PROJECT NAME: Interactive Weather Visualizer

Objective

The main objective of this project is to create a simple interactive program using C++ and OpenGL that visually shows different types of weather such as sunny, rainy, snowy, and cloudy. Through this project, we aim to understand how basic graphics shapes are drawn on the screen and how user input from the keyboard can be used to change the displayed graphics. This project helps us learn the practical use of computer graphics concepts in an easy and visual way.

Theory

Computer Graphics is a branch of computer science that focuses on creating, displaying, and manipulating visual content using computers. It plays a vital role in modern applications such as simulations, games, user interfaces, and scientific visualization.

OpenGL (Open Graphics Library) is a cross-platform graphics API used to render 2D and 3D graphics. It provides a set of functions to draw geometric shapes, apply colors, and handle user interactions. In this experiment, OpenGL with Free GLUT is used to simulate different weather conditions in a 2D environment. Keyboard interaction allows users to switch between various weather visualizations such as sunny, rainy, snowy, and cloudy scenes.

Requirements

- Operating System: Windows
- Programming Language: C++
- Code::Blocks
- Graphics Library: OpenGL (FreeGLUT)

Procedure / Description of Code

1. Code::Blocks IDE is opened and a C++ project is created.
2. The OpenGL and FreeGLUT libraries are linked properly with the project.
3. An OpenGL window is initialized using GLUT functions.
4. An orthographic 2D projection is set using gluOrtho2D().
5. Separate functions are written to draw different weather conditions:

    - **Sunny Weather:** A circular sun is drawn using polygon primitives.
    - **Rainy Weather:** Slanted lines are used to represent rainfall.
    - **Stormy Weather:** Combination of clouds, lightning lines, and rain.
    - **Winter Weather:** Snowflakes are drawn using points.
    - **Foggy Weather:** Cloud-like shapes with grayish color.

6. Keyboard input is handled to allow the user to select different weather conditions.
7. The display function refreshes the screen whenever a key is pressed.

Program Input and Output with Screenshots:

Input Description:

By pressing different keys, the displayed weather changes instantly.

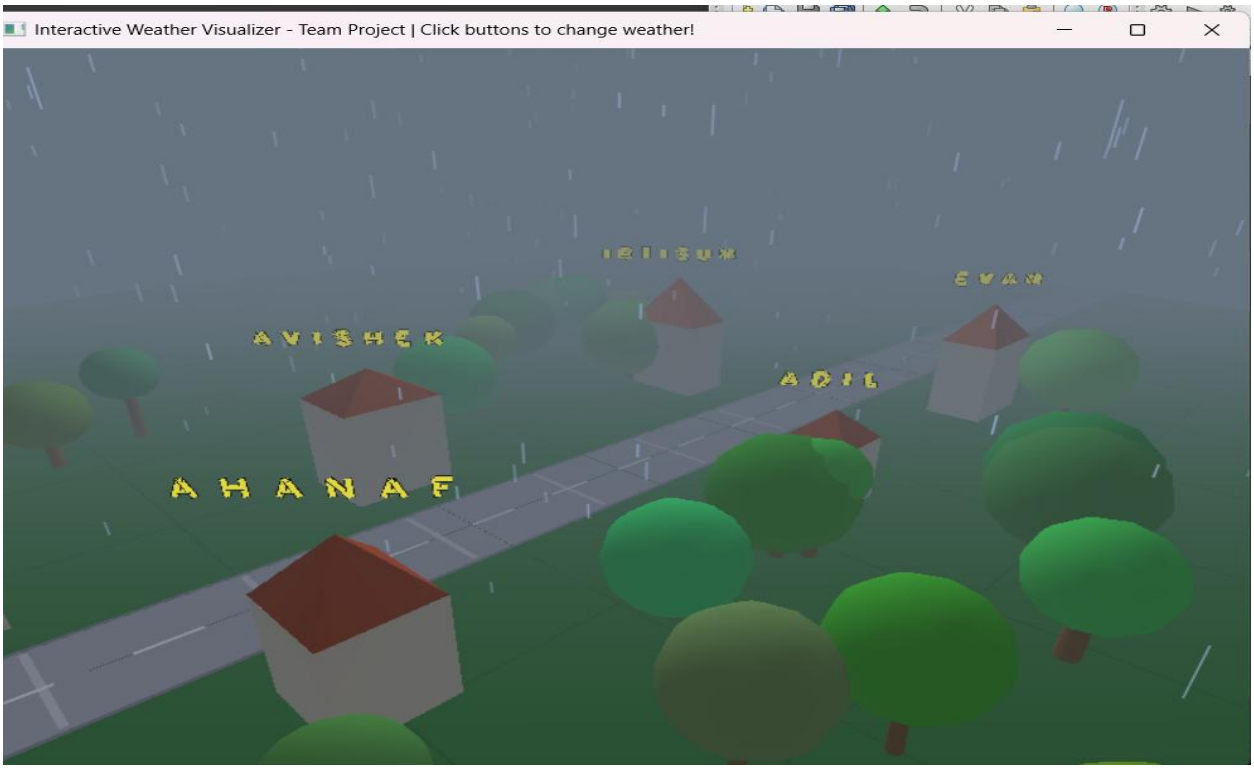
The program takes keyboard input from the user to display different weather conditions:

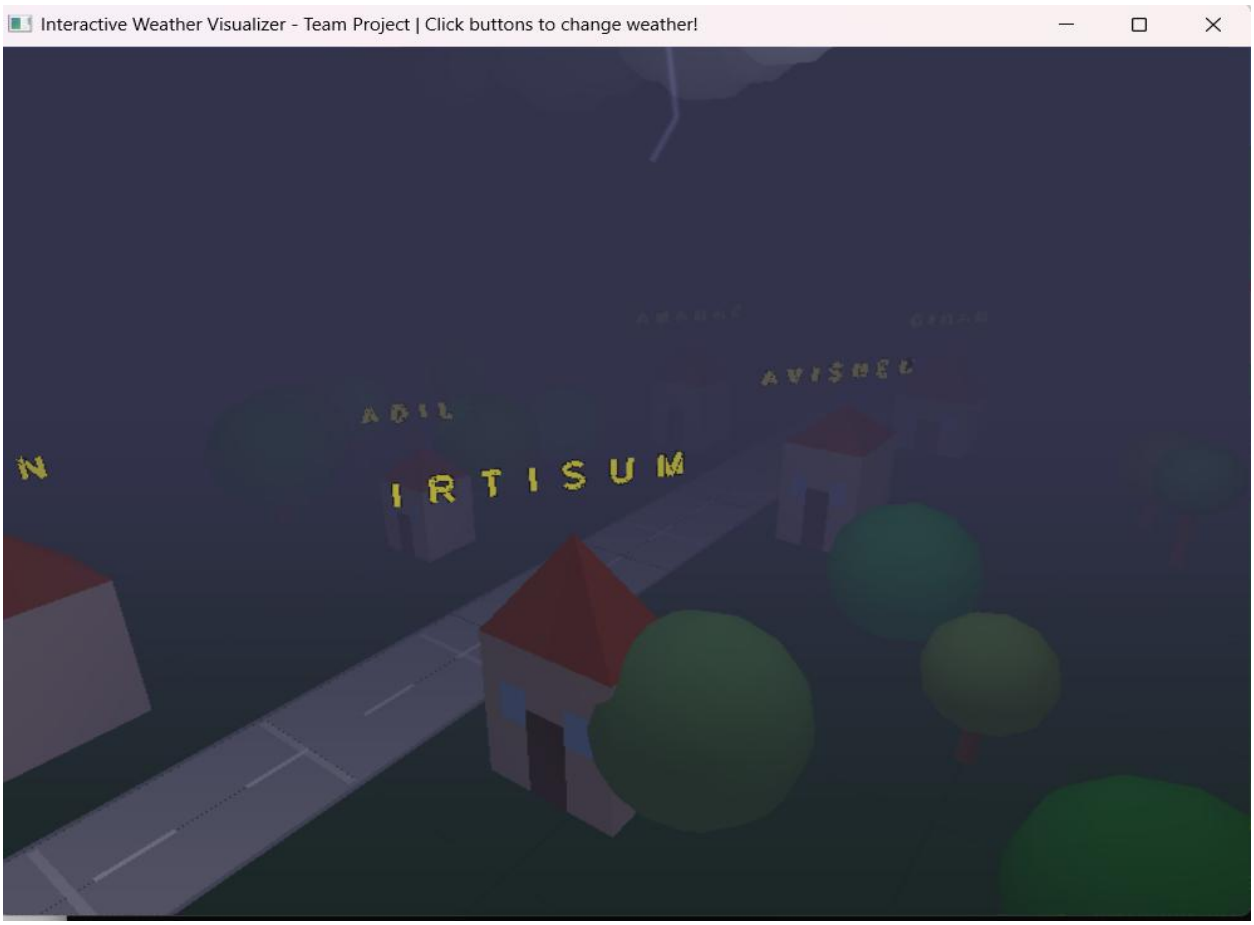
- Press 1 → Sunny Weather
- Press 2 → Rainy Weather
- Press 3 → Stormy Weather
- Press 4 → Winter Weather
- Press 5 → Foggy Weather

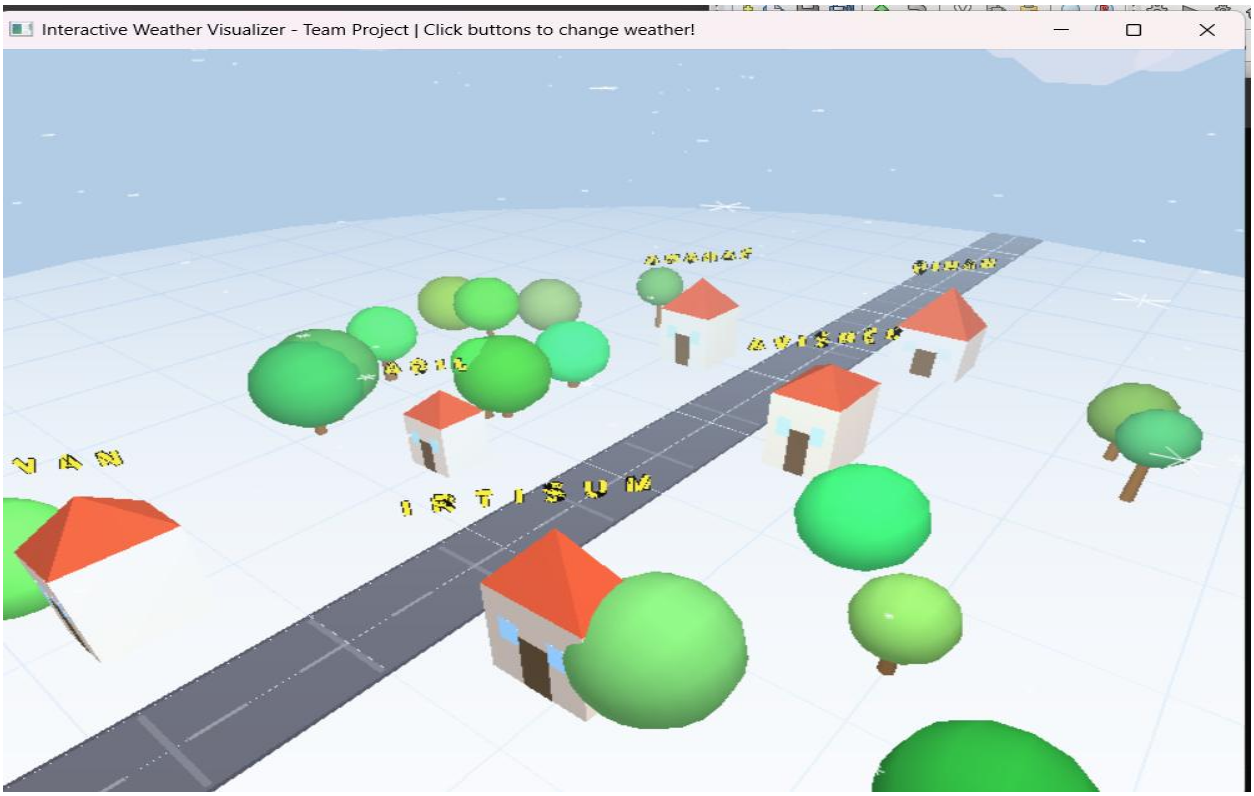
Output Description:

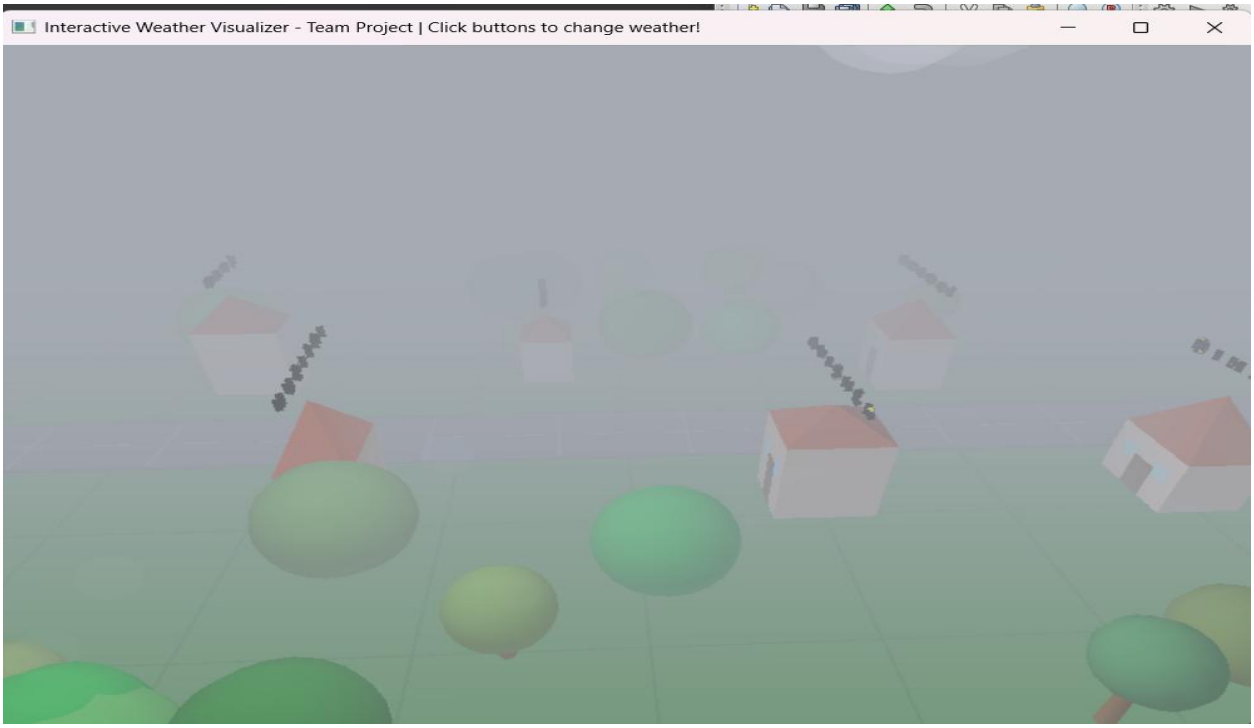
The screenshots below show the output of the project for different weather conditions. Each output is displayed when a specific key is pressed, and the graphics change according to the selected weather. These outputs demonstrate the correct working of the program and verify that user interaction and graphical rendering have been implemented successfully.


Interactive Weather Visualizer - Team Project | Click buttons to change weather!
AHANAF (Team Leader)
ADIL
AVISHEK
IRTISUM
Sunny
Rainy
Storm
Winter
Fog
CONTROLS:
Mouse: Click weather buttons to change weather
Arrow Keys: Rotate camera view
U: Toggle UI | T: Toggle Team Names
N: Day/Night Toggle | R: Reset Camera
1-5: Quick Weather Select | ESC: Exit
STATUS:
Current Weather: Sunny
Time of Day: DAY
Team Names Display: ON
Project Leader: AHANAF
Computer Graphics Lab Final Project


Interactive Weather Visualizer - Team Project | Click buttons to change weather!
AVISHEK
ADIL
AHANAF


Interactive Weather Visualizer - Team Project | Click buttons to change weather!
IRTISUM


Interactive Weather Visualizer - Team Project | Click buttons to change weather!
IRTISUM



Conclusion / Discussion

This project was successfully completed by the team members Ahanaf, Adil, Ovishek, and Irtisum. We worked together to design and develop an Interactive Weather Visualizer using C++ and OpenGL. The program shows different weather conditions such as sunny, rain, storm, winter, and fog based on keyboard input.

Through this project, we learned how to draw basic graphics, use OpenGL functions, and handle user input in a graphical program. Working as a team helped us share ideas, solve problems, and complete the project effectively. Overall, the project helped us understand computer graphics concepts in a practical and easy way.