

Documentation: Mara Soft: Episode I – Gröller's Cup

Brief Description

Gameplay

- **Playable (11 points)**
 - the player can move around freely and objects are already placed in the scene. A coarse structure of the final game is already visible.
 - The movements were implemented as a callback with the `Player.pollMoves()` function and is called in the renderloop.
- **3D Geometry (6 points)**
 - we have simple 3D geometries, like spheres, cubes and cylinders in our scene as well as complex 3D geometry like the wall and the asteroids.
 - The geometry is implemented in the `NewGeometry.cpp` and is based on the `Geometry` class. The `loadMesh` function was added to load complex meshes. The complex geometry objects (asteroid and wall) are downloaded from free3d.com and are loaded with `assimp`.
- **Win/ Loose Condition (3 points)**
 - by touching the cup (cylinder) the player (sphere) wins the game. Touching the asteroids leads to losing lives and losing the game.
 - The implementation makes use of the `PhysX` engine and its availability to filter collisions.
- **Intuitive Controls (2 points)**
 - the player can move around with W, A, S, D and jump by pressing the `SPACE` bar.
 - The movements were implemented as a callback with the `Player.pollMoves()` function and is called in the render loop.
- **Intuitive Camera (2 points)**
 - the camera moves with the player and creates a third person view
 - the camera has a fixed target and changes its position with the player's movement
- **Textures (2 points)**
 - The textures are custom to their object.
- **Documentation (1 point)**
- **Adjustable Parameters (1 point)**
 - 75% of the parameters can be adjusted in the settings file. The brightness is not implemented yet.
 - The implementation reads the parameters from the `settings.ini` file with a stream where the user can adjust them.
- **Moving Objects (2 points)**
 - We have catapults that act as enemies that move around, trying to impede the player from getting to his/ her goal
 - The player is also visible and is being maneuvered by the user.

Effects

All effects were implemented using learnopengl and with the help of the tutors on the forum and in Tutorien. Wikipedia (Cel Shading) and the tutorials provided by the CG-Team (Particle System, Cel Shading) have also been used.

- **Cel Shading (4 points)**
 - All Objects are cel shaded, except some particular walls on which environment mapping has been applied
- **Environment Mapping (8 points)**
 - There are some obstacles (walls) that have such an effect applied to them, in order to be a „tricky“ obstacle for the player
 - The Elements reflect the Skybox that was created.
 - We decided to replace the „Bloom/ Glow“ effect we initially thought of, in order to create more interesting obstacles
- **GPU Particle System using Compute Shader (12 points)**
 - Some of the obstacles presented (asteroids on the ground) look like they are on fire. For that we used a GPU particle system.
- **Hierarchical animation (4 points)**
 - One obstacle (the catapult) currently has a hierarchical animation

Additional Features

- **Physics Engine (12 points)**
 - the PhysX engine is used to apply physical attributes of the scene. Objects have a gravity and fall downwards.
 - The PhysX libraries were included for debug and release mode.
- **Heads-Up Display (4 points)**
 - In order to show the player his/ her progress, we are showing all the information needed such as the number of lives and the time on the display, as well as Messages regarding the End of a level (Win or Loose)
 - The text for this effect was taken from a library, downloaded from <https://www.dafont.com/de/adventure.font>

Features

The player can move and jump around. As soon as the player touches the cup, which is hidden somewhere, the level is complete. The asteroids and the catapult represent enemies/obstacles, which have to be avoided and we have also included walls that reflect the environment and thus are harder to spot. Since the game is timed, the player needs to move quickly in order to get a better time, so spotting the walls and avoiding the enemies (touching 3 enemies leads to losing the game) needs to be done fast

Objects

Simple geometries, spheres, cubes and cylinders are textured with the "default" wood texture. All objects are textured with their own texture.

The objects were downloaded from www.free3d.com and then modified, rotated and scaled using Blender. The textures were downloaded from google images and/ or modified using Photoshop and Paint.

Lighting

The illumination consists of a point light in the center of the scene and directional light

Additional Libraries

- Assimp
- PhysX
- freetyped.lib