BRACT’s

Vishwakarma Institute of Information Technology, Pune

**Practical Implementation Sheet**

| **Department:** IT | **Semester:** IV | **Academic Year:** 2024-25 | **Practical No: 3** |
| --- | --- | --- | --- |
| **Class/ Division/ Batch: SY (B)- B3** | | **Roll no: 70** | |
| **Course:** Data Structures and Analysis of Algorithms | | **Name of Student**: Anushka Kadam | |

**Aim:** a) The internship is offered to students based on rank obtained in the second year of

graduation. Create a suitable non-linear data structure to identify the next topper

student for internship. (Create max-heap).

b) Sort the student data in ascending order of grades

**Code:**

#include <iostream>

#include <vector>

#include <limits>

#include <algorithm>

#include <sstream>

using namespace std;

struct Student

{

int roll\_no;

string name;

int grade;

};

void max\_heapify(vector<Student>& students, int i, int n)

{

int largest = i;

int left = 2 \* i + 1;

int right = 2 \* i + 2;

if (left < n && students[left].grade > students[largest].grade)

largest = left;

if (right < n && students[right].grade > students[largest].grade)

largest = right;

if (largest != i)

{

swap(students[i], students[largest]);

max\_heapify(students, largest, n);

}

}

void build\_max\_heap(vector<Student>& students, int n)

{

for (int i = n / 2 - 1; i >= 0; i--)

max\_heapify(students, i, n);

}

void heap\_sort(vector<Student>& students, int n)

{

build\_max\_heap(students, n);

for (int i = n - 1; i > 0; i--)

{

swap(students[0], students[i]);

max\_heapify(students, 0, i);

}

}

void display\_students(const vector<Student>& students)

{

for (const auto& student : students)

{

cout << "Roll No: " << student.roll\_no

<< ", Name: " << student.name

<< ", Grade: " << student.grade << endl;

}

}

int main()

{

int n;

cout << "Enter the number of students: ";

cin >> n;

vector<Student> students(n);

cout << "Enter student details (Roll No, Name, Grade):" << endl;

cin.ignore();

for (int i = 0; i < n; i++)

{

string line;

getline(cin, line);

istringstream iss(line);

iss >> students[i].roll\_no;

string temp;

iss >> temp;

students[i].name = temp;

while (iss >> temp && isalpha(temp[0]))

{

students[i].name += " " + temp;

}

students[i].grade = temp[0] - 'A';

}

build\_max\_heap(students, n);

cout << "\nTopper for internship (Max-Heap Root): Roll No: "

<< students[0].roll\_no << ", Name: " << students[0].name

<< ", Grade: " << students[0].grade << endl;

heap\_sort(students, n);

reverse(students.begin(), students.end());

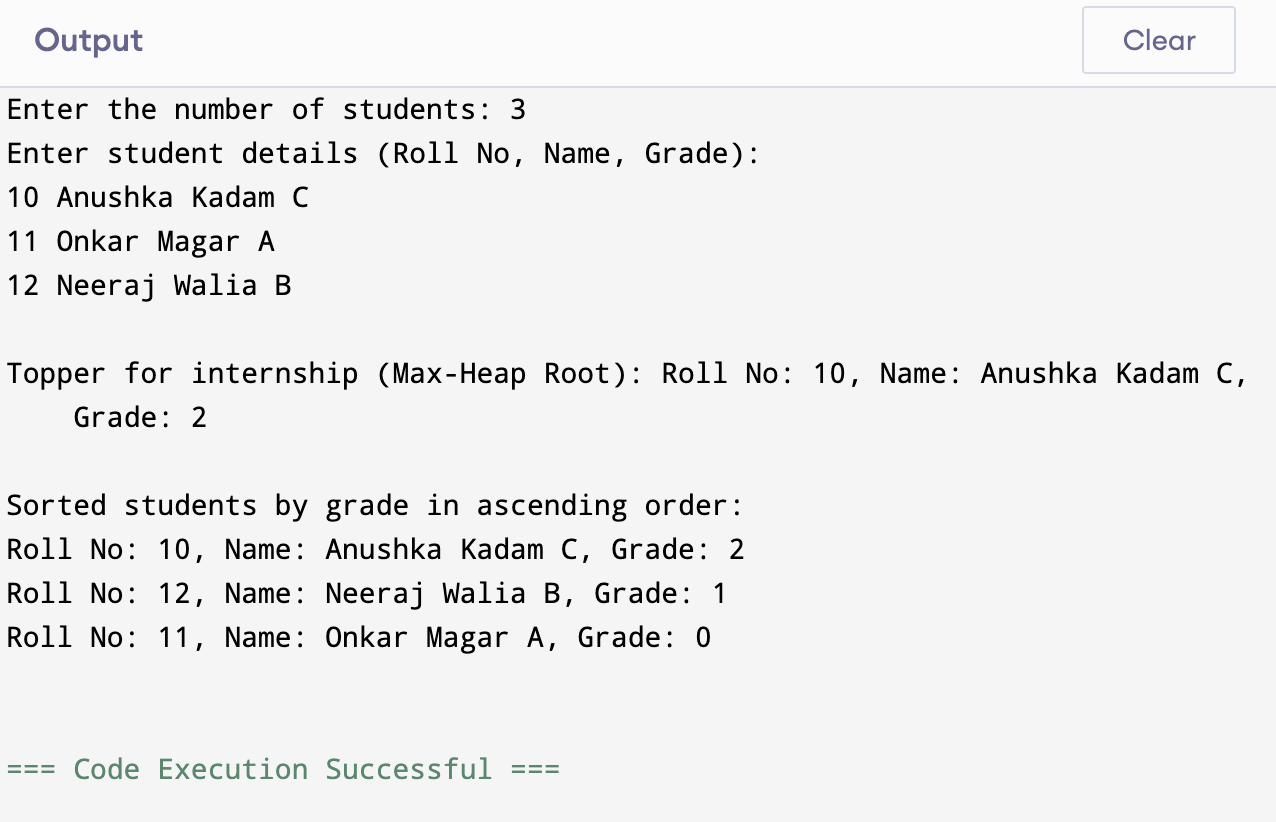
cout << "\nSorted students by grade in ascending order:\n";

display\_students(students);

return 0;

}

**Output:**

****