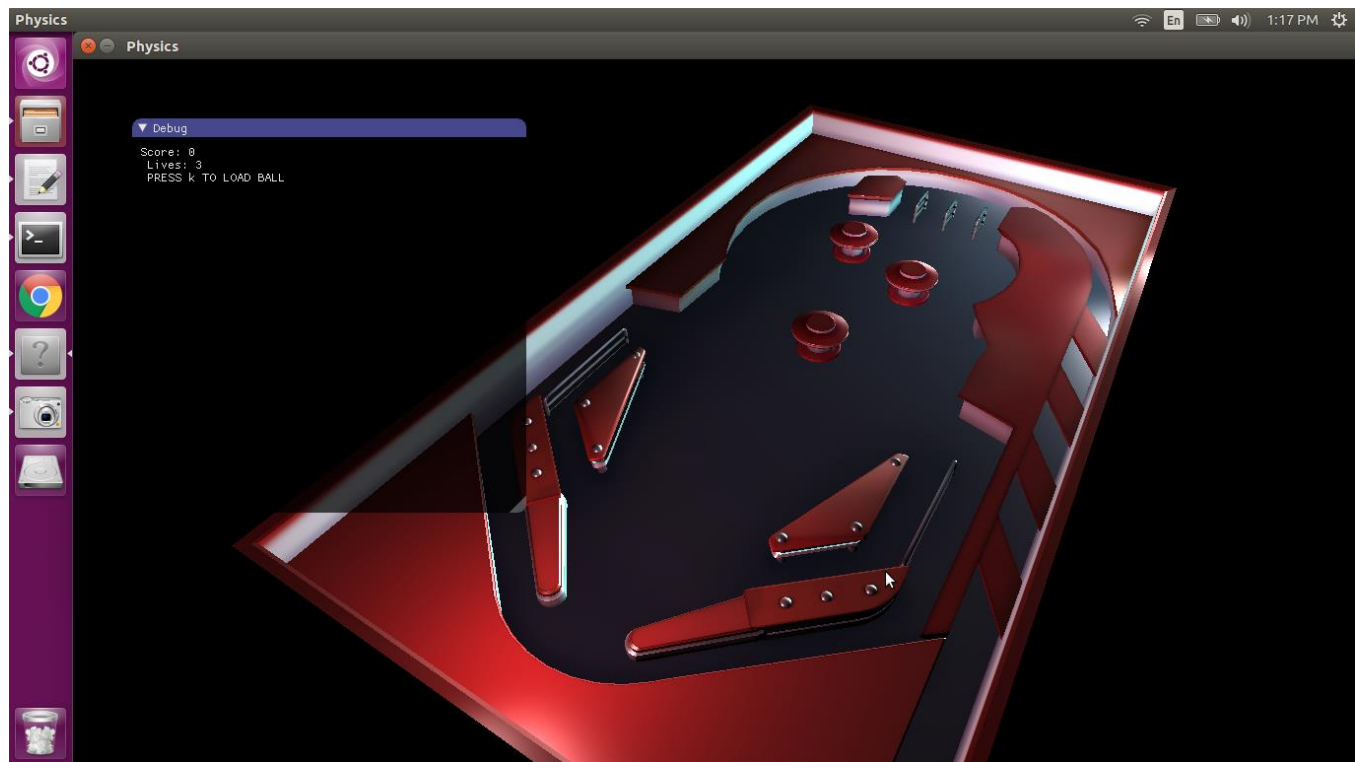


PA 10: Pinball

By Ronn Quijada and Sean Stevens



Overview

Features include custom models and a GUI. This is a simple pinball game with two paddles, a plunger, and a whole bunch of bumpers. Separate models were made for static collision meshes so that concave collisions would behave correctly and predictably.

Libraries needed

GLEW, GLM, SDL, Assimp 3.3.1, Magick++, Bullet 2.87

Build instructions

Create a build folder called build and do cmake .. inside of it.

Build from within the PA9/build directory via the make file generated by cmake. Simply do make to build and make clean to remove the build files.

Run the program by doing ./Tutorial after building the program.

Ensure that you don't remove the build folder entirely, as the assets folder exists in both the PA9 and build directory due to root directory differences between the IDE and Ubuntu.

Controls

- K to load a ball on the plunger
- B to launch the ball
- G to activate left paddle
- H to activate right paddle
- Y to switch to per vertex lighting
- T to switch to per fragment lighting
- Q W E to increase ambient RGB lighting
- A S D to decrease ambient RGB lighting
- Move mouse to move viewport (camera)

Extra Credit

- Scoring is done through a GUI in addition to terminal
- Multiple balls (you can load multiple balls onto the plunger)
- Bumpers that bounce and light when hit.
- Spotlight that follows the ball
- Top 10 Scores List

Tech Manual

- Issues
 - Too many lights will cause the game to crash. Currently, the limit is hard-coded to resolve this.
 - Per vertex lighting does not have proper ambient lighting due to an issue with getting it to work in the vertex shader
- What to do different
 - Adding more methods for adding points than hitting the bumpers.
 - Add sound effects for a better game feel
 - Find a way to get the top ten scores to display on GUI instead of terminal

Code Outline

Camera.h – Container class for viewport

Light.h – Container class for light data (position, color, etc)

Ball.h – Ball physics object

Bumper.h – Class for cylindrical and triangle bumpers

Engine.h – Container class for game engine

Event.h – Container class for managing keyboard & physics events

Graphics.h – Container class for graphics rendering

Graphics_Headers.h – Container class for OPENGGL, GLEW, and GLM

Gui.h – Container class for GUI functions

Imconfig.h – Used for IMGUI

Imgui.h – Used for IMGUI

Imgui_impl_sdl_gl3.h – Used for IMGUI

Imgui_internal.h – Used for IMGUI

Loader.h – Class for loading textures & models

Obj.h – Class for loading models

Object.h – Base class for graphics rendering

Paddle.h – Object that hits the paddles

physObject.h – Base class for objects using bullet api

plunger.h – Object that launches the ball

shader.h – Container class for shader compiling and binding

stb_rect_pack.h – Used for IMGUI

stb_textedit.h – Used for IMGUI

stb_truetype.h – Used for IMGUI

window.h – Container class for SDL window

world.h – Object that represents pinball game