**REDUCED LEVEL CALCULATOR**

**Overview**

This program is designed to calculate the reduced levels of different points in a surveying field. It offers two methods for calculating reduced levels: the Rise and Fall method and the Height of Collimation method. The user can choose the preferred method, and the program will guide them through the necessary steps to input the readings and calculate the reduced levels.

**Features**

**1. Rise and Fall Method:**

- Enter the benchmark RL.

- Input readings (backsights, foresights, or inter-sights) along with their values.

- The program computes the RLs for each point and performs an arithmetic check.

- Outputs a table of point numbers and corresponding RLs.

**2. Height of Collimation (HOC) Method:**

- Enter the benchmark RL.

- Input readings (backsights or foresights) along with their values.

- The program calculates the HOC and RL for each point.

- Outputs a table of point numbers, sight types, sight values, and RLs.

**Functions in the program**

1. **calculateHOC(double benchmarkRl, double backsight):** This function calculates the Height of Collimation (HOC). It takes the benchmark reduced level and the backsight reading as parameters and returns the sum of these two values as the HOC.

2. **calculateRL(double hoc, double sight, bool isBacksight):** This function calculates the reduced level based on the HOC, sight reading, and whether it's a backsight or a foresight. It returns the difference between the HOC and sight for intersight or foresight.

3. **groupName():** This function displays the name of the group (GROUP 9) as the heading of the program.

4. **calculateRlRiseAndFall():** This function calculates the reduced levels using the Rise and Fall method. It prompts the user for the necessary inputs and performs the calculations accordingly. It also displays the calculated reduced levels in a table format.

5. **calculateRlHOC():** This function calculates the reduced levels using the Height of Collimation method. It prompts the user for the necessary inputs and performs the calculations accordingly. It also displays the calculated reduced levels in a table format.

6. **errorMessage**(): This function displays an error message when the user enters an invalid option for calculating reduced levels.

7. **main():** This is the entry point of the program. It prompts the user to choose a method for calculating reduced levels and calls the appropriate function based on the user's choice. It returns 0 indicating successful execution of the program.

**Return Values**

The functions **calculateHOC** and **calculateRL** return double values representing the Height of Collimation and the reduced level, respectively.

The functions **calculateRlRiseAndFall**, **calculateRlHOC**, **groupName**, and **errorMessage** do not return any value. The **main** function returns an integer value (0) indicating the successful execution of the program.

**Usage**

- Compile the program using a C++ compiler (e.g., g++).

- Run the compiled executable.

- Choose the method (1 for Rise and Fall, 2 for HOC).

- Follow the prompts to input benchmark RL, number of readings, and individual readings.

- Enter individual readings with the reading type and value being separated by space

- View the computed RLs in tabular format.

**Example Input**

CALCULATING REDUCED LEVELS

Choose a method for calculating reduced levels:

1. Rise and Fall Method

2. Height of Collimation Method

Enter your choice (1 or 2): 1

Enter the reduced level of the benchmark: 100.0

Enter the number of readings: 4

Enter reading type (B=backsight, F=foresight, I=inter-sight) and value for point 1: B 2.5

Enter reading type (B=backsight, F=foresight, I=inter-sight) and value for point 2: F 3.0

Enter reading type (B=backsight, F=foresight, I=inter-sight) and value for point 3: B 1.8

Enter reading type (B=backsight, F=foresight, I=inter-sight) and value for point 4: F 2.2

Arithmetic Check: Sum of Rises - Sum of Falls = 1.9m

Should equal Last RL - First RL: 1.9m

Point Reduced Level(m)

1 100.0

2 102.5

3 99.7

4 101.9

**Authors**

1. Okumu Reagan Otim
2. Tugume Innocent
3. Nyakoojo Brian
4. Jabbo Martin
5. Mugisha Collins
6. Atukunda Earnest
7. Najjuma Christine Katende
8. Namakula Esther
9. Kusiima linnet
10. Anyanzo Emmanuel Izama

Attached are the flow charts for main, calculateRlRiseAndFall, calculateRlHOC functions