① Explain how main sorting Algorithms can be performed array an appropriate array example.

There are 4 main sorting algorithms. They are Bubble sort, Selection sort, Insertion sort, Merge Sort.

Bubble Sort

Ex :- [5, 2, 8, 12, 3]

* select two by two bubble and choose the minimum and sortout

⑤, ②, 8, 12, 3
↑___↑
swap

output → 2, 5, 8, 12, 3

* Consider the next pair and go on.

* Finally you'll get the corrected array
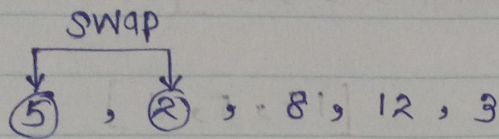
2, 5, 8, 3, 12 ← 1st Iteration
2, 5, 3, 8, 12 ← 2nd Iteration
2, 3, 5, 8, 12

Selection Sort

Ex :- [5, 2, 8, 12, 3]

* Consider the first element and compare and search for an element lesser than the if found, then swap.

swap

5 , 2 , 8 , 12 , 3

2 , $\underline{5}$, 8 , 12 , 3

sorted	unsorted
[2, 3]	[8, 12, 5]
[2, 3, 5]	[8, 12]
[2, 3, 5, 8, 12]	

sorted //

Merge Sort

Ex:- $[5, 4, 6, 9, 3]$

* Divide the array two halves and sort them separatdy. Later add them together to make the final sorted array.

$[5, 4, 6]$ $[9, 3]$
$[4, 5, 6]$ $[3, 9]$

$[3, 4, 5, 6, 9]$

Insertion Sort

Ex:- $[9, 8, 14, 5, 6]$

* Consider the first element and sort and insert the relevent values in the relevent order.

$$[9, 8, 14, 5, 6]$$

$$[8, 9] \qquad [14, 5, 6]$$
$$[5, 8, 9] \qquad [6, 14]$$

$$[5, 6, 8, 9, 14]$$

② Compare and contrast bubble sort and selection sort algorithms.

Feature	Bubble Sort	Selection Sort
Concept	Compares adjacent elements and swaps them if they are in the wrong order	Finds the smallest elements in the array and swaps it with the first element
Complexity	O (1)	O (1)
Stability	Stable	Not Stable
Simplicity	Simple to understand and emplement	Simple to understand and implement
Efficiency	Not very efficient	More efficient than bubble sort

③ What are the real world examples of sorting

      Contact List

      Web search engines

      Online shoping platforms

      Databases

      Social Media platforms.

④ Write a function using pseudo on source codes to sort an integer array aseing bubble sort and selection sort.

Bubble Sort

(01) PseudoCode

```
def bubble_sort (array):
    for i in range (len (array)-1):
        for j in range (len (array) - i-1):
            if array [j] > array [j+1]:
                array [j], array [j+i] = array [j+1], array [j]
```

(02) Source Code

```
def bubble_sort (array):
    for i in range (len(array)-1):
        for j in range (len (array)-i-1):
            if array [j] > array [j+1]:
                array [j], array [j+i] = array [j+1], array [j]


if __name__ == "__main__":
    array = [10,5,2,7,3,1,9,6,4,8]
    print ("Unsorted array:", array)
    bubble_sort (array)
    print ("sorted array:", array)
```