



PIMPRI CHINCHWAD EDUCATION TRUST'S.
PIMPRI CHINCHWAD COLLEGE OF ENGINEERING
(An Autonomous Institute)

Class : SY BTech**Acad. Yr. 2025-26****Semester : I****Name of the student:** Samruddhi Ramswami Bansode.**PRN : 124B1B019****Department:** Computer Engineering**Division : A****Course Name :** Data Structure Lab**Course Code:** BCE23PC02**Completion Date :**

Assignment No. 1

Problem Statement:

A manufacturing plant collects temperature readings from sensors every second. For analysis, the system needs to sort recent temperature readings to detect anomalies. Write a program for the above scenario.

Hint:

Implement Insertion Sort to sort a stream of sensor temperature data (integers or floats) collected over a period. The sorting algorithm should be efficient enough to run in near real-time on large batches of data.

Source Code :

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

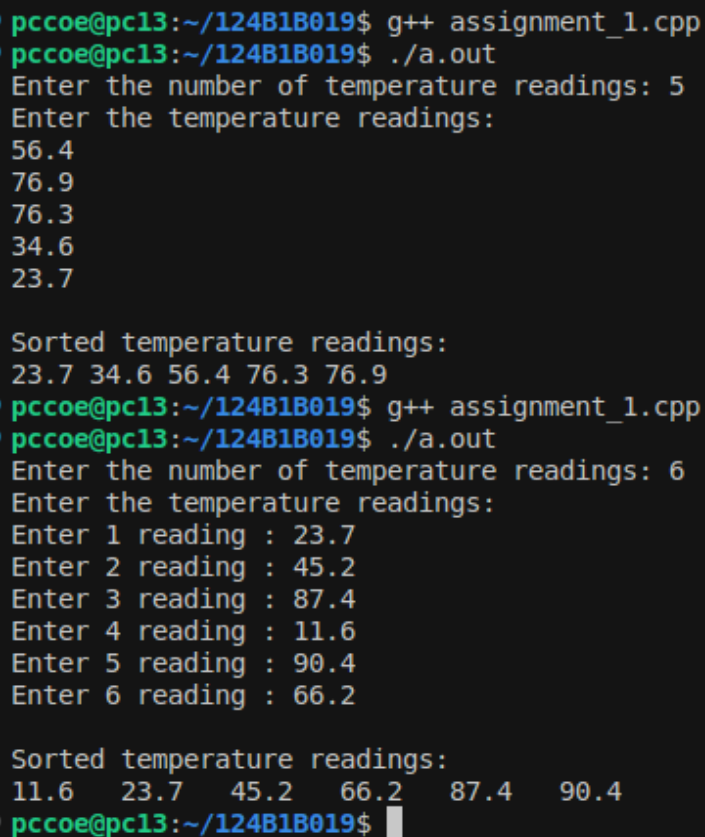
void insertion_sort(float arr[], int n){
    for(int i = 1; i < n; i++) {
        float key = arr[i];
        int j = i - 1;
        while(j >= 0 && arr[j] > key) {
            arr[j + 1] = arr[j];
            j--;
        }
        arr[j + 1] = key;
    }
}
```

```
    }

int main() {
    int n;
    cout << "Enter the number of temperature readings: ";
    cin >> n;
    float arr[n];
    cout << "Enter the temperature readings:\n";
    for(int i = 0; i < n; i++) {
        cin >> arr[i];
    }

    insertion_sort(arr, n);
    cout << "\nSorted temperature readings:\n";
    for(int i = 0; i < n; i++) {
        cout << arr[i] << " ";
    }
    cout << endl;
return 0;
}
```

Screen Shot of Output :



```
pccoe@pc13:~/124B1B019$ g++ assignment_1.cpp
pccoe@pc13:~/124B1B019$ ./a.out
Enter the number of temperature readings: 5
Enter the temperature readings:
56.4
76.9
76.3
34.6
23.7

Sorted temperature readings:
23.7 34.6 56.4 76.3 76.9
pccoe@pc13:~/124B1B019$ g++ assignment_1.cpp
pccoe@pc13:~/124B1B019$ ./a.out
Enter the number of temperature readings: 6
Enter the temperature readings:
Enter 1 reading : 23.7
Enter 2 reading : 45.2
Enter 3 reading : 87.4
Enter 4 reading : 11.6
Enter 5 reading : 90.4
Enter 6 reading : 66.2

Sorted temperature readings:
11.6 23.7 45.2 66.2 87.4 90.4
pccoe@pc13:~/124B1B019$
```

Conclusion:

This C++ program sorts a list of temperature readings using the insertion sort algorithm. It begins by taking input from the user for the number of readings and their values. The `insertion()` function then arranges the readings in ascending order. After sorting, the program displays the sorted temperature values. Overall, it demonstrates a simple and effective use of insertion sort for floating-point data.