Bubble SORT ALGORITHM

Let array be

int a[] = | 10 | 9 | 8 | 7 | 6 |
0 1 2 3 4

no. of elements = 5 = n

now we will apply bubble-sort.

10]
9]
8]
7
6

Comparision if (A[J] > A[J+1]) then swap & so on.

10]	9	9	9	9
9]	10]	8	8	8
8	8]	10]	7	7
7	7	7]	10]	6
6	6	6	6]	10

Pass-1 is completed & array is.

9	8	7	6	10
0	1	2	3	4

Now we can say that by Pass-1 greatest element move to the end of array.

Here no. of Comp = 4.
Max^m no. of swaps = 4.

Pass-2

9	8	7	6	10
0	1	2	3	4

comp(A[i] & A[i+1])
if (A[i] > A[i+1])
swap (& A[i] & A[i+1])

9]	8	8	8
8]	9]	7	7
7	7]	9]	6
6	6	6]	9
10	10	10	10

Now pass-2 is completed.
& no. of comparisia = 3
& no of swaps = 3
& array is

8	7	6	9	10

Pass-3

By following same condition & comparing & swapping
we get the array is

7	6	8	9	10
0	1	2	3	4

No. of comparision = 2
No of comparision = 2.

Pass-4

By following same condition

7]	6	array is
6]	7	
8	8	Now
9	9	comparision = 1
10	10	swaps = 1

6	7	8	9	10

Now by analising the above array we conclude.
No. of passes = 4 = (n-1)

Total comparision = $1+2+3+4$

$= \text{or } 1+2+3+\cdots+n-1$

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

Total swaps = $\frac{n(n-1)}{2}$

Now we can say that time complexity of bubble sort is $O\left(\frac{n(n-1)}{2}\right)$

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

Now algorithm of bubble sort is :

```
Void bubblesort (int a[], int n)
{
    int i,J;
    for(i=0; i< n-1 ; i++)
   {for(J=0; J< n-1-i; J++)
    {
        if (A[J] > A[J+1])
        {
            swap (&A[J], &A[J+1]);
        }
    }}
}
```

We can modify this algorithm by using a variable flag

Modified algorithm

```
Void bubblesort(int A[], int n)
{
    int i,J,flag;
    for(i=0; i<n-1;i++)
    {
        flag=0;
        for(J=0;J<n-1-i;J++)
        {
            if (A[J] > A[J+1])
            {
                swap(&A[i] & A[J]);
                flag=1;
            }
            if (flag==0)
            break;
        }
    }
}
```

Using this modified algorithm

If array is already sorted than it's time complexity $O(n)$;

But If we don't use this modified algorithm If array is already sorted & we apply bubble sort, then it's complexity is $O(n^2)$.