+1$

$\quad$ 2 $\quad$ 4 $\quad$ | 5 $\quad$ 6 $\quad$ 8

$\quad$ 1

$\qquad$ j $\quad$ $j+1$

$\qquad$ 2 $\quad$ 4 $\quad$ 5 $\quad$ 6 $\quad$ 8

$\qquad$ 1

$\quad$ 1 $\quad$ 2 $\quad$ | 4 $\quad$ 5 $\quad$ 6 $\quad$ 8

$\left.\begin{array}{l}\end{array}\right\}$ 4th pass

$j=0$ $\qquad$ $j+1$

$\quad$ 2 $\quad$ | 4 $\quad$ 5 $\quad$ 6 $\quad$ 8

$\quad$ 1

$\quad$ 1 $\quad$ | 2 $\quad$ 4 $\quad$ 5 $\quad$ 6 $\quad$ 8

$\left.\begin{array}{l}\end{array}\right\}$ 5th pass

Sorted elements

$$\text{for (int } i = 0; \ i < N-1; \ i++) \longrightarrow \quad 0 \text{ to } 4 = 5 \text{ times}$$

$\qquad\qquad\qquad\qquad\qquad$ 6-1 = 5

$\qquad\qquad\qquad\qquad\qquad\qquad$ N-1-0 = 6-1-0 = 5

$\qquad\qquad\qquad\qquad\qquad\qquad\qquad$ 6 to 8

$\qquad\qquad\qquad\qquad\qquad\qquad\qquad$ < 6 = 4

$\qquad$ {

$\qquad\qquad$ for (int $j = 0$; $\ j < N-1-i$; $\ j++$)

$\begin{array}{ll} 1 & 0 \\ 2 & 1 \\ 3 & 2 \end{array}$

$\qquad\qquad$ {

$\qquad\qquad\qquad$ if (arr[j] > arr[j+1])

$\qquad\qquad\qquad$ {

$\qquad\qquad\qquad\qquad$ swap (arr[j], arr[j+1]);

No of passes

$\lfloor^{4} \quad \checkmark \quad ^{3}$

$\downarrow_5 \quad \checkmark \quad 4 \quad \lfloor \quad \} \quad \}$ $\qquad \qquad \qquad \rfloor$

$\}$

$n = 6 \quad \}$ $\qquad \qquad < n-1 = < 6-1 = <5$

$i = 0 \qquad \qquad 1 \qquad \qquad 2 \qquad \qquad 3 \qquad \qquad 4$

$j = \underset{\downarrow}{}$

$0 \text{ to } n-1-i \qquad 0 \text{ to } n-1-i$

$0 \text{ to } 6-1-0 \qquad 0 \text{ to } 6-1-1$

$< 6-1-0 \qquad \qquad \qquad \qquad 3 \qquad \qquad 2 \qquad \qquad 1$

$< 5 \qquad \qquad 4 \text{ times}$

5 times
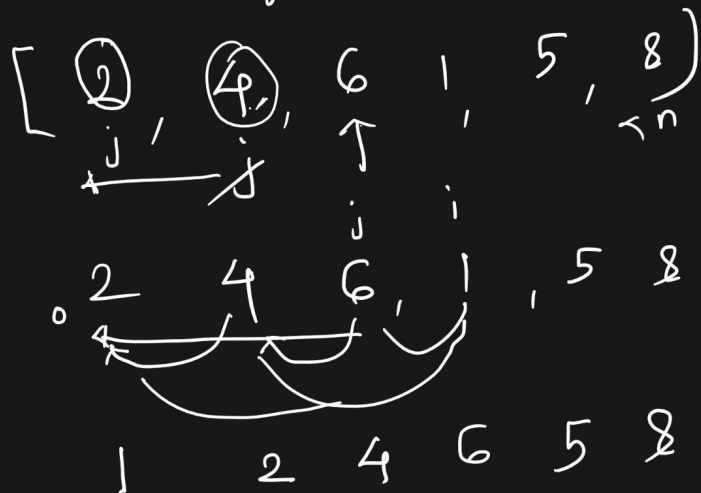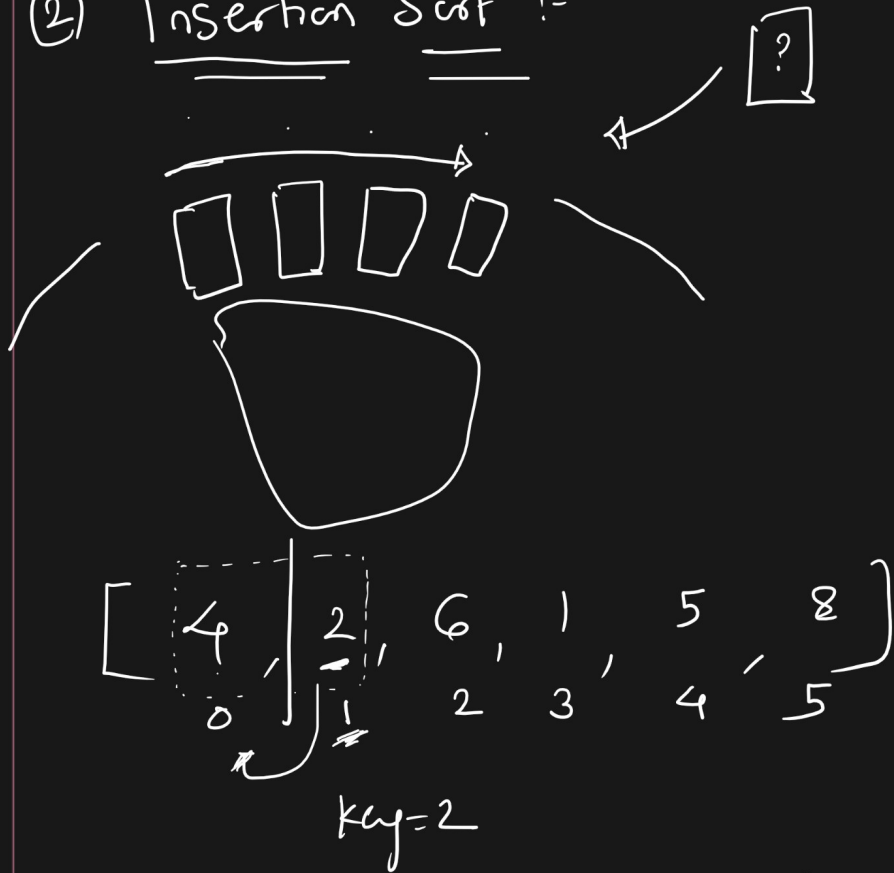
$5 + 4 + 3 + 2 + 1 = \text{sum of } n = \boxed{\dfrac{n(n+1)}{2}} = \dfrac{n^2}{2} + \dfrac{n}{2}$

$\uparrow$

$O(n^2)$

② Insertion Sort :-

?

$\begin{bmatrix} \boxed{4} & | & 2 & , & 6 & , & 1 & , & 5 & , & 8 \\ 0 & | & 1 & & 2 & & 3 & & 4 & & 5 \end{bmatrix}$

key = 2

$\begin{bmatrix} ② & , & ④ & , & 6 & 1 & , & 5 & , & 8 \\ & J & & i & \end{bmatrix}$ n

$\underset{0}{.2} \quad 4 \quad 6 \quad , \quad 1 \quad , \quad 5 \quad 8$

$1 \quad 2 \quad 4 \quad 6 \quad 5 \quad 8$

Ⓐ 0 1 2

Compare

2 6 1 5

4

2 4

2 4 6

1 2 4 6

1 2 4 5 6

```
for (int i = 1; i < n; i++)
{
    int key = arr[i];
    int j = i-1;

    while ( arr[j] > key && j >= 0)
    {
        arr[j+1] = arr[j];
    } j--;
```

$$arr[j+1] = key;$$

}

---

for (int i=1; i<n; i++)

index 0   1   2   3   4   5

4   ②   6   1   5   8   {

    int key = arr[i]; | n times

    int j = i-1;   ↓ n time

i=1

j

4 > 2

②key   4   6   1   5   8

while (arr[j] > key && j>=0)

{

    arr[j+1] = arr[j]

    j--;

}

⑥

j

0   1   2   3 | 4 ⑤   i=4

0   1   2   3

1   2   3   4 | 5

arr[j+1] = key;

}

1 + 2+   3 + 4 + 5 = ⑨

4 | 2

Time Complexity = $O(n^2)$

H.W. Write a code to print following patterns

1)
```
*
* *
* * *
* * * *
```

2)
```
    *
   * *
  * * *
 * * * *
```

3)
```
      *
    * *
  * * *
* * * *
```

4)
```
* * * *
 * * *
  * *
   *
```

5)
```
* * * *
* * *
* *
*
```