

DEADLY LIBRARY

Description - implementation of requirements

Our game provides a basic, first level and now also a second, more interesting one. The aim is to jump from one block to another, without falling into the abyss. In case of missing a block, the player is falling down and loses life (the hearts displayed at lower left). Lost all lives? The raccoon becomes a delicious meal for the unknown being. Otherwise, by reaching the opposite wall, the player wins against the terrific threat and the raccoon gets away with it's life.

PhysX is integrated. The library is primarily used in our PhysicsPipeline.cpp/.h. and Player.cpp/.h. For the player we implemented a CharacterController with PhysX help. In combination with our Player.cpp/.h, inputHandler.cpp/.h and gameCameraHandler.cpp/.h, the camera follows the player while moving.

Our map for the second lvl can be found in assets/2.lvl. To read this map, we have our LevelReader.cpp/.h. The blocks, marked as numbers 1-9, use the PhysicsPipeline.cpp/.h. The different numbers are responsible for the different block-heights.

Our gameloop is implemented in loop/Gameloop.cpp/.h. It starts the gameState/dL.cpp/.h coordinating the whole gameplay. With the input/*Handler.cpp/.h the inputs are handled.

We also implemented the Default-Camera, to look around freely.

The adjustable Parameters can be found in settings.ini.

The heads-up display shows the amount of lives the player has, also debug information (F1) and FPS (F2) will be displayed in the hud.

Now, we included some new shaders, and other things. Our status:

“Features”:

- Hud at Beginning (storytelling)

- Lives (3 hearts)

- Custom input level

- Win/Loose Condition

- raccoon: walking-Animation while being in touch with the floor; is able to jump; shows cel shading

- After reaching the win- or loose-Condition there is the possibility to restart the game without having to quit and start anew.

Implemented Effects:

- Heads-Up Display

- Physics Engine

- Light Mapping +In-Game-Calculation (1.0) -> blocks

- Normal Mapping (0.5) -> bookshelves, door

- Cel Shading (0.5) -> player

- Subdivision Surfaces (raccoon - 3 steps) (2.0) -> player

Control Keys:

W,A,S,D	control Player	F3	WireFrame on/off
SPACE	Player jumps	F4	NormalMapping on/off
ESC	Quit Game	F5	LightMap on/off
TAB	Debug-Camera	F6	SubdivisionSurfaces on/off(3 steps)
F1	Help	F7	CelShading on/off (raccoon)
F2	FrameTime on/off	F8	View-frustum Culling on/off

Illumination - how & which objects

The blocks are illuminated by some point light's.

All objects are textured.

Textures

The blocks are textured with wood- and bricks-texture. The walls and the door are textured separately by depth, normal and color.

The raccoon is textured with grey, his backpack with brown color.

Additional Libraries / including references(URLs)

- tinyobjloader, a simple lib to import wavefront.obj files
<https://github.com/syoyo/tinyobjloader>
- stb_image, a simple lib to load image files
<https://github.com/nothings/stb>