

PA199 – Advanced Game Development

Project Assignment

Semester: Spring 2020

The aim of the assignment is twofold:

- develop a small game engine in C++ including basic graphics and physics components
- develop a simple game using your engine

The engine should cover the general motion of a rigid body in three-dimensional space, including the collision of objects and their subsequent movement.

1. Game Engine

You should implement from scratch a mini game engine, that should provide support to developers with:

- Modelling the 3D scene of the game.
- Providing necessary physics simulations usable in different computer games.

1.1. Specification

- Based on the vector theory, you should implement a 3D vector class. Vector will have components X, Y and Z
- Based on the matrices theory, you should implement a 3×3 matrix class.
- You should implement a ray class based on defined vector and matrix classes

1.2. Minimum functionality of engine:

- Vector class
 - Unit vector
 - Magnitude of a vector
 - Invert a vector
 - Add, subtract two vectors
 - Dot product
 - Cross product
- Matrix class
 - Addition, Subtraction, Multiplication
 - Transpose
 - Inverse
- Ray class
 - Calculate distance between
 - two rays
 - ray and a point

2. The Game

2.1. Motivation – Breakout game

Your goal will be to implement a “circular” variant of classic breakout game – the goal of the game is to break all bricks in the level using a ball(s), controlled by in-game physics and paddle(s), controlled by the player.

Basic rules:

- The ball starts at the position of a paddle, moves at a constant speed
- When the ball collides with a brick, the ball bounce and brick is destroyed.
- If the ball leaves the game area, the player loses one life, and the ball is respawned near the paddle

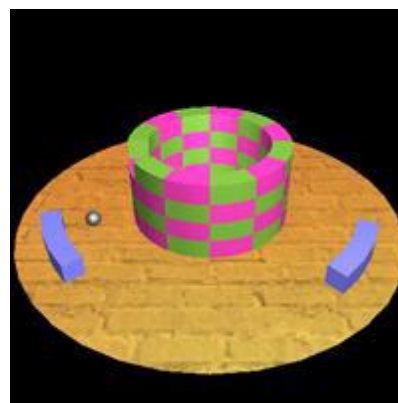
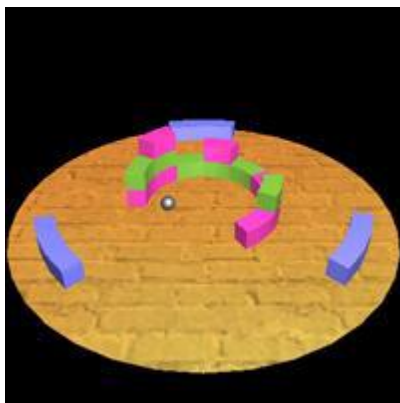
Typical extensions:

- There can be multiple balls in the game at the same time. Player lives as long as at least one ball is inside the area.
- The paddle can change its size.
- The speed of ball(s) is increasing over time.
- Various types of bricks with various properties (durability, power-up drops, etc.)

2.2. Game specification

You should develop a “circular” breakout in 3D.

- The bricks will be arranged in a cylindrical wall in the centre of the play area. Wall will have several stores, and each store will contain at least 12 bricks. When a brick in the bottom row is destroyed, bricks above it falls down.
- The player will be controlling three curved paddles, positioned at the outer edge of the play area.
- All paddles move simultaneously – the player can rotate with them in CW and CCW directions



2.3. Minimum functionality

- Game mechanics
 - The player can control the paddles (e.g. arrow keys) and launch the ball (e.g. spacebar)
 - Simple score counting and player life counting
 - Simple “Game over” and “You win” messages at appropriate times
 - Camera switching (e.g. keys “1”, “2” and “3”)
- Graphics
 - Construct the geometry of level from scratch, including:
 - ball,
 - paddles,
 - wall made of individual bricks,
 - ground.
 - Colours: ball, paddles, bricks
 - Texturing: ground
 - Multiple cameras
 - “perspective”, “top” and “ball” views
 - Lighting
 - ambient, diffuse and specular light
- Physics - real-time simulation of the ball movement, based on collision detection and reaction:
 - Ball – paddle (the curved shape of paddle matters)
 - Ball – brick (the curved shape of brick matters)
 - Ball – (invisible) the outer edge of the play area

2.4. Extra Features (optional)

Some inspiration:

- Game-play
 - Increasing the speed of the ball over time
 - Splitting the ball into two
 - Durability of bricks
 - Changing the size of paddles
 - Whatever you find in countless breakout games :)
- Graphics
 - Transparency - bricks of the wall will become semi-transparent when the ball is “behind” the wall
 - The durability of bricks – changing colour after each hit
- Physics – extra elements influencing the movement of the ball
 - Static, unbreakable colliders
 - Small areas that decrease or increase the speed of ball

3. Report

No report required this semester.

4. Deadlines

- regular term (řádný termín): September 13
- second, “resit” term (opravný termín): September 27 (only for the unexpected case that somebody will not submit acceptable solution before the regular deadline and still want a good grade.

Nevertheless, you can submit your projects at any time from now till September. Just let us know that you submitted, so we can check your work and grade you.

5. Submission

Please create a folder: “PA199 - [your surname name]” and put following into this folder:

- Folder “**bin**” containing the runnable version of the project
- Folder “**src**” containing all source codes.

Pack the whole “PA199...” into .zip archive and upload this archive into homework vault in IS.