# Week 8 & 9: Sorting

## **Aim**: Implement Insertion Sort.

## Program:

#include <bits/stdc++.h>

using namespace std;

vector<int> arr;

void print(int n) {

    for (int i = 0; i < n; i++)

        cout << arr[i] << " ";

    cout << endl;

}

void insertionSort1(int n) {

    int cnt = 0;

    for (int i = 1; i < n; i++) {

        cnt = 0;

        int tmp = arr[i];

        int j = i - 1;

        for (; j >= 0; j--) {

            if (arr[j] > tmp) {

                cnt = 1;

                arr[j + 1] = arr[j];

                print(n);

            }

            else

                break;

        }

        arr[j + 1] = tmp;

        if (cnt == 1)

            print(n);

    }

}

int main() {

    int n;

    cin >> n;

    arr.resize(n);

    for (int i = 0; i < n; i++) {

        cin >> arr[i];

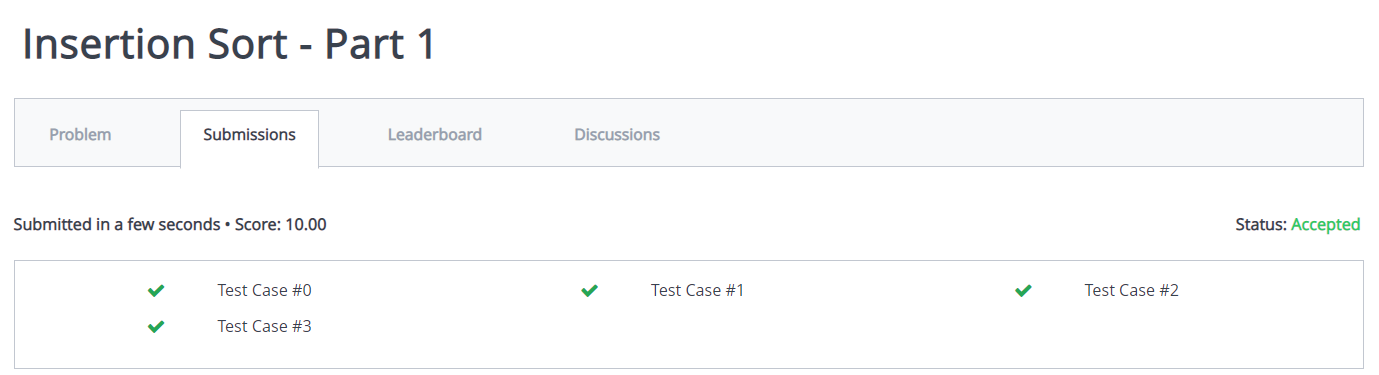
    }

    insertionSort1(n);

*return* 0;

}

### Output:



## **Aim:** Sort Colors

### Program:

class Solution {

public:

    void sortColors(vector<int> *&*nums) {

        int red = 0, white = 0, blue = nums.size() - 1;

        while (white <= blue) {

            if (nums[white] == 0)

                swap(nums[red++], nums[white++]);

            else if (nums[white] == 1)

                white++;

            else

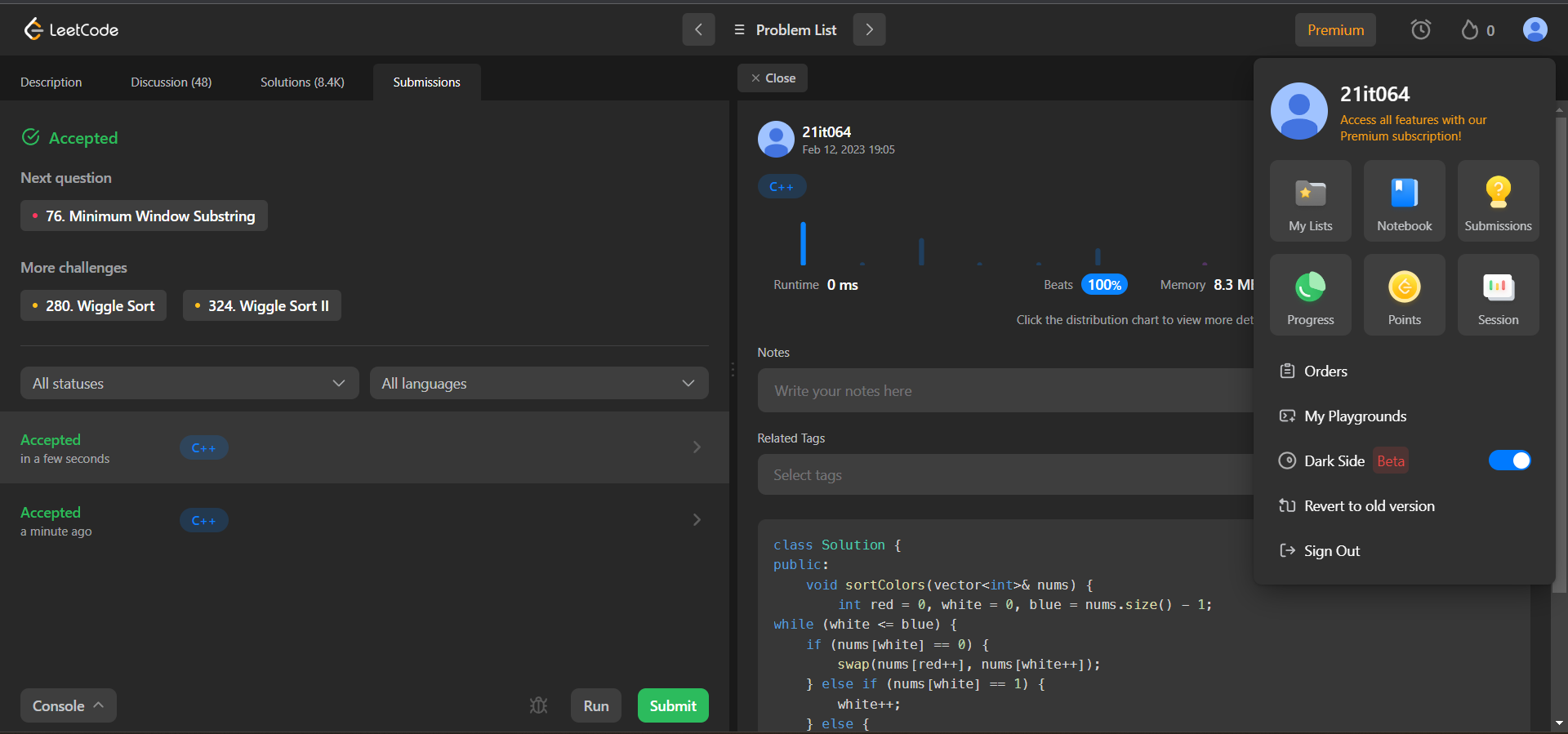
                swap(nums[white], nums[blue--]);

        }

    }

};

### Output:



## **Conclusion:**

Sorting algorithms are used in a variety of applications, including database management, data analysis, and computer graphics.