NUST

NATIONAL UNIVERSITY
OF SCIENCES & TECHNOLOGY

Programming lab manual 10

Name	Hasnain Ali
Reg. No	478806
Section	C

Q.No. 1

```
#include <iostream>
#include <vector>
#include <algorithm>
#include <map>

using namespace std;

// Function to calculate the mean of grades
double calculateMean(const vector<int>& grades) {
    int sum = 0;
    for (int grade : grades) {
        sum += grade;
    }
    return static_cast<double>(sum) / grades.size();
}

// Function to calculate the median of grades
double calculateMedian(const vector<int>& grades) {
    vector<int> sortedGrades = grades;
    sort(sortedGrades.begin(), sortedGrades.end());

    size_t size = sortedGrades.size();
    if (size % 2 == 0) {
        // If the size is even, average the middle two elements
        return (sortedGrades[size / 2 - 1] + sortedGrades[size / 2]) / 2.0;
    } else {
        // If the size is odd, return the middle element
        return sortedGrades[size / 2];
    }
}

// Function to calculate the mode of grades
vector<int> calculateMode(const vector<int>& grades) {
    map<int, int> gradeFrequency;

    // Count the frequency of each grade
    for (int grade : grades) {
        gradeFrequency[grade]++;
    }

    // Find the mode(s)
    int maxFrequency = 0;
    vector<int> modeGrades;

    for (const auto& entry : gradeFrequency) {
        if (entry.second > maxFrequency) {
            maxFrequency = entry.second;
            modeGrades = {entry.first};
        } else if (entry.second == maxFrequency) {
            modeGrades.push_back(entry.first);
        }
    }

    return modeGrades;
}

int main() {
    // Task 1: Iterate Through Vector Using Iterators
    vector<int> numbers = {1, 2, 3, 4};

    cout << "Vector Elements: ";
    for (auto it = numbers.begin(); it != numbers.end(); ++it) {
        cout << *it << " ";
    }
    cout << endl;

    // Add integer 5 and remove element at that position
    numbers.push_back(5);
    auto positionToRemove = find(numbers.begin(), numbers.end(), 5);
    if (positionToRemove != numbers.end()) {
        numbers.erase(positionToRemove);
    }
```

```
// Display modified vector
cout << "Modified Vector Elements: ";
for (int num : numbers) {
   cout << num << " ";
}
cout << endl;
```

Task 2: Names and Grades

```
int numPairs;
cout << "\nEnter the number of name/grade pairs: ";
cin >> numPairs;

vector<string> names;
vector<int> grades;

// Input name/grade pairs
for (int i = 0; i < numPairs; ++i) {
   string name;
   int grade;

   cout << "Enter name for student " << i + 1 << ": ";
   cin >> name;
   names.push_back(name);

   cout << "Enter grade for student " << i + 1 << ": ";
```

```
    cin >> grade;

    grades.push_back(grade);

}


// Display mean

cout << "Mean of the grades: " << calculateMean(grades) << endl;


// Display median

cout << "Median of the grades: " << calculateMedian(grades) << endl;


// Display mode

vector<int> modeGrades = calculateMode(grades);

cout << "Mode of the grades: ";

for (int mode : modeGrades) {

    cout << mode << " ";

}

cout << endl;


// Display names of students with the mode as their grade

cout << "Students with mode as their grade: ";

for (size_t i = 0; i < grades.size(); ++i) {

    if (find(modeGrades.begin(), modeGrades.end(), grades[i]) != modeGrades.end()) {

        cout << names[i] << " ";

    }

}
```

```
    cout << endl;

    return 0;
}
```