Practical No. 4

Date :-

Aim :- Write a menu driven program to sort an array using bubble sort.

Objective :- To implement Sorting Algorithm having worst case complexity of $O(n^2)$.

Practical No. 4

Date :-

Aim :- Write a menu driven program to sort an array using Bubble sort

Objective :- To implement sorting Algorithm having worst case complexity of $O(n^2)$.

Theory :- Bubble sort works on the repeatdely sorting swapping of Adjcent elements until they are not in the intended order.
The Algorithm gets it's name from the way smaller elements known as "bubbles", to the top of the list with each PASS.
Bubble sort has a time complexity of $O(n^2)$, making it less efficient compared to more advanced Algorithms for larger datasets.
The bubble sort Algorithm is adaptive, meaning it performs better on partially sorted Data.

Program Code :-

```
#include<stdio.h>
#include<stdlib.h>
void display(int a[], int n);
void bubble_sort(int a[], int n);
int main()
{
```

```
int n, choice, i;
while (1) {
printf("Enter the number of elements:\n");
scanf("%d", &n);
int arr[n];

for(i=0; i<n; i++)
  {
    printf("Enter the %d element:", i+1);
    scanf("%d", &arr[i]);
  }
  bubble_sort(arr,n);
}
}

void display(int arr[n], int n)
{
  for(i=0; i<n; i++)
    {
      printf("%d", arr[i]);
    }
}

void bubble_sort(int arr[], int n) {
int i,j, temp;
for(i=0; i<n; i++){
for(j=0; j<n-i; j++ {
if(arr[j] > arr[j+1]){
```

```
            temp = arr [j];
            arr [j] = arr [j+1];
            arr [j+1] = temp;
        }
    }
}
printf ("Bubble sorted elements are : \n");
display (arr, n);
}
```

Algorithm :-

i) Start at the Beginning :
Begin with the first element in the List
Compare Adjcent Elements

Compare the current element with the next element, swap them if Necessary

If the current pair is greater than the next element, swap them
Move to the next pair.

Move to the next pair of elements & repeat steps 2 & 3.
Continue Passes.

Continue this process for each pair of adjcent elements in the list, performing multiple passes until no more swaps are needed.

Repeat Until Sorted.

Output :- Enter the no. of elements : 3
Enter 1 Element : 69
Enter 2 Element : 96
Enter 3 Element : 696
Bubble Sorted elements are : 696 96 69
Enter the no. of elements :
⋮

Conclusion:- Program of bubble sort is implemented & executed successfully using C Programming Langauge.

ARISE & SHINE

11/03/24