**Name:** Rohan Arun Nalawade

**Roll No:** 231012

**PRN:** 22310407

**SY IT A**

**Assignment 3**

**Aim:** To implement and understand the concept of heap, heapsort and priority queue with applications in selecting students for internship based on ranks using a max heap and sorting data using heap sort.

#include <iostream>

#include <vector>

using namespace std;

class MaxHeap {

private:

    vector<pair<int, string>> heap;  // Pair of (Rank, Student Name)

    // Heapify function to maintain max-heap property

    void heapifyUp(int index) {

        while (index > 0) {

            int parent = (index - 1) / 2;

            if (heap[parent].first < heap[index].first) {

                swap(heap[parent], heap[index]);

                index = parent;

            } else {

                break;

            }

        }

    }

    void heapifyDown(int index) {

        int size = heap.size();

        while (index < size) {

            int left = 2 \* index + 1;

            int right = 2 \* index + 2;

            int largest = index;

            if (left < size && heap[left].first > heap[largest].first)

                largest = left;S

            if (right < size && heap[right].first > heap[largest].first)

                largest = right;

            if (largest != index) {

                swap(heap[index], heap[largest]);

                index = largest;

            } else {

                break;

            }

        }

    }

public:

    void insertStudent(int rank, string name) {

        heap.push\_back(make\_pair(rank, name));

        heapifyUp(heap.size() - 1);

    }

    void getTopper() {

        if (heap.empty()) {

            cout << "No students available.\n";

            return;

        }

        cout << "Next Topper: " << heap[0].second << " (Rank: " << heap[0].first << ")\n";

        heap[0] = heap.back();

        heap.pop\_back();

        heapifyDown(0);

    }

};

// Driver Code

int main() {

    MaxHeap students;

    int n, rank;

    string name;

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

    cin >> n;

    for (int i = 0; i < n; i++) {

        cout << "Enter student name: ";

        cin >> name;

        cout << "Enter student rank: ";

        cin >> rank;

        students.insertStudent(rank, name);

    }

    cout << "\nExtracting toppers in order:\n";

    for (int i = 0; i < n; i++) {

        students.getTopper();

    }

    return 0;

}

