Binary search using recursion

```
#include<iostream>
using namespace std;
void binary_search(int[],int,int,int);
void bubblesort(int[],int);
int main()
{
    int key,size;
    int list[25];
    cout<<"enter the size of the list\n";
    cin>>size;
    cout<<"enter elements";
    for(int i=0;i<size;i++)
    cin>>list[i];
    bubblesort(list,size);
    cout<<"\n";
    cout<<"enter key element to search";
    cin>>key;
    binary_search(list,0,size,key);
    return 0;
}
void bubblesort(int list[],int size)
{
    int temp,j,i;
    for(i=0;i<size-1;i++)
    {
        for(j=1;j<size-i-1;j++)
        {
            if(list[j]>list[j+1])
            {
```

```
            if(list[j]>list[j+1])
            {
            temp=list[j];
            list[j]=list[j+1];
            list[j+1]=temp;
            }
        }
    }
}
void binary_search(int list[],int lo,int hi,int key)
{
    int mid;
    if(lo>hi)
    {
        cout<<"key not found\n";
        return;
    }
    mid=(lo+hi)/2;
    if(list[mid]==key)
        cout<<"key found at position:"<<(mid+1)<<endl;
    else if(list[mid]>key)
    binary_search(list,lo,mid-1,key);
    else if(list[mid]<key)
    binary_search(list,mid+1,hi,key);
}
```

Output:

```
enter the size of the list
10
enter elements
5
4
6
8
7
2
3
1
9
10

enter key element to search
8
key found at position:8


...Program finished with exit code 0
Press ENTER to exit console.
```