C++ Mini Project Documentation

Title of the Project

Online Feedback System Using C++

Group Details

- Group Number: 5

Team Members:

  - Raj Rasal
  - Kartik Wagh
  - Ajit Singh
  - Pallavi Sarovar

Introduction

This mini project is a console-based Online Feedback System developed using the C++ programming language. The main objective of this project is to collect user ratings and comments, validate the input, and display a comprehensive summary with an average rating. The project helps in understanding the fundamental concepts of C++ including loops, string handling, input/output operations, and data aggregation.

Objectives of the Project

- To design a feedback collection system using C++
- To understand and apply do-while loops for repeated input
- To implement input validation for ratings and menu choices
- To handle user text input efficiently using getline()
- To calculate and display the average rating from collected data

Scope of the Project

This feedback system is suitable for beginners learning C++. It collects numerical ratings (1–5) and text comments from multiple users in a single session. The project can be extended in the future to include file storage, user authentication, or a graphical user interface.

Tools and Technologies Used

- Programming Language: C++
- Compiler: GCC / Turbo C++ / Any standard C++ compiler
- Platform: Console-based application

Program Description

The program uses a do-while loop to repeatedly collect a rating and comment from the user. After each entry, the user is asked whether they wish to submit more feedback. Once all entries are collected, the program displays a formatted summary showing each rating and comment along with the total count and average rating.

Concepts Used

- Variables and Data Types (int, double, string, char)
- do-while Loop
- while Loop for Input Validation
- String Concatenation and to_string()
- Input/Output Streams (cin, cout, getline)
- Arithmetic Operations (average calculation)
- Conditional Statements

Code Structure Explanation

Header Files

```
#include <iostream>
```

Used for input and output operations.

```
#include <string>
```

Used for string variables and the to_string() function.

Namespace

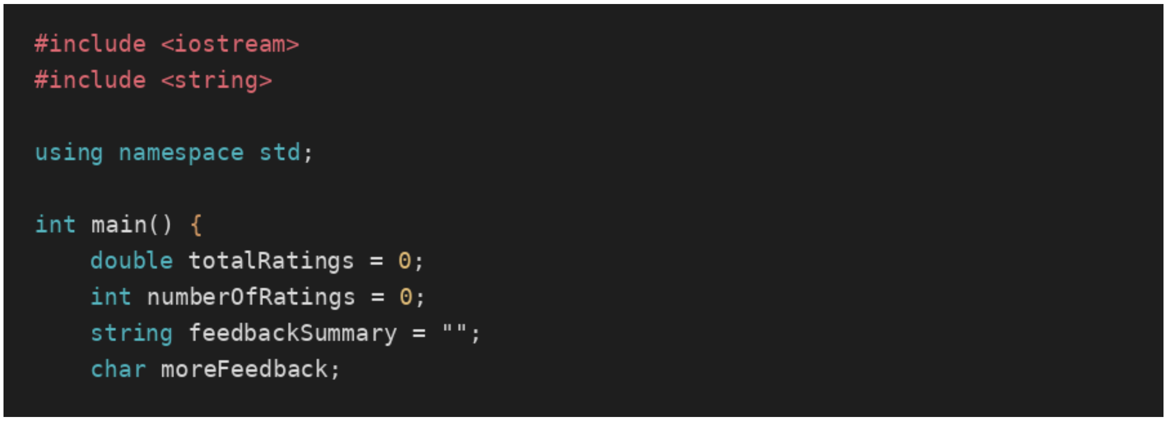
```
using namespace std;
```

Allows the use of standard library objects without the std:: prefix.

Variable Declarations

The main() function declares the following variables before the loop:

- totalRatings (double) — accumulates the sum of all ratings
- numberOfRatings (int) — counts how many entries were submitted
- feedbackSummary (string) — stores all formatted feedback lines
- moreFeedback (char) — controls whether the loop continues

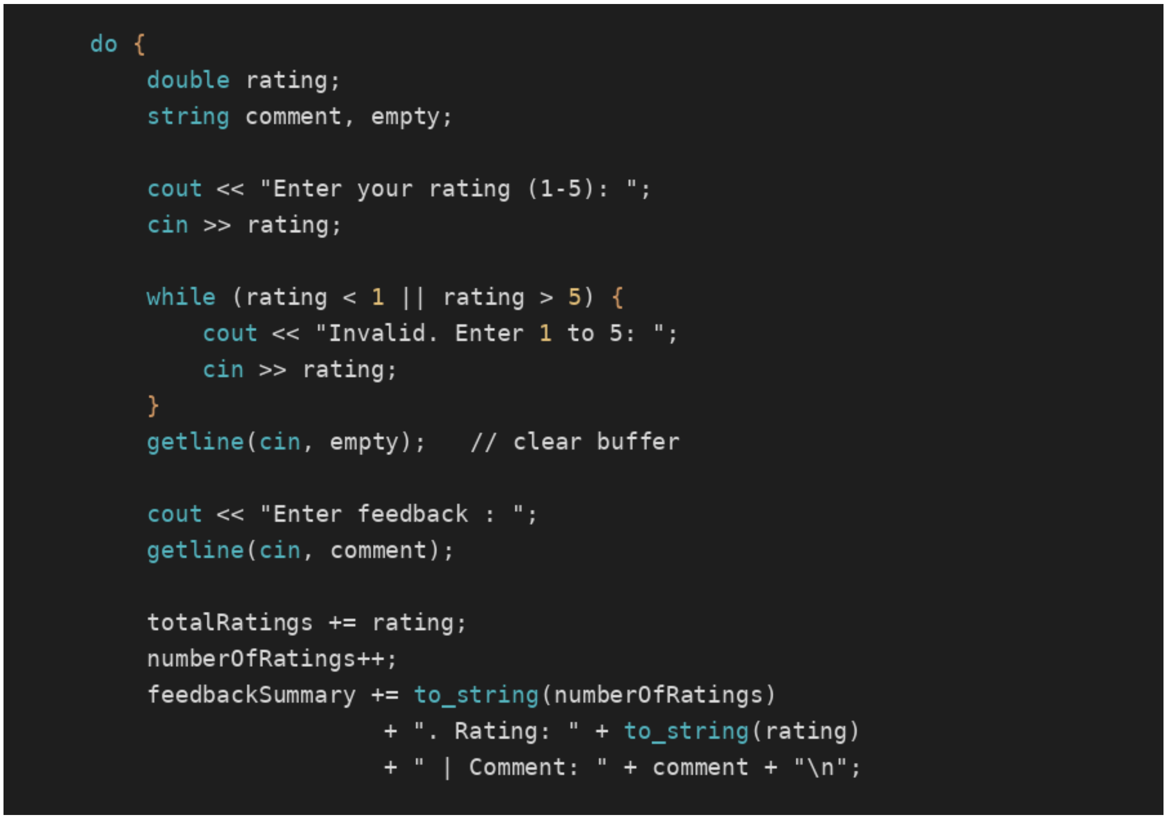


```cpp
#include <iostream>
#include <string>

using namespace std;

int main() {
    double totalRatings = 0;
    int numberOfRatings = 0;
    string feedbackSummary = "";
    char moreFeedback;
```

Input Loop Explanation

The do-while loop is the core of the program. Steps performed inside the loop:

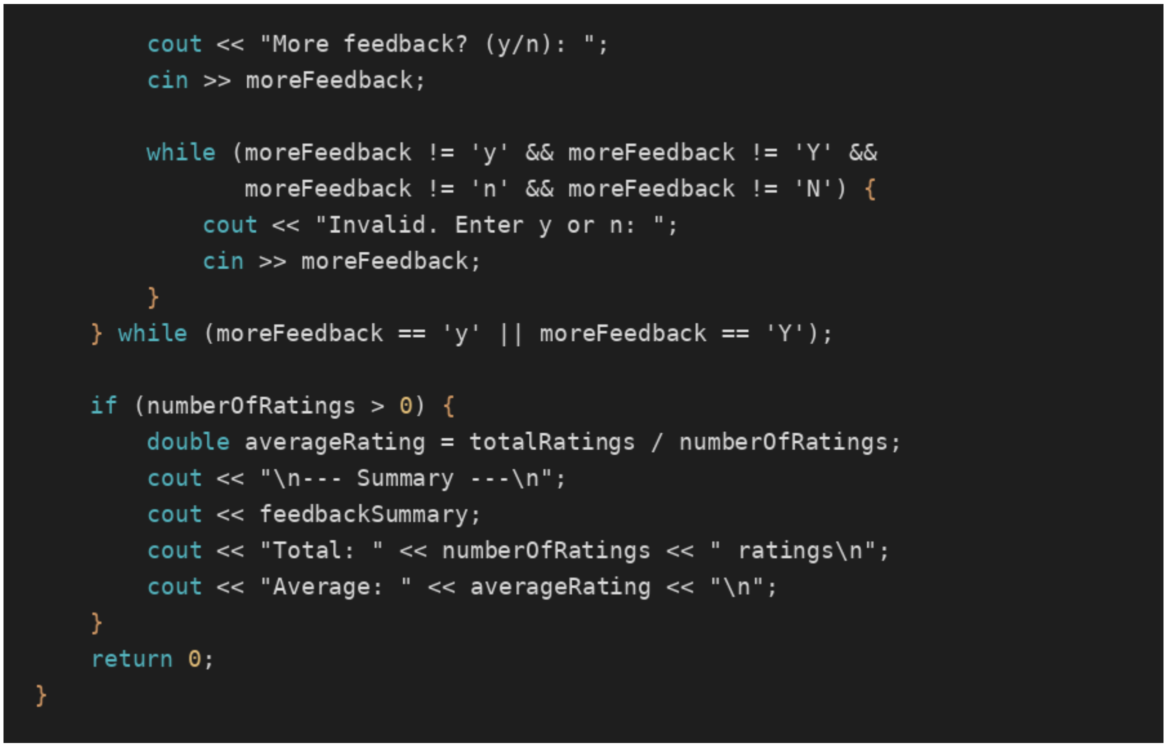
1. Prompts the user to enter a rating between 1 and 5.
2. Validates the rating using a while loop — re-prompts if out of range.
3. Clears the input buffer with getline() before reading the comment.
4. Reads the full comment text using getline().
5. Increments the counter and adds the entry to feedbackSummary.
6. Asks if the user wants to submit more feedback (y/n) with validation.



```
do {
    double rating;
    string comment, empty;

    cout << "Enter your rating (1-5): ";
    cin >> rating;

    while (rating < 1 || rating > 5) {
        cout << "Invalid. Enter 1 to 5: ";
        cin >> rating;
    }
    getline(cin, empty);   // clear buffer

    cout << "Enter feedback : ";
    getline(cin, comment);

    totalRatings += rating;
    numberOfRatings++;
    feedbackSummary += to_string(numberOfRatings)
                     + ". Rating: " + to_string(rating)
                     + " | Comment: " + comment + "\n";
```

Output & Summary Logic

After the loop ends, the program checks if at least one feedback entry was collected. If so, it calculates the average rating and prints the complete formatted summary.



```cpp
        cout << "More feedback? (y/n): ";
        cin >> moreFeedback;

        while (moreFeedback != 'y' && moreFeedback != 'Y' &&
               moreFeedback != 'n' && moreFeedback != 'N') {
            cout << "Invalid. Enter y or n: ";
            cin >> moreFeedback;
        }
    } while (moreFeedback == 'y' || moreFeedback == 'Y');

    if (numberOfRatings > 0) {
        double averageRating = totalRatings / numberOfRatings;
        cout << "\n--- Summary ---\n";
        cout << feedbackSummary;
        cout << "Total: " << numberOfRatings << " ratings\n";
        cout << "Average: " << averageRating << "\n";
    }
    return 0;
}
```

Sample Output

```
Enter your rating (1-5): 4
Enter feedback : Great service overall.
More feedback? (y/n): y
Enter your rating (1-5): 5
Enter feedback : Very helpful and fast.
More feedback? (y/n): n

--- Summary ---
1. Rating: 4.000000 | Comment: Great service overall.
2. Rating: 5.000000 | Comment: Very helpful and fast.
Total: 2 ratings
Average: 4.5
```

Advantages of the Project

- Simple and easy to understand for beginners
- Accepts multiple feedback entries in one session
- Validates both rating input and yes/no responses
- Provides a clear formatted summary with average
- Demonstrates core C++ concepts in a practical context

Future Enhancements

- Save feedback to a file for permanent storage
- Add user name or ID to each feedback entry
- Filter or search feedback by rating
- Convert to object-oriented design using classes
- Add GUI using graphics libraries

Conclusion

This Online Feedback System project demonstrates a practical and beginner-friendly approach to collecting and analysing user feedback in C++. It effectively uses loops, input validation, string handling, and arithmetic operations. The project serves as a strong foundation for more advanced applications such as database-backed feedback portals or web-based review systems.

Acknowledgement

We would like to thank our faculty and institution for providing guidance and support in completing this mini project.