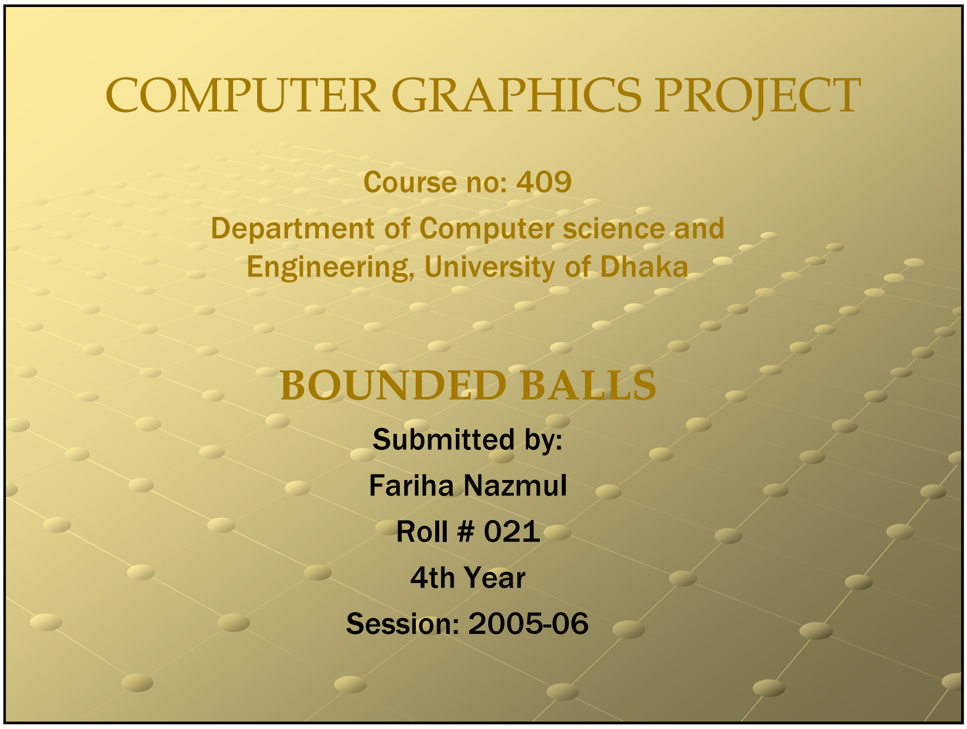


COMPUTER GRAPHICS PROJECT

Course no: 409

Department of Computer science and
Engineering, University of Dhaka

BOUNDED BALLS

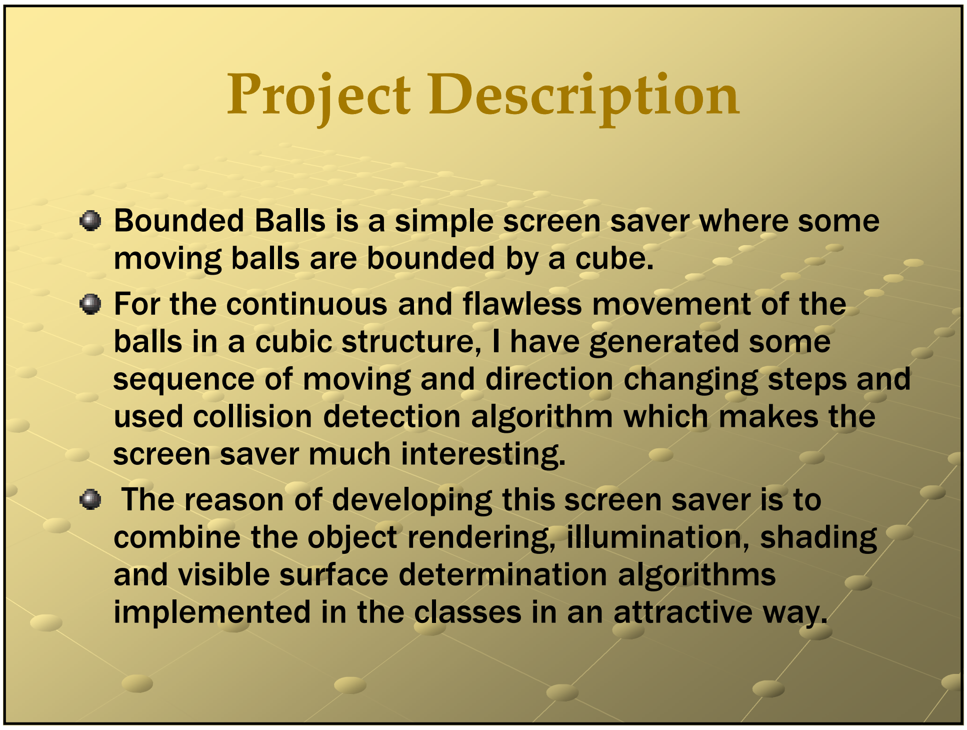
Submitted by:

Fariha Nazmul

Roll # 021

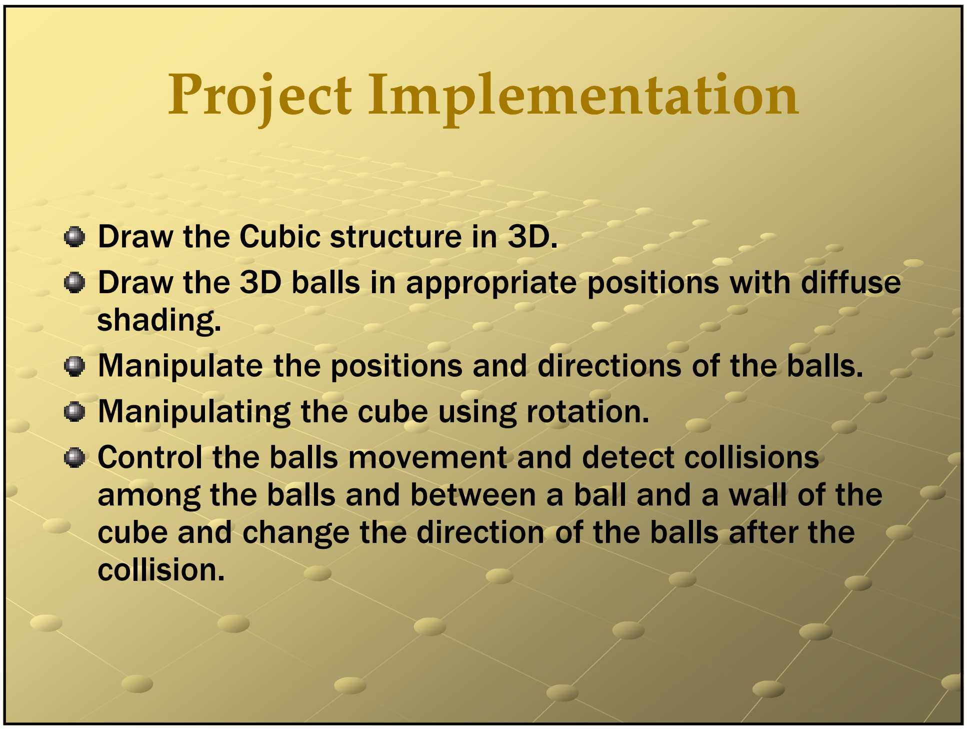
4th Year

Session: 2005-06



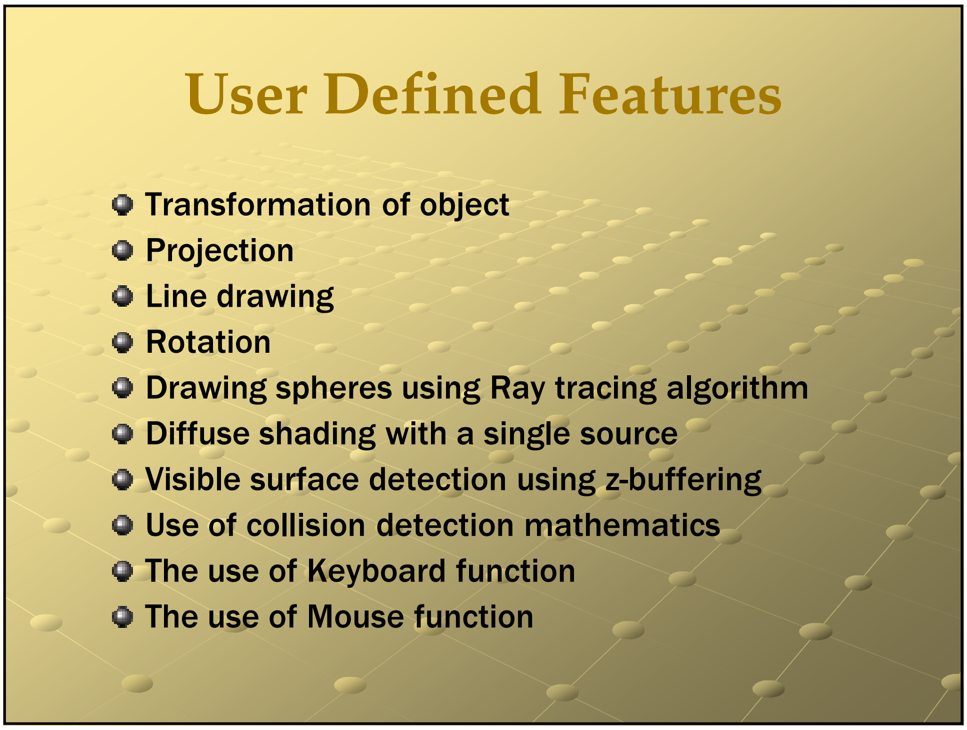
Project Description

- Bounded Balls is a simple screen saver where some moving balls are bounded by a cube.
- For the continuous and flawless movement of the balls in a cubic structure, I have generated some sequence of moving and direction changing steps and used collision detection algorithm which makes the screen saver much interesting.
- The reason of developing this screen saver is to combine the object rendering, illumination, shading and visible surface determination algorithms implemented in the classes in an attractive way.



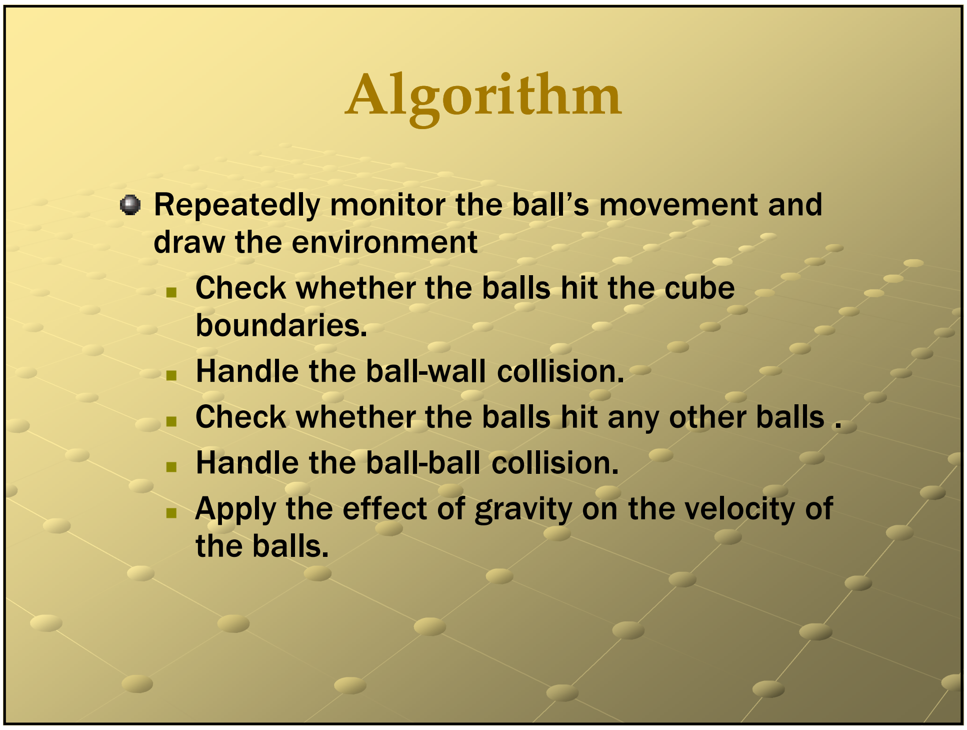
Project Implementation

- Draw the Cubic structure in 3D.
- Draw the 3D balls in appropriate positions with diffuse shading.
- Manipulate the positions and directions of the balls.
- Manipulating the cube using rotation.
- Control the balls movement and detect collisions among the balls and between a ball and a wall of the cube and change the direction of the balls after the collision.



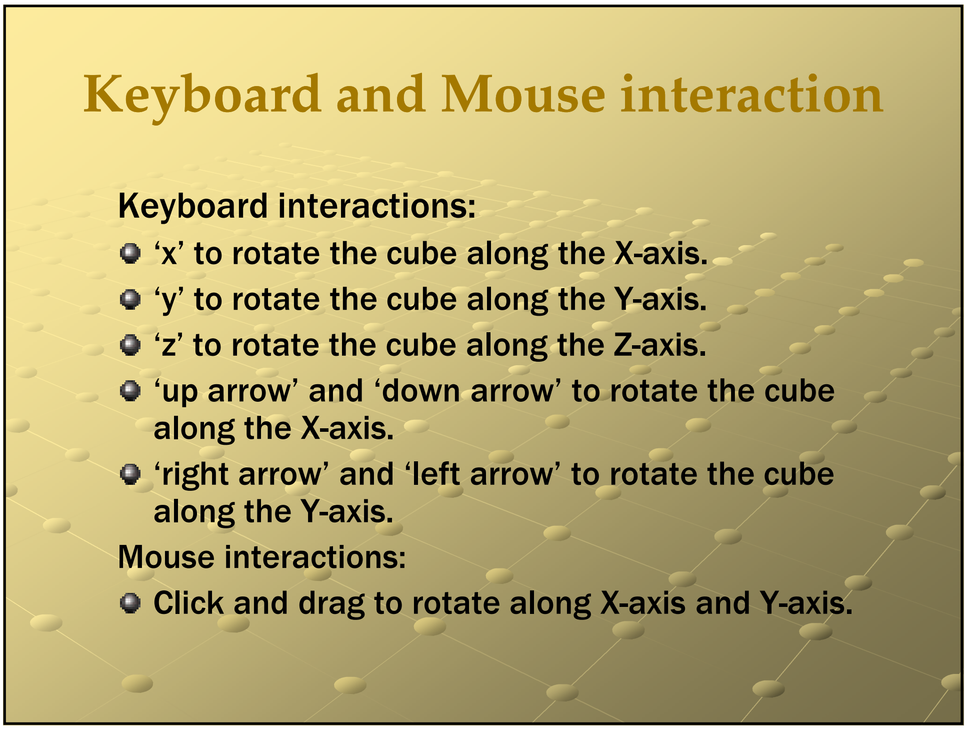
User Defined Features

- Transformation of object
- Projection
- Line drawing
- Rotation
- Drawing spheres using Ray tracing algorithm
- Diffuse shading with a single source
- Visible surface detection using z-buffering
- Use of collision detection mathematics
- The use of Keyboard function
- The use of Mouse function



Algorithm

- Repeatedly monitor the ball’s movement and draw the environment
  - Check whether the balls hit the cube boundaries.
  - Handle the ball-wall collision.
  - Check whether the balls hit any other balls .
  - Handle the ball-ball collision.
  - Apply the effect of gravity on the velocity of the balls.



Keyboard and Mouse interaction

Keyboard interactions:

- ‘x’ to rotate the cube along the X-axis.
- ‘y’ to rotate the cube along the Y-axis.
- ‘z’ to rotate the cube along the Z-axis.
- ‘up arrow’ and ‘down arrow’ to rotate the cube along the X-axis.
- ‘right arrow’ and ‘left arrow’ to rotate the cube along the Y-axis.

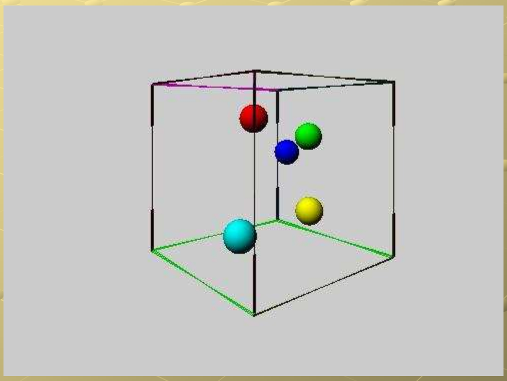
Mouse interactions:

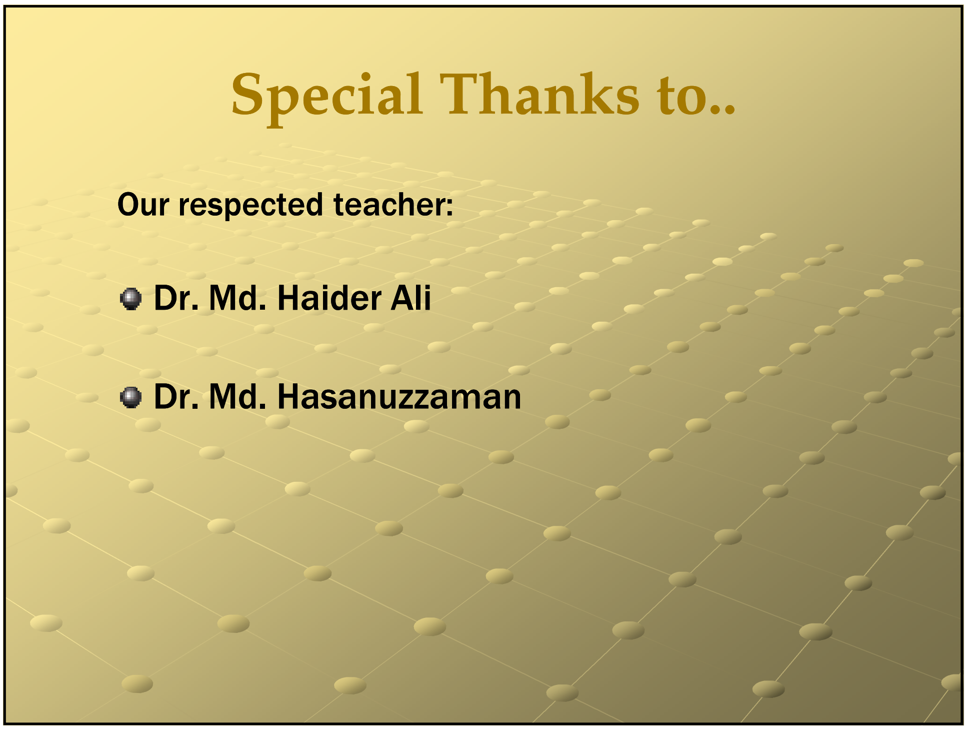
- Click and drag to rotate along X-axis and Y-axis.

Limitations

- Implementation of the collision detection algorithm is not perfect.
- Implementation of z-buffering algorithm is dependent on the screen size. So resizing option is not there.
- Graphical representation of the background is not attraction.
- Shadowing of the balls on the cube surface in not implemented.
- There in no option for zoom in or zoom out.

Screenshot

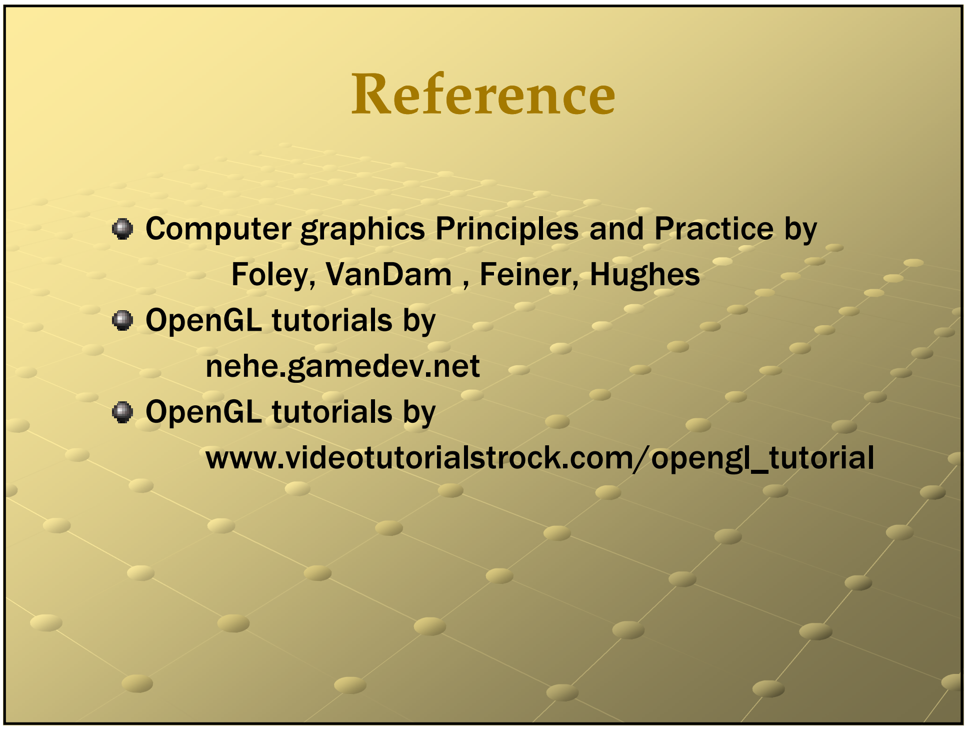




Special Thanks to..

Our respected teacher:

- Dr. Md. Haider Ali
- Dr. Md. Hasanuzzaman



Reference

- Computer graphics Principles and Practice by
  Foley, VanDam , Feiner, Hughes
- OpenGL tutorials by
  nehe.gamedev.net
- OpenGL tutorials by
  www.videotutorialstrock.com/opengl_tutorial