



American International University-Bangladesh (AIUB)

**Department of Computer Science  
Faculty of Science & Technology (FST)**

Course Title  
**COMPUTER GRAPHICS**

**PROJECT TITLE**

**Obstacle odyssey**

**Supervised By**

MAHFUJUR RAHMAN

Submission Date: 1/19/2025

Submitted by:

| Semester: Fall 24-25 |                         | Section: H | Group Number: 2        |
|----------------------|-------------------------|------------|------------------------|
| SN                   | Student Name            | Student ID | Contribution (CO3+CO4) |
| 01                   | MD. FAHIM RAHMAN        | 21-45303-2 | 20%                    |
| 04                   | MD. NAZMUL ISLAM RAHAT  | 22-46323-1 | 20%                    |
| 22                   | MD. NAIMUL ISLAM        | 22-47899-2 | 20%                    |
| 18                   | TASURRON NAZERIN NAFISA | 22-47386-2 | 20%                    |
| 32                   | AL ISRAFIL NILOY        | 22-49033-3 | 20%                    |

## **Introduction:**

**Obstacle Odyssey** is a thrilling 2D platformer game created using OpenGL in C++, where every move matters, and a single mistake means the end of your journey. In this game, players embark on an adventurous journey through multiple levels, each designed to be harder than the previous one. Players must navigate through a series of increasingly challenging levels filled with deadly traps, moving platforms, and unpredictable hazards. With vibrant visuals and increasing difficulty, this game promises an exciting and engaging experience, pushing players to their limits as they strive to conquer all levels.

## **What We Implemented:**

We developed **Obstacle Odyssey**, a thrilling 2D platformer game using OpenGL in C++. The game features multiple progressively challenging levels with vibrant visuals, dynamic obstacles, and smooth gameplay mechanics. Key implementations include:

- Smooth player movement with refined gravity and collision detection.
- Static and moving platforms, disappearing platforms, and environmental hazards like lava and spikes.
- Auto-reset mechanics when the player falls or touches hazards.
- A life system with limited attempts and a game-over screen upon failure.
- Diverse level designs with themes ranging from grassy fields to volcanic landscapes and sky-high platforms.

## **Why We Implemented This:**

The primary goal of this project was to apply computer graphics concepts in a practical and engaging way by creating an interactive 2D platformer game. This game serves as a medium to:

- Demonstrate the application of OpenGL for rendering dynamic game environments.
- Develop complex mechanics such as object movement, collision detection, and game logic.
- Enhance programming skills in C++ while integrating advanced gameplay mechanics.
- Create a fun and challenging experience that appeals to players and showcases technical creativity.

## **Significance of This Project (Motivation):**

The motivation behind **Obstacle Odyssey** stems from the desire to blend creativity with technical skills, leading to the development of a complete, engaging game. The project is significant because:

- It integrates complex algorithms and real-time graphics rendering for interactive gameplay.
- It challenges problem-solving skills through the implementation of physics-based mechanics.
- It provides a foundation for future developments in game design and software engineering.
- It encourages creativity in designing visually appealing and mechanically challenging game levels.

## **Target Population:**

The target audience for **Obstacle Odyssey** includes:

- Casual and hardcore gamers who enjoy platformer and adventure games.
- Students and developers interested in learning OpenGL and game development concepts.
- Individuals seeking engaging, skill-based gameplay that progressively increases in difficulty.
- Fans of classic platformers who appreciate challenging, reflex-driven games.

## **Knowledge Gained from Tools Used in This Project:**

Developing **Obstacle Odyssey** involved using several tools and technologies that contributed to gaining valuable technical skills and practical knowledge in computer graphics and game development:

### **1. OpenGL (Open Graphics Library)**

- **Graphics Rendering:** Learned how to render 2D shapes, textures, and objects using OpenGL primitives.
- **Transformations:** Gained experience in applying transformations (translation, scaling, rotation) for dynamic object movement and animations.
- **Rendering Pipeline:** Understood how OpenGL's rendering pipeline processes data to display visuals efficiently.
- **Color Manipulation:** Applied color models to design visually distinct game elements using glColor3f.
- **Real-Time Graphics:** Implemented real-time object rendering and updates for smooth gameplay.

### **2. C++ Programming Language**

- **Object-Oriented Programming (OOP):** Designed modular and reusable code using classes and functions for player movement, game logic, and collision detection.
- **Memory Management:** Gained experience managing resources efficiently in a graphics-intensive environment.
- **Algorithm Development:** Implemented collision detection algorithms, gravity simulation, and reset logic.
- **Performance Optimization:** Learned to write optimized code for smooth performance in real-time applications.

### **3. GLUT/FreeGLUT (OpenGL Utility Toolkit)**

- **Window and Input Handling:** Managed window creation, rendering contexts, and user input (keyboard controls for movement and jumping).
- **Event-Driven Programming:** Implemented real-time updates and event-driven behavior for interactive gameplay.

- **Cross-Platform Support:** Gained knowledge about building and running the game across different operating systems.

#### **4. Integrated Development Environment (IDE) –Code::Blocks**

- **Code Management:** Used IDE tools for code organization, debugging, and performance analysis.
- **Debugging Skills:** Improved problem-solving by identifying and fixing errors in complex systems.
- **Compilation and Linking:** Gained experience with building C++ projects and managing dependencies.

#### **5. OpenGL Extensions and Graphics Drivers**

- **Hardware Interaction:** Understood how graphics drivers and extensions optimize rendering based on hardware capabilities.
- **Performance Tuning:** Learned how to ensure compatibility and performance across different devices.

### **Knowledge Applied for Job Market Purpose**

The development of *Obstacle Odyssey* applies practical skills and concepts highly relevant to the game development and software engineering industries, including:

- **Game Development Skills:** Proficiency in C++ and OpenGL, which are foundational in game engines and graphics programming.
- **Graphics Programming:** Experience with rendering, object transformations, and animations using industry-standard graphics APIs.
- **Problem-Solving and Algorithms:** Implementing collision detection, physics simulation, and object interactions demonstrates critical thinking and algorithm development.
- **Software Development Practices:** Project structuring, debugging, and optimizing game performance align with industry coding standards.
- **Real-Time Systems:** Handling real-time updates and smooth gameplay mechanics reflects skills in time-critical application development.
- **User Experience (UX) Design:** Designing engaging levels and intuitive controls improves understanding of player-centric game design.

### **Relevant Job Roles:**

- Game Developer (C++/OpenGL)
- Graphics Programmer
- Software Engineer in Interactive Media
- Simulation and Visualization Developer
- Game Designer

### **Knowledge Applied for Higher Study Purpose**

For advanced studies in computer science, graphics, and game design, this project demonstrates the application of complex concepts that form the basis for further research and specialization:

- **Advanced Computer Graphics:** Understanding of rendering pipelines, graphics APIs, and interactive design, providing a strong foundation for research in 3D graphics, shaders, and real-time rendering.
- **Game Physics and Simulation:** Practical experience with physics simulation (gravity, collisions) is beneficial for advanced study in simulation, animation, and virtual environments.
- **Software Engineering Concepts:** Project management, modular coding, and optimization prepare for higher-level research in scalable software systems.
- **Human-Computer Interaction (HCI):** Applying user experience design principles aligns with studies focused on improving interaction in gaming and virtual environments.
- **Artificial Intelligence (Potential for Expansion):** Foundation for integrating AI-based enemy behaviors or procedural level generation in future research projects.

### **Relevant Higher Study Fields:**

- Master's/PhD in Computer Graphics or Game Development
- Advanced Game Design and Interactive Media
- Virtual Reality (VR) and Augmented Reality (AR) Development
- Artificial Intelligence in Games
- Simulation and Visualization Studies

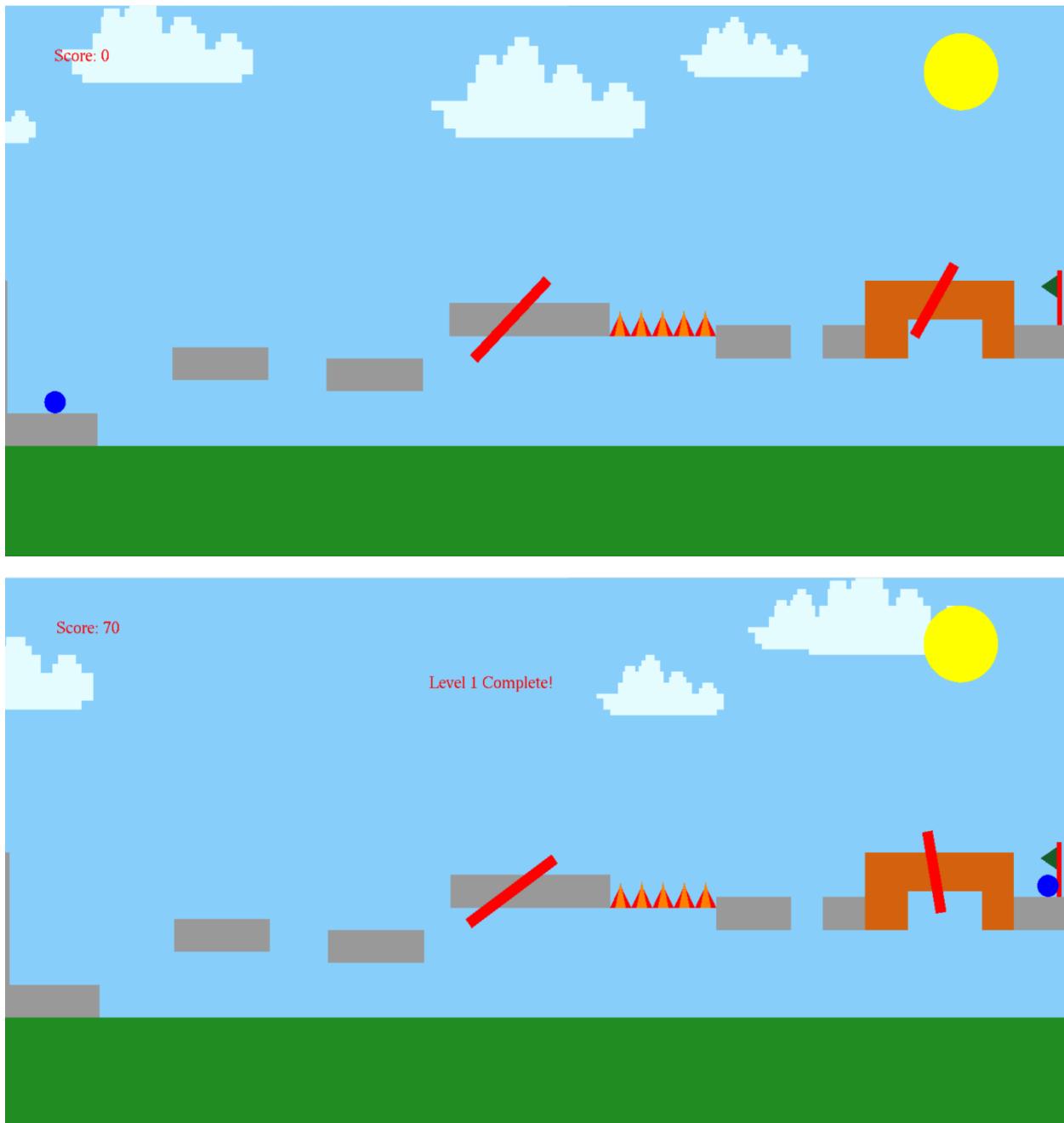
## **References:**

**Github link:** <https://github.com/Tasurron/Computer-Graphics-Project>

**Youtube link:** [https://youtu.be/wz3\\_HlyHEVc?feature=shared](https://youtu.be/wz3_HlyHEVc?feature=shared)

## **Screenshot of the project:**

### **Level 1: The Beginner's Path**



## Level 2: The Fiery Descent



### Level 3: The Skyfall Gauntlet

