

```

#include<iostream>
#include<conio.h>
using namespace std;

int bsort(int a[],int n)
{
    int temp ;
    for(int i=1;i<n;i++)
    {
        for(int j=0;j<n-i ; j++)
        {
            if(a[j+1]<a[j])
            {
                temp = a[j];
                a[j] = a[j+1];
                a[j+1] = temp;
            }
        }
    }
    cout<<"\nThe list after sorting is: \n";
    for(int i=0;i<n;i++)
    {
        cout<<a[i]<<" ";
    }
}

```

```

int insertion(int a[],int n)
{
    int key;
    for(int i=1;i<n;i++)
    {
        key=a[i];
        int j=i-1;
        while(j>=0&& a[j]>key)
        {
            a[j+1]=a[j];
            j--;
        }
        a[j+1]=key;
    }
    cout<<"\nThe list after sorting is: \n";
    for(int i=0;i<n;i++)
    {
        cout<<a[i]<<" ";
    }
}

```

```

int selection(int a[],int n)
{
    int small ,pos ,temp;
    for(int i=0;i<n-1;i++)
    {
        small = a[i];
        pos = i ;
        for(int j=i+1;j<n;j++)
        {
            if(a[j]<small)

```



```

        {
            small=a[j];
            pos=j;
        }
        temp=a[pos];
        a[pos]=a[i];
        a[i]=temp;
    }
}
cout<<"\n\nThe list after sorting is: \n";
for(int i=0;i<n;i++)
{
    cout<<a[i]<<" ";
}
}

void Merge(int *a, int low, int high, int mid)
{
    int i, j, k, temp[high-low+1];
    i = low;
    k = 0;
    j = mid + 1;

    while (i <= mid && j <= high)
    {
        if (a[i] < a[j])
        {
            temp[k] = a[i];
            k++;
            i++;
        }
        else
        {
            temp[k] = a[j];
            k++;
            j++;
        }
    }

    while (i <= mid)
    {
        temp[k] = a[i];
        k++;
        i++;
    }

    while (j <= high)
    {
        temp[k] = a[j];
        k++;
        j++;
    }
}

```



```

        for (i = low; i <= high; i++)
        {
            a[i] = temp[i-low];
        }
    }
}

```

```

void MergeSort(int *a, int low, int high)
{
    int mid;
    if (low < high)
    {
        mid=(low+high)/2;

        MergeSort(a, low, mid);
        MergeSort(a, mid+1, high);

        Merge(a, low, high, mid);
    }
}

```

```

int lsearch(int arr[],int size,int key)
{
    for(int i=0;i<size;i++)
    {
        if(key==arr[i])
        {
            cout<<"Key Found At Index Number : " <<i<<endl;
            break;
        }
    }
}

```

```

int bsearch(int arr[],int search,int n)
{
    int first = 0;
    int last = n-1;
    int mid = (first+last)/2;
    while (first <= last)
    {
        if(arr[mid] < search)
        {
            first = mid + 1;
        }
        else if(arr[mid] == search)
        {
            cout<<search<<" found at location " <<mid+1<<"\n";
            break;
        }
        else
        {
            last = mid - 1;
        }
    }
}

```



```

    }
    mid = (first + last)/2;
}
if(first > last)
{
    cout<<"Not found! "<<search<<" is not present in the list.";
}
return 0;
}

```

```
int partition(int a[],int l,int u)
```

```

{
    int v,i,j,temp;
    v=a[l];
    i=l;
    j=u+1;

    do
    {
        do
            i++;

        while(a[i]<v&& i<=u);

        do
            j--;
        while(v<a[j]);

        if(i<j)
        {
            temp=a[i];
            a[i]=a[j];
            a[j]=temp;
        }
    }while(i<j);

    a[l]=a[j];
    a[j]=v;

    return(j);
}

```

```
void quick_sort(int a[],int l,int u)
```

```

{
    int j;
    if(l<u)
    {
        j=partition(a,l,u);
        quick_sort(a,l,j-1);
        quick_sort(a,j+1,u);
    }
}

```

```
int main()
```

```
{
```



Edit with WPS Office

```

int A[20],n ;

cout<<"\nEnter the no of elements in array: ";
cin>>n;
int lb=0,ub=n-1;
cout<<"\nEnter list elements\n";
for(int i=0;i<n;i++)
{
    cout<<i+1<<" ";
    cin>>A[i];
    cout<<endl;
}
cout<<"\nThe list is: \n";
for(int i=0;i<n;i++)
{
    cout<<A[i]<<" ";
}
int c;
do{

cout<<"\n\t\t\t\t MENU \t\t\t\t\n";
cout<<"1)Bubble Sort \n";
cout<<"2)Selection Sort \n";
cout<<"3)Insertion Sort \n";
cout<<"4)Quick Sort \n";
cout<<"5)Merge Sort \n";
cout<<"6)Binary Search\n";
cout<<"7)Linear Search \n";
cout<<"8)Exit\n";
cout<<"\n Enter the choice: \n";
cin>>c;

switch(c)
{
    case 1:
        bsort(A,n);
        break;
    case 2:
        selection(A,n);
        break;
    case 3:
        insertion(A,n);
        break;
    case 4:
        quick_sort(A,lb,ub);
        cout<<"\nArray after sorting:";
        for(int i=0;i<n;i++)
            cout<<A[i]<<" ";
        break;
    case 5:

        MergeSort(A,0,n-1);
        cout<<"\nSorted Data ";
        for (int i = 0; i < n; i++)
            cout<<"->"<<A[i];

        return 0;
}
}

```



```
        break;
    case 6:
        int search;
        cout<<"Enter a number to find :";
        cin>>search;
        bsearch(A,search,n);
        break;
    case 7:
        int key;
        cout<<"Enter Key To Search in Array";
        cin>>key;
        lsearch(A,n,key);
        break;
    }
}while(c!=8);
}
```

