SOURCE CODE

```cpp
//Time Compexity :
// best case: O(nlogn)
// worst case: O(n^2)
#include<iostream>
using namespace std;

void Swap(int *a,int i,int j){
    int temp=a[i];
    a[i]=a[j];
    a[j]=temp;
}
```

```cpp
int partition(int* arr,int l,int r){
    int pivot=arr[r];
    int i=-1;
    for(int j=0;j<r;j++){
        if(arr[j]<pivot){
            i++;
            Swap(arr,i,j);
        }
    }
    Swap(arr,i+1,r);
    return i+1;
}
void QuickSort(int *arr,int l,int r){
    if(l<r){
        int pi=partition(arr,l,r);
        QuickSort(arr,l,pi-1);
        QuickSort(arr,pi+1,r);
    }
}
```

```cpp
int main(){
    int n,*arr;
    cout<<"Enter the size: ";
    cin>>n;
    arr=new int[n];
    cout<<"Enter the elements: ";
    for(int i=0;i<n;i++){
        cin>>arr[i];
    }
    QuickSort(arr,0,n-1);
    cout<<"After sorting: ";
    for(int i=0;i<n;i++){
        cout<<arr[i]<<" ";
    }
    cout<<endl;
}
```

OUTPUT

```
PS C:\Users\anil kumar\Documents\anil\.vscode\DataSructure_in_nsut> cd "c:\Users\anil kumar\Documents\anil\.v
cture_in_nsut\" ; if ($?) { g++ -std=c++17 QuickSort.cpp -o QuickSort } ; if ($?) { .\QuickSort }
Enter the size: 5
Enter the elements: 2
4
3
1
5
After sorting: 1 2 3 4 5
PS C:\Users\anil kumar\Documents\anil\.vscode\DataSructure_in_nsut> cd "c:\Users\anil kumar\Documents\anil\.v
cture_in_nsut\" ; if ($?) { g++ -std=c++17 QuickSort.cpp -o QuickSort } ; if ($?) { .\QuickSort }
Enter the size: 10
Enter the elements: 11
44
23
456
78
346
234
45
34
123
After sorting: 11 23 34 44 45 78 123 234 346 456
PS C:\Users\anil kumar\Documents\anil\.vscode\DataSructure_in_nsut>
```