🔷 What is Bubble Sort?

Bubble Sort is a simple sorting algorithm that:

- Repeatedly compares adjacent elements
- Swaps them if they are in the wrong order
- Largest values **"bubble up"** to the end of the array in each round

---

🛍️ Real-World Analogy:

Imagine sorting a row of books by height. You walk through the row and swap two books if the taller one comes before the shorter one. You keep doing this until no more swaps are needed.

✅ Bubble Sort C Code

```
#include <stdio.h>

int main() {

  int arr[] = {5, 1, 4, 2, 8};

  int size = sizeof(arr) / sizeof(arr[0]);


  for(int i = 0; i < size - 1; i++) {

    for(int j = 0; j < size - i - 1; j++) {

      if(arr[j] > arr[j + 1]) {

        // Swap

        int temp = arr[j];

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

        arr[j + 1] = temp;

      }

    }

  }

  // Print sorted array

  printf("Sorted array: ");

  for(int i = 0; i < size; i++) {

    printf("%d ", arr[i]);

  }

  return 0;

}
```

🧠 Beginner-Friendly Explanation

- **Outer loop**: Controls the number of passes (size - 1)
- **Inner loop**: Compares and swaps adjacent values
- **Swap**: If current element is bigger than the next, swap them

🔁 Repeats until the largest elements are at the end.

✍️ Bubble Sort Algorithm

1. Start
2. Initialize array and find its size
3. Repeat steps 4–6 for i = 0 to size - 1
4. Repeat for j = 0 to size - i - 2:
   a. If array[j] > array[j + 1], swap them
5. After inner loop, largest element is in correct position
6. Repeat until all elements are sorted
7. Print the sorted array
8. Stop

📈 Time Complexity

Case	Time
Best (sorted)	O(n) ✅
Worst	$O(n^2)$ ❌
Average	$O(n^2)$ ❌

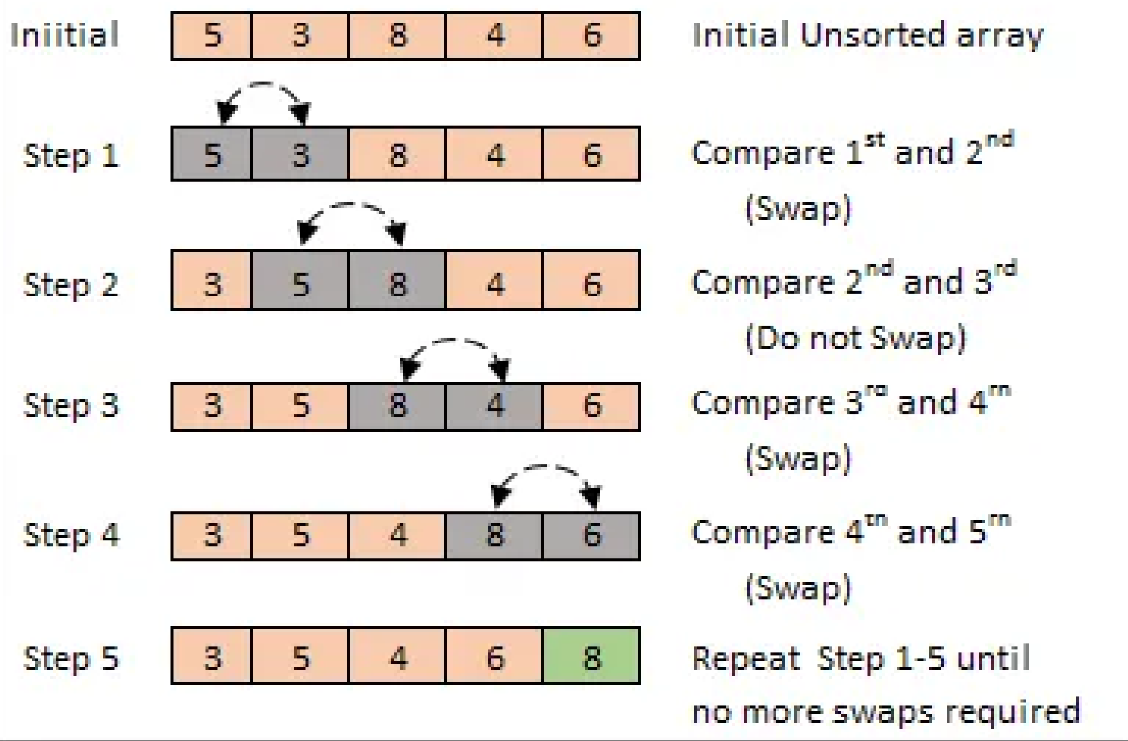
🧠 Use When:

- Working on small datasets

❌ Avoid When:

- You need performance
- Dealing with large data

Bubble sort example


Iniitial
5 3 8 4 6
Initial Unsorted array
Step 1
5 3 8 4 6
Compare 1st and 2nd
(Swap)
Step 2
3 5 8 4 6
Compare 2nd and 3rd
(Do not Swap)
Step 3
3 5 8 4 6
Compare 3rd and 4th
(Swap)
Step 4
3 5 4 8 6
Compare 4th and 5th
(Swap)
Step 5
3 5 4 6 8
Repeat Step 1-5 until
no more swaps required