**BINARY SEARCH:**

#include <iostream>

#include<conio.h>

using namespace std;

int main()

{

int count, i, a[5], n, fst, lst, mid, j, c;

cout<<"enter the number of elements:";

cin>>count;

cout<<"enter array number";

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

{

cin>>a[i];

}

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

{

for (j=i+1;j<count;++j)

{

if (a[i]>a[j])

{

c=a[i];

a[i]=a[j];

a[j]=c;}}}

printf("The numbers arranged in ascending order are given below \n");

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

{

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

}

cout<<"Enter the number to search:";

cin>>n;

fst = 0;

lst = count-1;

mid = (fst+lst)/2;

while (fst<=lst)

{

if(a[mid]<n)

{

fst = mid+1;}

else if(a[mid] == n)

{

cout<<n<<" found in the array at the location "<<mid+1<<"\n";

break;

}

else

{

lst = mid - 1;

}

mid = (fst + lst)/2;

}

if(fst > lst)

{

cout<<n<<" not found in the array";

}

return 0;

}

OUTPUT:

enter the number of elements:4

enter array number11

2

5

6

The numbers arranged in ascending order are given below

2

5

6

11

Enter the number to search:6

6 found in the array at the location 3

**LINEAR SEARCH:**

#include <iostream>

#include<conio.h>

using namespace std;

int main()

{

int a[5],i,searchel;

cout<<"enter the array";

for(i=0;i<5;i++)

{

cin>>a[i];

}

cout<<"Enter the element u want to search";

cin>>searchel;

for(i=0;i<5;i++)

{

if(searchel==a[i])

break;

}

if(i>=5)

cout<<"Element not found";

else

cout<<"Element found at position "<<i+1;

getch();

}

OUTPUT:

enter the array1

3

5

33

6

Enter the element u want to search33

Element found at position 4

**...Program finished with exit code 0**

**Press ENTER to exit console.**

**Insertion Sort:**

#include<iostream>

using namespace std;

int main()

{

int a[10],temp,i,j,n;

cout<<"Enter the number of elements you want to enter in the array:";

cin>>n;

cout<<"\nEnter array elements\n";

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

{cin>>a[i];}

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

{

temp=a[i];

j=i-1;

while((temp<a[j])&&(j>=0))

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

j=j-1;

}

a[j+1]=temp;

}

cout<<"\ninserted elements in sorted order are:\n";

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

{cout<<a[i]<<" ";}

return 0;

}

Output:

Enter the number of elements you want to enter in the array:5

Enter array elements

55

19

87

2

22

inserted elements in sorted order are:

2 19 22 55 87

**Bubble sort:**

#include <iostream>

using namespace std;

int main()

{

int i, n,j,av,passes=0;

int arr[5];

cout<<"enter number of elements of array: ";

cin>>n;

cout<<"enter the array elements";

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

{

cin>>arr[i];

}

cout <<"unsorted array: \n";

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

cout <<arr[i]<<" ";

}

cout<<endl;

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

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

{

if(arr[j] < arr[i]) {

av = arr[i];

arr[i] = arr[j];

arr[j] = av;

}

}

passes++;

}

cout <<"Sorted array: \n";

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

cout <<arr[i]<<" ";

}

return 0;

}

Output:

Enter number of elements of array: 4

Enter the array elements: 11

45

23

9

Unsorted array:

11 45 23 9

Sorted array:

9 11 23 45

**QUICK SORT:**

#include <stdio.h>

int partition(int A[],int p,int r)

{ int i=0,j=0,x=0,temp=0;

x=A[r];

i=p-1;

for(j=p;j<=r-1;j++)

{

if(A[j]<=x)

{

i=i+1;

temp=A[i];

A[i]=A[j];

A[j]=temp;

}

}

temp=A[i+1];

A[i+1]=A[r];

A[r]=temp;

return(i+1);

}

void quicksort(int A[],int p,int r)

{

if(p<r)

{ int q=0;

q=partition(A,p,r);

quicksort(A,p,q-1);

quicksort(A,q+1,r);

}

}

int main() {

int A[10],n,p,q,r,x,i=0,temp=0;

printf("enter the size of an array");

scanf("%d",&n);

printf("enter the elements of an array");

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

{

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

}

quicksort(A,0,n-1);

printf("sorted array: ");

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

{

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

}

return 0;

}

OUTPUT:

enter the size of an array5

enter the elements of an array1

3

5

2

4

sorted array: 12345

**HEAP SORT:**

#include<stdio.h>

int temp;

void maxheapify(int A[],int n,int i)

{

int largest = i;

int left = 2\*i + 1;

int right = 2\*i + 2;

if (left<n && A[left]>A[largest])

{

largest = left;

}

if (right<n && A[right]>A[largest])

{

largest=right;

}

if (largest != i)

{

temp = A[i];

A[i]=A[largest];

A[largest] = temp;

maxheapify(A,n,largest);

}

}

void heapsort(int A[], int n)

{

int i;

for (i=n/2-1;i>=0;i--)

{

maxheapify(A,n,i);

}

for (i=n-1; i>=0; i--)

{

temp=A[0];

A[0]=A[i];

A[i]=temp;

maxheapify(A,i,0);

}

}

void main()

{

int A[10],i,n;

printf("enter the number of elements");

scanf("%d",&n);

printf("enter elements");

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

{

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

}

heapsort(A,n);

printf("sorted array\n");

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

{

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

}

}

OUTPUT:

enter the number of elements5

enter elements1

3

5

2

4

sorted array

12345

**COUNTING SORT:**

#include <stdio.h>

void counting\_sort(int A[], int k, int n)

{

int i, j;

int B[15], C[100];

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

{

C[i] = 0;

}

for (j = 1; j <= n; j++)

{

C[A[j]] = C[A[j]] + 1;

}

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

{

C[i] = C[i] + C[i-1];

}

for (j = n; j >= 1; j--)

{

B[C[A[j]]] = A[j];

C[A[j]] = C[A[j]] - 1;

}

printf("The Sorted array is : ");

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

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

}

int main()

{

int n, k = 0, A[15], i;

printf("Enter the number of input : ");

scanf("%d", &n);

printf("\nEnter the elements to be sorted :\n");

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

{

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

if (A[i] > k) {

k = A[i];

}

}

counting\_sort(A, k, n);

printf("\n");

return 0;

}

OUTPUT:

Enter the number of input : 6

1

Enter the elements to be sorted :

2

32

45

13

10

65

The Sorted array is : 10 12 13 32 45 65

**MERGE SORT:**

#include<stdio.h>

void mergesort(int a[],int i,int j);

void merge(int a[],int i1,int j1,int i2,int j2);

int main()

{

int a[30],n,i;

printf("Enter no of elements:");

scanf("%d",&n);

printf("Enter array elements:");

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

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

mergesort(a,0,n-1);

printf("\nSorted array is :");

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

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

return 0;

}

void mergesort(int a[],int i,int j)

{

int mid;

if(i<j)

{

mid=(i+j)/2;

mergesort(a,i,mid);

mergesort(a,mid+1,j);

merge(a,i,mid,mid+1,j);

}

}

void merge(int a[],int i1,int j1,int i2,int j2)

{

int temp[50];

int i,j,k;

i=i1;

j=i2;

k=0;

while(i<=j1 && j<=j2)

{

if(a[i]<a[j])

temp[k++]=a[i++];

else

temp[k++]=a[j++];

}

while(i<=j1)

temp[k++]=a[i++];

while(j<=j2)

temp[k++]=a[j++];

for(i=i1,j=0;i<=j2;i++,j++)

a[i]=temp[j];

}

OUTPUT:

Enter no of elements:5

Enter array elements:5

1

3

4

2

Sorted array is :1 2 3 4 5