Name Amr Magdy

ID 19016116

Graphics | Solar System Project

Introduction

The Solar System project is a visualization of the celestial bodies in our solar system using OpenGL. It includes the Sun, planets, moon, and Saturn's ring. The project allows for navigation and observation of the solar bodies.

Required Libraries

- Freeglut -> not included in source code.
- Glew -> not included in source code.
- FreeImage -> included in source code.

Source Code

https://drive.google.com/file/d/1yhF458bL0D1PtV5klNsnDYE74OYEG0kY/view?usp=sharing

Executable

https://drive.google.com/file/d/1zjvLlTO3eBkAiyAVF7XIqe5pxdYIYyKn/view?usp=sharing

YouTube Video

https://youtu.be/EmEFZJJ1SzM

References

- Solar Textures | Solar System Scope
- LearnOpenGL - Textures
- c++ - How do I load textures to OpenGL using FreeImage library? - Stack Overflow

Project Modules & Classes

- **Vec3**
    - Represents a 3D vector with x, y, and z components.
    - Provides various vector operations such as addition, subtraction, scaling, normalization, cross product, magnitude, and dot product.
- **camera**
    - Represents a camera in the 3D space.
    - Contains the camera position, view direction, up vector, and right vector.
    - Provides functions to move the camera in different directions and rotate it around the x, y, and z axes.
- **CosmicSphere**
    - base class representing a cosmic sphere.
    - Contains properties such as radius, distance from the rotation target, rotation speed around the target, self-rotation speed, and texture.
    - Provides a **draw()** function to draw the cosmic sphere.
- **Star**
    - Derived class from **CosmicSphere** representing a star.
    - Contains additional properties such as the light source index and color.
    - Overrides the draw() function to draw the star.
- **Ring**
    - Derived class from **CosmicSphere** representing a ring around a planet.
    - Contains an additional property for the inner radius of the ring.
    - Overrides the draw() function to draw the ring.

Implementation Details & Controls

a. Texture Loading

- The **LoadTextureFromImg()** function loads a texture from an image file using the **FreeImage** library.
- The texture is bound and applied to the cosmic spheres during drawing.

b. Drawing Functions

- The **draw()** function of each cosmic sphere class is responsible for drawing the respective object.
- OpenGL functions are used to set material properties, bind textures, and draw geometries such as spheres, rings, and orbits.

c. Animation and Navigation

- The project supports animation and navigation controls.
- **Animation** can be paused or resumed using the "P" key.
- **Navigation** is achieved by moving the camera using the WASD keys and rotating it using the arrow keys.
- **Acceleration** by repeatedly pressing the left-shift key.

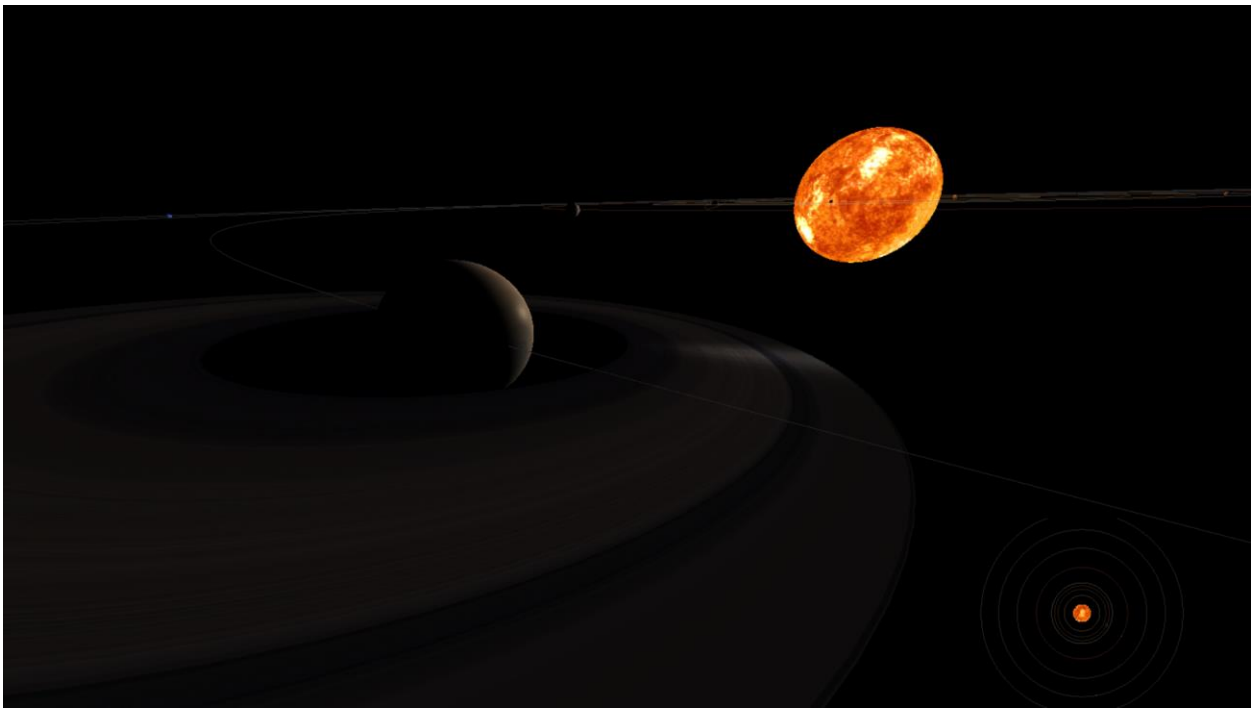
d. Solar System Setup

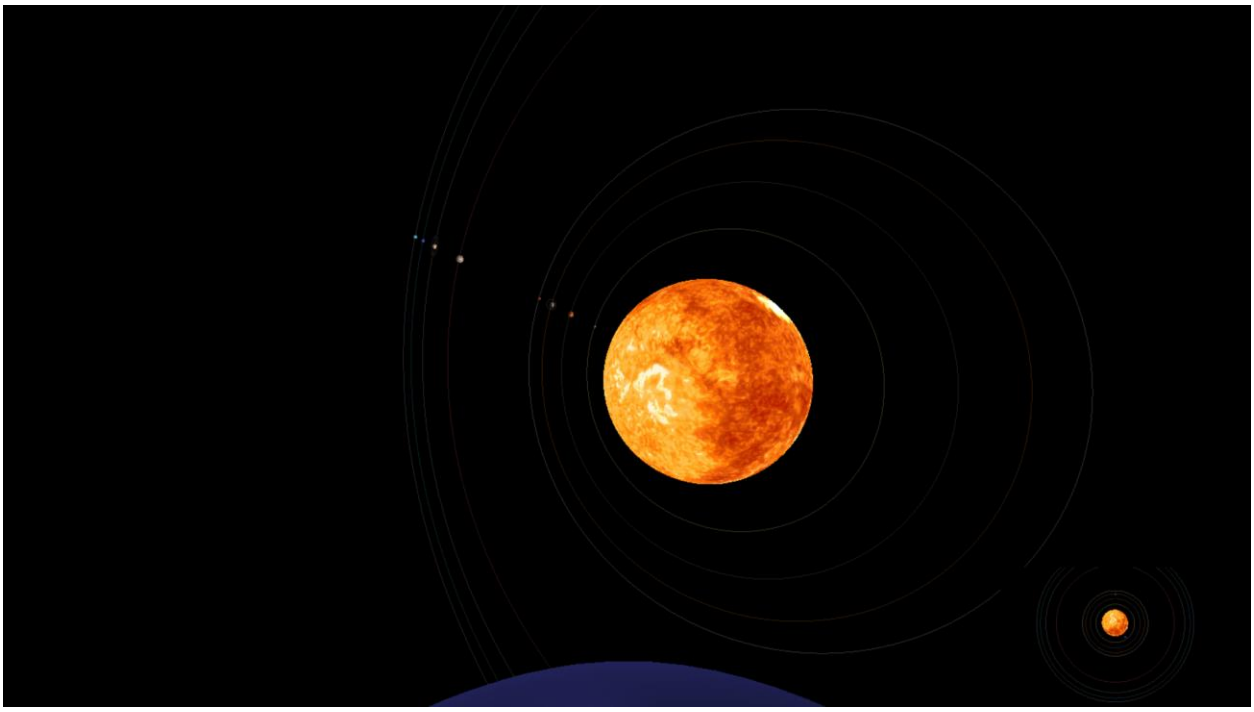
- The solar system is initialized with the Sun and planets using the **CosmicSphere** class.
- Each planet is positioned at a specific distance from the Sun and rotates around it at a certain speed.
- The Moon is a satellite of the Earth and rotates around it.
- The Saturn's ring is represented by the Ring class and positioned around Saturn.
- Every solar object rotates around itself with some velocity.

e. Other Controls

- Toggling of orbits visibility using the "O" key.
- Fullscreen mode can be toggled using the "F" or F11 keys.

In-game Screenshots





Please note that the relative distances between the solar objects do not match the actual ones within our solar system.