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: Varad Amol Pisale		**PRN : 124B1B043**
Department: Computer Engineering		**Division : A**
Course Name : Data Structures Laboratory		**Code:** BCE23PC02
Completion Date : 13/08/2025		

_____

Assignment No.

Problem Statement:
An education platform needs to rank students based on their exam scores to publish results and generate merit lists. Write a program for above scenario.
Hint:
Given a list of student scores, use Quick Sort to sort them in descending order. Handle cases where many students have the same score and analyze how this affects the performance of Quick Sort.

Source Code :
```cpp
#include <bits/stdc++.h>
using namespace std;

struct Student
{
   string name;
   double score;
};

// Partition
int partition(vector<Student> &arr, int low, int high)
{
   int pivot = arr[high].score;
   int i = low - 1;

   for (int j = low; j < high; j++)
   {
```

```
      if (arr[j].score > pivot)
      {
         i++;
         swap(arr[i], arr[j]);
      }
   }
   swap(arr[i + 1], arr[high]);
   return i + 1;
}

// Quick Sort
void quickSort(vector<Student> &arr, int low, int high)
{
   if (low < high)
   {
      int pIndex = partition(arr, low, high);

      quickSort(arr, low, pIndex - 1);
      quickSort(arr, pIndex + 1, high);
   }
}

int main()
{
   int n;
   cout << "Enter number of students: ";
   cin >> n;
   cout << endl
        << endl;

   vector<Student> students(n);
   for (int i = 0; i < n; i++)
   {
      cout << "Enter name of student " << i + 1 << ": ";
      getchar();
      getline(cin, students[i].name);
      cout << endl
           << "Enter score of student " << i + 1 << ": ";
      cin >> students[i].score;
      cout << endl
           << endl;
   }

   quickSort(students, 0, n - 1);

   cout << "\n=== Merit List (Descending Order) ===\n";
```
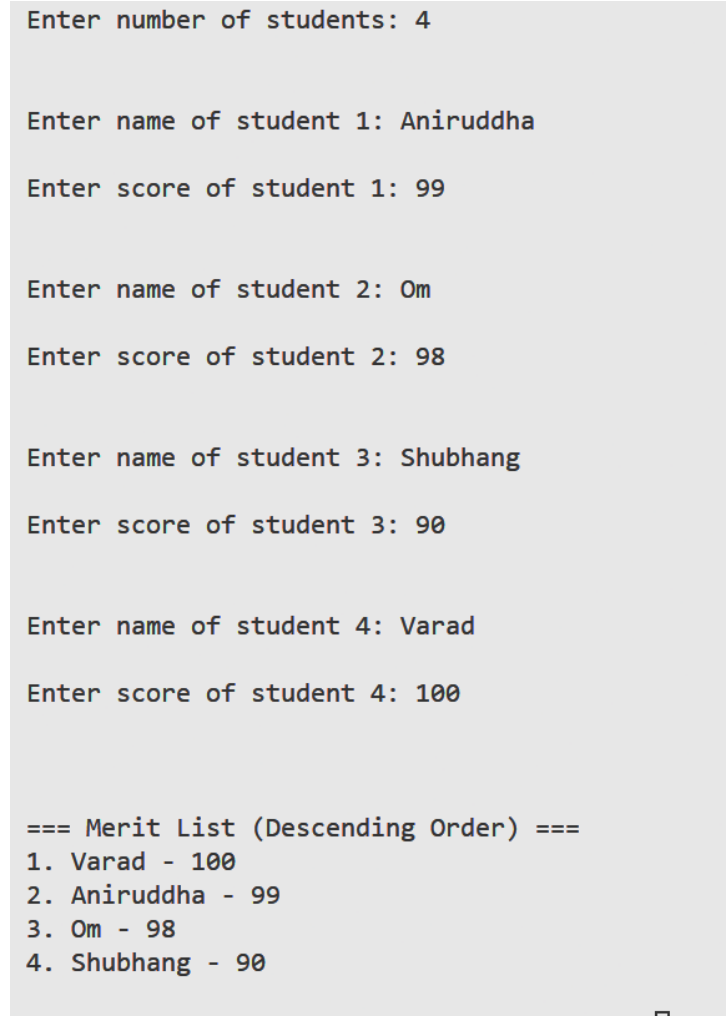
```
for (int i = 0; i < n; i++)
{
    cout << i + 1 << ". " << students[i].name << " - " << students[i].score << endl;
}

return 0;
}
```
Screen Shot of Output :

```
Enter number of students: 4


Enter name of student 1: Aniruddha

Enter score of student 1: 99


Enter name of student 2: Om

Enter score of student 2: 98


Enter name of student 3: Shubhang

Enter score of student 3: 90


Enter name of student 4: Varad

Enter score of student 4: 100



=== Merit List (Descending Order) ===
1. Varad - 100
2. Aniruddha - 99
3. Om - 98
4. Shubhang - 90
```

Conclusion: Hence we have implemented an education platform needs to rank students based on their exam scores to publish results and generate merit lists.