Practical-3

AIM: Implement Sorting Algorithm(s).

(a) Bubble Sort

(b) Selection Sort

(c) Insertion Sort

- Program

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

void bubbleSort(vector<int> vec, int x){
    int exch;
    int last = x-1;
    cout << "Enter elements : ";
    for(int i=1; i<=x; i++){
        exch = 0;
        for(int j=0; j<last; j++){
            if(vec[j] > vec[j+1]){
                swap(vec[j], vec[j+1]);
                exch++;
            }
        }
        if(exch==0)
            break;
        else
            last--;
    }
    for(int i = 0; i < vec.size(); i++){
        cout << vec[i] << " ";
    }
}

void selectionSort(vector<int> vec, int n){
    int exch, min_index;
    for(int i=0; i < n-1; i++){
        min_index=i;
        for(int j=i+1; j < n; j++){
            if(vec[min_index] > vec[j]){
```

```
                min_index = j;
            }
        }
        if(min_index != i)
            swap(vec[i], vec[min_index]);
    }
    for(int i=0; i < n; i++){
        cout << vec[i] << " ";
    }
}

void insertionSort(vector<int> vec, int n){
    int key, j;
    for(int i=0; i<n; i++){
        key = vec[i];
        j = i - 1;
        while(j>=0 && vec[j] > key){
            vec[j+1] = vec[j];
            j--;
        }
        vec[j+1] = key;
    }
    cout << "After Sorting the vector the elemets are : ";
    for(int i=0; i<n; i++){
        cout << vec[i] << " ";
    }
}

int main(){
    cout << "This program is developed by
22CE097_ShivangPatel" << endl;
    int x, a;
    cout << "Enter number of element : ";
```

```
        cin >> x;
        vector<int> vec;
        for(int i=0;i<x;i++){
            cout << "Enter Element Array [" << i << "] : ";
            cin >> a;
            vec.push_back(a);
        }
        int choice;
        cout << "1. Bubble Sort" << endl;
        cout << "2. Selecting Sort" << endl;
        cout << "3. Insertion Sort" << endl;
        cout << "Select number of selection of sorting method :
";
        cin >> choice;
        switch (choice)
        {
            case 1:
                bubbleSort(vec, x); break;

            case 2:
                selectionSort(vec, x); break;

            case 3:
                insertionSort(vec, x); break;

            default:
                break;
        }
        return 0;
    }
```

Output

```
This program is developed by 22CE097_ShivangPatel
Enter number of element : 5
Enter Element Array [0] : 5
Enter Element Array [1] : 10
Enter Element Array [2] : 4
Enter Element Array [3] : 2
Enter Element Array [4] : 54
1. Bubble Sort
2. Selecting Sort
3. Insertion Sort
Select number of selection of sorting method : 1
Enter elements : 2 4 5 10 54
```

```
This program is developed by 22CE097_ShivangPatel
Enter number of element : 5
Enter Element Array [0] : 5
Enter Element Array [1] : 10
Enter Element Array [2] : 4
Enter Element Array [3] : 2
Enter Element Array [4] : 54
1. Bubble Sort
2. Selecting Sort
3. Insertion Sort
Select number of selection of sorting method : 2
2 4 5 10 54
```

```
This program is developed by 22CE097_ShivangPatel
Enter number of element : 5
Enter Element Array [0] : 5
Enter Element Array [1] : 10
Enter Element Array [2] : 4
Enter Element Array [3] : 2
Enter Element Array [4] : 54
1. Bubble Sort
2. Selecting Sort
3. Insertion Sort
Select number of selection of sorting method : 3
After Sorting the vector the elemets are : 2 4 5 10 54
```

- Conclusion

_____ _____ _____

Student Signature Faculty Signature Marks