









Code (Insertion Sort)

#include <iostream>

#include <chrono>

#include <bits/stdc++.h>

using namespace std;

using namespace std::chrono;

int main()

{

    int n, i, j, key;

    cout << "Enter the Number of Elements: ";

    cin >> n;

    int arr[n];

    cout << "Enter the Elements: ";

    for (i = 0; i < n; i++)

    {

        cin >> arr[i];

    }

    auto start = steady\_clock::now();

    for (i = 0; i < n; i++)

    {

        key = arr[i];

        j = i - 1;

        while (j >= 0 && arr[j] > key)

        {

            arr[j + 1] = arr[j];

            j = j - 1;

        }

        arr[j + 1] = key;

    }

    auto stop = steady\_clock::now();

    cout << "Sorted array is: ";

    for (j = 0; j < n; j++)

    {

        cout << arr[j] << " ";

    }

    auto duration = duration\_cast<nanoseconds>(stop - start);

    cout << "\nTime taken by function: " << duration.count() << " nanoseconds" << endl;

}

Output (Insertion Sort)

Best Case



Worst Case



Code (Bubble Sort)

#include <iostream>

#include <chrono>

#include <bits/stdc++.h>

using namespace std;

using namespace std::chrono;

int main()

{

    int i, j, k, n, temp;

    cout << "Enter Number of Elements: ";

    cin >> n;

    int arr[n];

    cout << "Enter the elements of array: ";

    for (i = 0; i < n; i++)

    {

        cin >> arr[i];

    }

    auto start = steady\_clock::now();

    for (i = 0; i < n; i++)

    {

        for (j = i; j < n; j++)

        {

            if (arr[i] > arr[j])

            {

                temp = arr[j];

                arr[j] = arr[i];

                arr[i] = temp;

            }

        }

    }

    auto stop = steady\_clock::now();

    cout << "Sorted Array is: ";

    for (i = 0; i < n; i++)

    {

        cout << arr[i] << " ";

    }

    auto duration = duration\_cast<nanoseconds>(stop - start);

    cout << "\nTime taken by function: " << duration.count() << " nanoseconds" << endl;

}

Output (Bubble Sort)

Best Case



Worst case



Code (Selection Sort)

#include <iostream>

#include <chrono>

using namespace std;

using namespace std::chrono;

int main()

{

    int i, j, n, min, temp, k;

    cout << "Enter the Number of Elements: ";

    cin >> n;

    int arr[n];

    cout << "Enter the elements: ";

    for (i = 0; i < n; i++)

    {

        cin >> arr[i];

    }

    auto start = steady\_clock::now();

    for (i = 0; i < n - 1; i++)

    {

        temp = i;

        for (j = i + 1; j < n; j++)

        {

            if (arr[temp] > arr[j])

            {

                temp = j;

            }

        }

        min = arr[temp];

        arr[temp] = arr[i];

        arr[i] = min;

    }

    auto stop = steady\_clock::now();

    cout << "Sorted Array" << endl;

    for (i = 0; i < n; i++)

    {

        cout << arr[i] << " ";

    }

    auto duration = duration\_cast<nanoseconds>(stop - start);

    cout << "\nTime taken by function: " << duration.count() << " nanoseconds" << endl;

}

Output (Selection Sort)

Best Case



Worst Case

