**Practical-10**

**Sorting Techniques**

1. Write a program to implement bubble sort.

* Code:

#include <stdio.h>

void swap(int \*, int \*);

void bubblesort(int [], int );

void bubblesort(int arr[], int size)

{

int i, j;

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

{

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

{

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

swap(&arr[j], &arr[j+1]);

}

}

}

void swap(int \*a, int \*b)

{

int temp;

temp = \*a;

\*a = \*b;

\*b = temp;

}

int main()

{

int arr[100], i, size;

printf("Enter How Many Number You Want To Short : ");

scanf("%d", &size);

printf("\nEnter %d Numbers : \n", size);

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

{

printf("Enter Number %d=",(i+1));

scanf("%d", &arr[i]);

}

bubblesort(arr, size);

printf("\nSorted Array Is...");

for (i = 1; i< size+1; i++)

{

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

}

printf("\n");

return 0;

}

* Output:



1. Write a program to implement quick sort.

* Code:

#include <stdio.h>

int partition (int a[], int start, int end)

{

int pivot = a[end]; // pivot element

int i = (start - 1);

for (int j = start; j <= end - 1; j++)

{

if (a[j] < pivot)

{

i++; // increment index of smaller element

int t = a[i];

a[i] = a[j];

a[j] = t;

}

}

int t = a[i+1];

a[i+1] = a[end];

a[end] = t;

return (i + 1);

}

void quick(int a[], int start, int end)

{

if (start < end)

{

int p = partition(a, start, end);

quick(a, start, p - 1);

quick(a, p + 1, end);

}

}

void printArr(int a[], int n)

{

int i;

for (i = 0; i< n; i++)

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

}

int main()

{

int n;

printf("Enter value of n : ");

scanf("%d", &n);

int a[n];

printf("Enter %d elements : ", n);

for(int i=0; i<n; i++)

scanf("%d", &a[i]);

printf("\nBefore sorting array elements are - \n");

printArr(a, n);

quick(a, 0, n - 1);

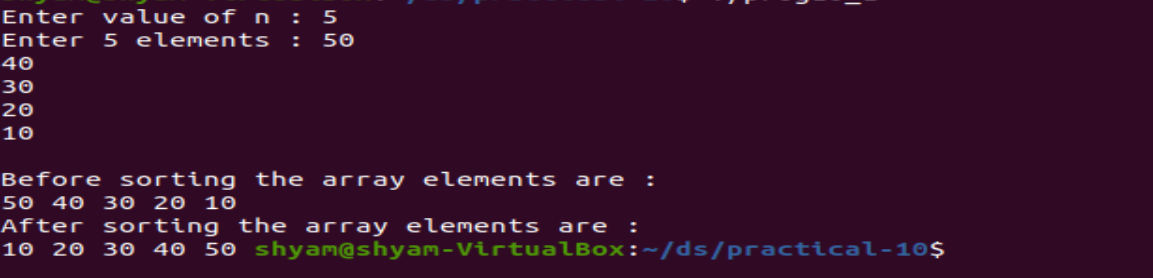
printf("\nAfter sorting array elements are - \n");

printArr(a, n);

return 0;

}

* Output:



1. Write a program to implement insertion sort.

* Code:

#include <stdio.h>

int main()

{

int n,i,j,temp;

int arr[50];

printf("Enter How Many Number You Want To Short : :");

scanf("%d", &n);

for (i=0;i<n;i++)

{

printf("Enter Number %d=",(i+1));

scanf("%d",&arr[i]);

}

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

{

j = i;

while (j>0&&arr[j-1]>arr[j])

{

temp=arr[j];

arr[j]=arr[j-1];

arr[j-1]=temp;

j--;

}

}

printf("-------------------Sorted list -----------------\n");

for (i=0;i<=n-1;i++)

{

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

}

printf("\n");

return 0;

}

* Output:

