**Name – Abhishek kumar**

**ROLL NO - 1901610100008**

**Implementation of Binary search**

#include<stdio.h>

#include<conio.h>

int main(){

int a[100],n;

printf("enter no. of elements\n");

scanf("%d",&n);

printf("enter elements\n");

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

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

}

printf("enter search element\n");

int s;

scanf("%d",&s);

int first=0;

int last=n-1;

int middle=(first+last)/2;

while(first<=last){

if(a[middle]<s){

first=middle+1;

}

else if(a[middle]==s){

printf("position:%d",middle+1);

break;

}

else{

last=middle-1;

}

middle=(first+last)/2;

}

if(first>last){

printf("not found");

}

}

output

**Implementation of Linear search**

#include<stdio.h>

#include<conio.h>

int main(){

int a[100],n;

printf("enter no. of elements\n");

scanf("%d",&n);

printf("enter elements\n");

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

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

}

printf("enter search element\n");

int s;

scanf("%d",&s);

int f=0;

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

if(a[i]==s){

printf("position:%d",i+1);

f=1;

}

}

if(f==0){

printf("element not found");

}

return 0;

output

}

**Write a program in c to perform traversal of an array**

#include<stdio.h>

#include<conio.h>

Void main()

{

Int n, i ,a[i];

Printf(“enter the limit of array\n”);

Scanf(“%d”,&n);

Printf(“enter an array\n”);

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

scanf(“%d”,&a[i]);

}

Printf(“ your array \n”);

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

printf(“%d”,&a[i]);

}

getch();

}

output



**Implementation of Insertion sort**

#include <stdio.h>

int main()

{

int a[100],m,n,temp=0,j;

printf("enter the size of array\n");

scanf("%d",&m);

printf("enter elements\n");

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

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

}

for(int i=1;i<m;i++){

temp=a[i];

for(j=i-1;temp<a[j]&&j>=0;j--){

a[j+1]=a[j];

}

a[j+1]=temp;

}

printf("sorted array is\n");

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

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

}

}

Output



**Implementation of Selection sort**

#include<stdio.h>

int main()

{

int a[100],n,min,minIndex;

printf("enter no. of elements\n");

scanf("%d",&n);

printf("enter elements\n");

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

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

}

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

min=a[i];

minIndex=i;

for(int j=i+1;j<n;j++){

if(a[j]<min){

min=a[j];

minIndex=j;

}

}

a[minIndex]=a[i];

a[i]=min;

}

// printing of sorted final array

printf("sotred array:\n");

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

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

}

}

output



**Implementation of Bubble sort**

#include<stdio.h>

#include<conio.h>

int main(){

int a[100],n;

printf("enter no. of elements\n");

scanf("%d",&n);

printf("enter elements\n");

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

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

}

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

{

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

{

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

{

// swap the elements

int temp = a[j];

a[j] = a[j+1];

a[j+1] = temp;

}

}

}

printf("Sorted Array: ");

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

{

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

}

}

Output



**Implementation of Merge sort**

#include<stdio.h>

#include<conio.h>

int main(){

int a[50],n ,i;

printf("enter total no. of elements\n");

scanf("%d",&n);

printf("enter the elements\n");

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

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

}

partition(n ,0,n-1);

printf("after merge sort\n");

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

{

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

{

return 0 ;

}

void partition(int a[],int low, int high)

{

int maid;

if( low<high)

{

mid =(low+high)/2;

partition(a, low, mid);

partition(a, mid+1, high);

mergesort(a, low, mid;high);

}

}

void mergesort(int a[],int low, int mid ,int high)

{

int i, mi, k, lo, temp[50];

lo = low;

i = low ;

mi mid + 1;

while(( lo<= mid) && (mmi <= high))

{

if (a[lo] <= a[mi])

{

temp[i] = a[lo];

lo++;

}

else

{

temp[i] = a[mi];

mi++;

}

i++ ;

}

if(lo > mid)

{

for(k = mi ; k <= high ; k++)

{

temp[i] = a[k];

i++;

}

}

else {

for(k = lo ; k <= mid ; k++)

{

temp[i] = a[k];

i++;

}

}

for(k = low; k <= high ; k++)

{

a[k] = temp[k];

i++;

}

}

output



**Implementation of Quick sort**

#include<stdio.h>

#include<conio.h>

void quicksort(int number[25],int first,int last){

int i, j, pivot, temp;

if(first<last){

pivot=first;

i=first;

j=last;

while(i<j){

while(number[i]<=number[pivot]&&i<last)

i++;

while(number[j]>number[pivot])

j--;

if(i<j){

temp=number[i];

number[i]=number[j];

number[j]=temp;

}

}

temp=number[pivot];

number[pivot]=number[j];

number[j]=temp;

quicksort(number,first,j-1);

quicksort(number,j+1,last);

}

}

int main(){

int i, count, number[25];

printf("How many elements are u going to enter?: ");

scanf("%d",&count);

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

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

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

quicksort(number,0,count-1);

printf("Order of Sorted elements: ");

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

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

return 0;

}

OUTPUT

