



East West University
Department of Computer Science and Engineering

Course: CSE 246(Algorithms)
Section - 02

Submitted To :	Submitted By :
Dr. Md. Tauhid Bin Iqbal Assistant Professor Department of CSE East West University	Name : Hafsa Ferdousi Student ID : 2023-3-60-321 Department : CSE Date : 19 / 10 / 2025

Lab Report:03

Insertion sort

```
#include <bits/stdc++.h>
using namespace std;
void insertion(int arr[],int n);
void print(int arr[],int n);
int main(){
    int n;
    cout<<"Size of array: ";
    cin >>n;
    cout <<"Element: ";
    int arr[n];
    for(int i=0;i<n;i++){
        cin>>arr[i];
    }
    insertion( arr, n);
    print( arr, n);

}

void insertion(int arr[],int n){
    for(int i=1;i<n;i++){
        int key=arr[i];
        int j=i-1;

        while(j>=0 && arr[j]>key){
            arr[j+1]=arr[j];
            j=j-1;
        }
        arr[j+1]=key;
    }
}
```

```

}

void print(int arr[],int n){
    cout<<"Sorted array: ";
    for(int i=0;i<n;i++){
        cout<<arr[i]<<" ";

    }

}

```

Output:

```

Size of array: 5
Element: 34 56 7 3 12
Sorted array: 3 7 12 34 56
Process returned 0 (0x0)    execution time : 17.889 s
Press any key to continue.
|

```

Merge sort

```

#include <bits/stdc++.h>
using namespace std;

void merge(int arr[],int left,int mid,int right){
    int n1=mid-left+1;
    int n2=right-mid;

    //create temp array
    int l[n1],r[n2];

```

```

//copy data to temp array
for(int i=0;i<n1;i++){
    l[i]=arr[left+i];
}
for(int j=0;j<n2;j++){
    r[j]=arr[mid+1+j];
}
int i=0,j=0,k=left;

//merge the temp array
while(i<n1 && j<n2){
    if(l[i] <= r[j]){
        arr[k] = l[i];
        i++;
    } else {
        arr[k] = r[j];
        j++;
    }

    k++;

}

// Copy remaining elements
while (i < n1) {
    arr[k] = l[i];
    i++;
    k++;
}
while (j < n2) {
    arr[k] = r[j];
    j++;
    k++;
}

}

```

```

//recursive merge sort
void mergesort(int arr[],int left,int right){
    if(left>=right)
        return;

    int mid=left+(right-left)/2;
    mergesort(arr,left,mid);
    mergesort(arr,mid+1,right);
    merge(arr,left,mid,right);
}

void print(int arr[],int n){
    cout<<"Sorted array: ";
    for(int i=0;i<n;i++){
        cout<<arr[i]<<" ";

    }

}

int main(){

    int n;
    cout<<"Size of array: ";
    cin >>n;
    cout <<"Element: ";
    int arr[n];
    for(int i=0;i<n;i++){
        cin>>arr[i];
    }
    mergesort(arr,0,n-1);
    print(arr,n);
}

```

Output:

```
Size of array: 5
Element: 65 98 3 100 5
Sorted array: 3 5 65 98 100
Process returned 0 (0x0)    execution time : 20.219 s
Press any key to continue.
```

Quick Sort

```
#include <iostream>
using namespace std;

// Partition function
int partition(int arr[], int low, int high) {
    int pivot = arr[high]; // choose pivot
    int i = low - 1;       // index of smaller element
    for (int j = low; j <= high - 1; j++) {
        if (arr[j] < pivot) {
            i++;
            swap(arr[i], arr[j]);
        }
    }
    swap(arr[i + 1], arr[high]); // place pivot in correct position
    return i + 1;               // return pivot index
}

// QuickSort function
void quickSort(int arr[], int low, int high) {
    if (low < high) {
        int pi = partition(arr, low, high);

        quickSort(arr, low, pi - 1); // sort left side
    }
}
```

```

        quickSort(arr, pi + 1, high); // sort right side
    }
}

void print(int arr[],int n){
    cout<<"Sorted array: ";
    for(int i=0;i<n;i++){
        cout<<arr[i]<<" ";

    }

}

int main(){

    int n;
    cout<<"Size of array: ";
    cin >>n;
    cout <<"Element: ";
    int arr[n];
    for(int i=0;i<n;i++){
        cin>>arr[i];
    }
    quickSort(arr,0,n-1);
    print(arr,n);
}

```

```

Size of array: 6
Element: 23 5 67 91 21 56
Sorted array: 5 21 23 56 67 91
Process returned 0 (0x0)   execution time : 18.271 s
Press any key to continue.
|

```