Name: Ramzan Sameer

Class: ME-15-SECTION-A

Instructor: Sir Affan

CMS ID: 464899

Date: 24/12/2023

Tasks (Due in last week of December):

1. Iterate Through Vector Using Iterators and print all pushed elements. Next you need to push integer 5 and remove element at that position.

```cpp
#include <iostream>
#include <vector>
using namespace std;
int main() {
    vector<int> mv = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10};
    cout << "The Original Vector is: ";
    for (auto it = mv.begin(); it != mv.end(); ++it) {
        cout << *it << " ";
    }
    cout << endl;
    mv.push_back(5);
    cout << "The Vector after pushing 5 is: ";
    for (auto it = mv.begin(); it != mv.end(); ++it) {
        cout << *it << " ";
    }
    cout << endl;
    int ptr = 2;
    if (ptr >= 0 && ptr < mv.size()) {
        mv.erase(mv.begin() + ptr);
        cout << "Vector after removing element at position" << ptr << ": ";
        for (auto it = mv.begin(); it != mv.end(); ++it) {
            cout << *it << " ";
        }
        cout << endl;
    } else {
        cout << "Invalid position to remove" << endl;
    }
    return 0;
}
```

```
/tmp/83TojZohiV.o
The Original Vector is: 1 2 3 4 5 6 7 8 9 10
The Vector after pushing 5 is: 1 2 3 4 5 6 7 8 9 10 5
Vector after removing element at position2: 1 2 4 5 6 7 8 9 10 5
```

2. Write a complete C++ program that uses 2 vectors, 1 for names (string) and 1 for grades (int)
    a. Ask the user for the number of name/grade pairs that will be entered.
    b. Display the mean of the grades.
    c. Display the median of the grades.
    d. Display the mode of the grades.
    e. Display the names of the students with the mode as their grade.

```cpp
#include <iostream>
#include <vector>
#include <algorithm>
#include <numeric> // Include for accumulate
#include <string>

using namespace std;

double calculateMean(const vector<int>& grades) {
    return accumulate(grades.begin(), grades.end(), 0.0) / grades.size();
}

double calculateMedian(vector<int>& grades) {
    sort(grades.begin(), grades.end());
    size_t size = grades.size();
    return (size % 2 == 0) ? (grades[size / 2 - 1] + grades[size / 2]) / 2.0 :
        grades[size / 2];
}

vector<int> calculateMode(const vector<int>& grades) {
    vector<int> sortedGrades = grades;
    sort(sortedGrades.begin(), sortedGrades.end());

    vector<int> modes;
    int currentNumber = sortedGrades[0];
    int currentCount = 1;
    int maxCount = 1;

```

```
    for (size_t i = 1; i < sortedGrades.size(); ++i) {
        if (sortedGrades[i] == currentNumber) {
            currentCount++;
        } else {
            if (currentCount > maxCount) {
                maxCount = currentCount;
                modes.clear();
                modes.push_back(currentNumber);
            } else if (currentCount == maxCount) {
                modes.push_back(currentNumber);
            }

            currentNumber = sortedGrades[i];
            currentCount = 1;
        }
    }

    if (currentCount > maxCount) {
        modes.clear();
        modes.push_back(currentNumber);
    } else if (currentCount == maxCount) {
        modes.push_back(currentNumber);
    }

    return modes;
}
```

```cpp
}

int main() {
    // Get the number of name/grade pairs
    int numPairs;
    cout << "Enter the number of name/grade pairs: ";
    cin >> numPairs;

    // Input names and grades
    vector<string> names(numPairs);
    vector<int> grades(numPairs);

    for (int i = 0; i < numPairs; ++i) {
        cout << "Enter name #" << i + 1 << ": ";
        cin >> names[i];

        cout << "Enter grade #" << i + 1 << ": ";
        cin >> grades[i];
    }

    // Display statistics
    cout << "Mean of the grades: " << calculateMean(grades) << endl;
    cout << "Median of the grades: " << calculateMedian(grades) << endl;

    vector<int> modes = calculateMode(grades);
    cout << "Mode of the grades: ";
    for (int mode : modes) {
        cout << mode << " ";
```

```
// Display statistics
cout << "Mean of the grades: " << calculateMean(grades) << endl;
cout << "Median of the grades: " << calculateMedian(grades) << endl;

vector<int> modes = calculateMode(grades);
cout << "Mode of the grades: ";
for (int mode : modes) {
    cout << mode << " ";
}
cout << endl;

cout << "Names of students with the mode as their grade: ";
for (size_t i = 0; i < grades.size(); ++i) {
    for (int mode : modes) {
        if (grades[i] == mode) {
            cout << names[i] << " ";
        }
    }
}
cout << endl;

return 0;
```

Output:

```
/tmp/83TojZohiV.o
Enter the number of name/grade pairs: 2
Enter name #1: Ramzan
Enter grade #1: 78
Enter name #2: Muzammil
Enter grade #2: 45
Mean of the grades: 61.5
Median of the grades: 61.5
Mode of the grades: 45 78
Names of students with the mode as their grade: Ramzan Muzammil
```

3. Write a program to print the area and perimeter of a triangle having sides of 3 m, 4 m and 5 m by creating a class named 'Triangle' with a function to print the area and perimeter.

```cpp
#include <iostream>
#include <cmath>
using namespace std;
class Triangle {
private:
    double side1;
    double side2;
    double side3;
public:
    Triangle(double s1, double s2, double s3) : side1(s1), side2(s2), side3(s3) {}
    double calculateArea() const {
        double s = (side1 + side2 + side3) / 2.0;
        return sqrt(s * (s - side1) * (s - side2) * (s - side3));
    }
    double calculatePerimeter() const {
        return side1 + side2 + side3;
    }
    void printInfo() const {
        cout << "Triangle with sides: " << side1 << " m, " << side2 << " m, " << side3 << " m" << endl;
        cout << "Area: " << calculateArea() << " square meters" << endl;
        cout << "Perimeter: " << calculatePerimeter() << " meters" << endl;
    }
};
int main() {
    Triangle myTriangle(3.0, 4.0, 5.0);
    myTriangle.printInfo();
    return 0;
}
```

```
C:\Users\HP\Desktop\Rubbish
Triangle with sides: 3 m, 4 m, 5 m
Area: 6 square meters
Perimeter: 12 meters

--------------------------------
Process exited after 0.08353 seconds with return value 0
Press any key to continue . . .
```

4. Write a structure to store the names, salary, and hours of work per day of 10 employees in a company. Write a program to increase the salary depending on the number of hours of work per day as follows and then print the name of all the employees along with their final salaries.

```cpp
#include <iostream>
#include <iomanip>
#include <string>
using namespace std;
//N_E = num employee
const int N_E = 10;
struct Employee {
    string name;
    double salary;
//    hw = hours worked
    int hw;
};
void adjustSalary(Employee& emp) {
    if (emp.hw >= 12) {
        emp.salary += 150.0;
    } else if (emp.hw >= 10) {
        emp.salary += 100.0;
    } else if (emp.hw >= 8) {
        emp.salary += 50.0;
    }
}
int main() {
    Employee employees[N_E];
    for (int i = 0; i < N_E; ++i) {
        cout << "Please Enter the name for employee " << i + 1 << ": ";
        cin >> employees[i].name;
        cout << "Please Enter  the salary for employee " << i + 1 << ": ";
```

```
        emp.salary += 50.0;
    }
}
int main() {
    Employee employees[N_E];
    for (int i = 0; i < N_E; ++i) {
        cout << "Please Enter the name for employee " << i + 1 << ": ";
        cin >> employees[i].name;
        cout << "Please Enter  the salary for employee " << i + 1 << ": ";
        cin >> employees[i].salary;
        cout << "Enter the  hours worked per day for employee " << i + 1 << ": ";
        cin >> employees[i].hw;
        adjustSalary(employees[i]);
    }
    cout << "\nEmployee Information:\n";
    cout << "-----------------------------------------\n";
    cout << setw(20) << "Name" << setw(15) << "Final Salary" << "\n";
    cout << "-----------------------------------------\n";
    for (int i = 0; i < N_E; ++i) {
        cout << setw(20) << employees[i].name
             << setw(15) << employees[i].salary << "\n";
    }
    cout << "-----------------------------------------\n";
    return 0;
}
```

```
Please Enter the name for employee 1: jack
Please Enter  the salary for employee 1: 500
Enter the  hours worked per day for employee 1: 15
Please Enter the name for employee 2: paul
Please Enter  the salary for employee 2: 3400
Enter the  hours worked per day for employee 2: 10
Please Enter the name for employee 3: ahmad
Please Enter  the salary for employee 3: 5000
Enter the  hours worked per day for employee 3: 2
Please Enter the name for employee 4: jacob
Please Enter  the salary for employee 4: 60000
Enter the  hours worked per day for employee 4: 2
Please Enter the name for employee 5: ramzan
Please Enter  the salary for employee 5: 100000
Enter the  hours worked per day for employee 5: 1
Please Enter the name for employee 6: muzammil
Please Enter  the salary for employee 6: 500000
Enter the  hours worked per day for employee 6: 2
Please Enter the name for employee 7: ahad
Please Enter  the salary for employee 7: 890
Enter the  hours worked per day for employee 7: 15
Please Enter the name for employee 8: kate
Please Enter  the salary for employee 8: 780
Enter the  hours worked per day for employee 8: 9
Please Enter the name for employee 9: tate
Please Enter  the salary for employee 9: 8000
Enter the  hours worked per day for employee 9: 10
```

```
Please Enter the name for employee 7: ahad
Please Enter  the salary for employee 7: 890
Enter the  hours worked per day for employee 7: 15
Please Enter the name for employee 8: kate
Please Enter  the salary for employee 8: 780
Enter the  hours worked per day for employee 8: 9
Please Enter the name for employee 9: tate
Please Enter  the salary for employee 9: 8000
Enter the  hours worked per day for employee 9: 10
Please Enter the name for employee 10: james
Please Enter  the salary for employee 10: 10000
Enter the  hours worked per day for employee 10: 9
Employee Information:
-----------------------------------------
                Name   Final Salary
-----------------------------------------
                jack            650
                paul           3500
               ahmad           5000
               jacob          60000
              ramzan         100000
            muzammil         500000
                ahad           1040
                kate            830
                tate           8100
               james          10050
-----------------------------------------
```