**2.2 Linear\_Sequential\_Search.cpp**

#include<iostream>

using namespace std;

class searching{

private:

int size, arr[100], key, i;

bool found;

public:

void getdata(){

cout <<"Enter the size of array: ";

cin >> size;

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

cout << "Enter element no. " << i+1 << " : ";

cin >> arr[i];

}

cout << endl;

}

void linear\_search(){

getdata();

cout << "Enter a key element to search in the array: ";

cin >> key;

found = false;

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

{

if(arr[i]==key){

found = true;

break;

}

}

if(found)

cout << "Element is present at index number: " << i << " and location number: " << i+1 << endl;

else

{

arr[size++] = key;

cout << "Element not found in the array and has been added to the last of array.\nArray elements: ";

display();

}

}

void display(){

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

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

cout << endl;

}

};

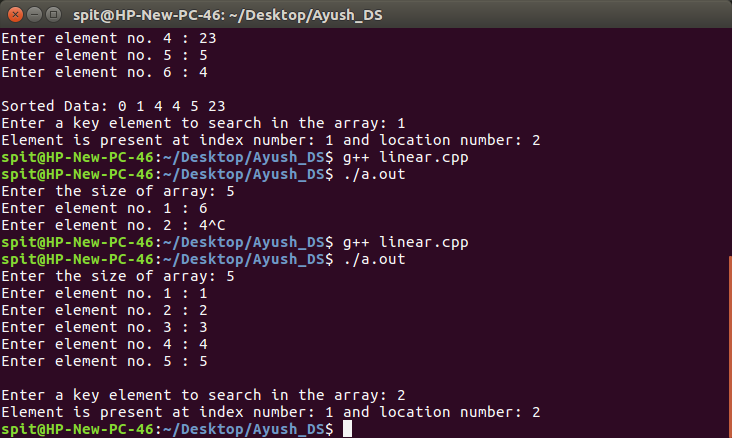
int main(){

searching ob;

ob.linear\_search();

}

**Output:**



**2.2 Binary\_Search.cpp**

#include<iostream>

using namespace std;

class searching{

private:

int size, arr[100], key, i, high, mid, low, temp;

bool found;

public:

void getdata(){

cout <<"Enter the size of array: ";

cin >> size;

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

cout << "Enter element no. " << i+1 << " : ";

cin >> arr[i];

}

cout << endl;

}

void binary\_search(){

getdata();

sortdata();

cout << "Sorted Data: ";

display();

cout << "Enter a key element to search in the array: ";

cin >> key;

high = size-1;

low = 0;

found = false;

while(low<=high){

mid = (low+high)/2;

if(arr[mid]==key){

found = true;

break;

}

else if(arr[mid]>key)

high = --mid;

else

low = ++mid;

}

if(found)

cout << "Element is present at index number: " << mid << " and location number: " << mid+1 << endl;

else

cout << "Element not found." << endl;

}

void display(){

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

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

cout << endl;

}

void sortdata(){

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

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

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

temp = arr[j];

arr[j] = arr[j+1];

arr[j+1] = temp;

}

}

}

}

};

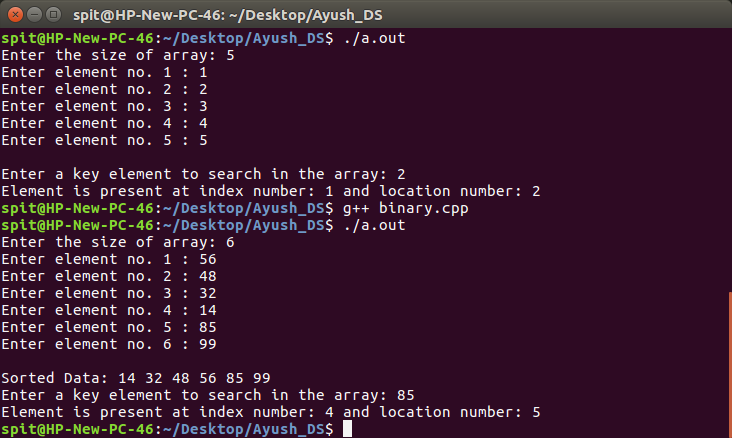
int main(){

searching ob;

ob.binary\_search();

}

**Output:**

****