Assignment 2
CSE251 - Graphics, Spring 2018

Deadline: March 6th, 7 PM

The Problem:

The previous assignment dealt with a 2D world. Now, we will move onto exploring 3D worlds. In this assignment, you will be trying to emulate a Legend of Zelda game. The goal of your game is to obtain the maximum amount of points in the duration of your game. You are Link and you operate one of the finest boats in the Great Sea. The game can be seen from multiple views controlled by both mouse and keyboards. The requirements mentioned are minimal so please be creative and add interesting features.

The 3D World:

The world consists of a large sea which has several enemies and objects present. Your boat must have a sail and a cannon that shoots fireballs. A booster that speeds up the boat for some time can be obtained from a monster boss drop. The world should be sufficiently big to incorporate all the elements of the assignment. Floating barrels would be present with gifts that give different amounts of points on top of them which can be collected by jumping over the barrels.

Your boat must have health points. Sea monsters would spawn and on killing them, would drop gifts. Ensure that you use different kinds of monsters in your game. The sea monsters can damage the boat and reduce your health points. The game ends when you lose all your health points. After killing several sea monsters, a monster boss would spawn and on killing would

==drop the booster.== The drops would float on the water and you must go over them to pick them up.

There would be rocks present in the sea and if you do not maneuver around them, you will lose health points. At random times, ==a strong wind would start blowing in any particular direction.== Your boat must change its speed according to the direction of the wind. Mechanics of the boat should be similar to the link: https://www.youtube.com/watch?v=-F6mewS47Fg#t=10m15s.

Controls:

Your boat should be able to move forward, jump, turn left and turn right. For uniformity, use arrow keys for move and rotating of the boat, ==space bar for jumping== and the ==f key for fireballs. Positioning the aim of the fireballs should be done by cursor.== Define other keys as per your convenience and mention them in your README file.

Additionally, the following camera views have to be incorporated:

• Boat view: This is a view from the ship's position where only a part of the world in front is visible. In other words, in this view, we see what the ship sees, as if we were the ship.

• Top View: This is a top-down view, as if we were looking vertically downwards from a position in the sky. This gives a clear picture of the path.

 • Tower view: Here, the camera is sitting on a tower, to the side of the plane of playing, observing it at an angle.

• Follow-cam view: This is a view of the ship and the region in front of it from a location behind and above it, as if the camera is following the ship.

• Helicopter-cam view: Here, the camera is movable with the mouse in an intuitive manner. Clicking and dragging should change the look angle, the up vector should remain up always,

and the scroll wheel will move the camera closer or farther away from the scene.up vector should remain up always, and the scroll wheel will move the camera closer or farther away from the scene.

Grading:

You will be graded based on the correctness and efficiency (speed) of the implementation of the elements described above. Grading would take place in several stages.
Version 1.0: A static world (Water, boat, rocks)
Version 2.0: Movement of the boat, wind, collisions with rocks
Version 3.0: Cannon, Monsters (including boss)
Version 3.5: Barrels, booster, gifts, health points
Version 4.0 (BONUS): An island where Link can disembark from the boat. A treasure chest on the island which can be opened for a gift with a high point reward. Be creative. You can make some monsters drop upgrades to your ship and fireball cannon. Further, you can experiment with more camera views, audio, animations for attacking and taking damage, textures, etc.

Submission:

You submissions should include your source code and a makefile. Do not use any non-standard libraries. In addition to these, include a file named README.md in the submission that gives a one page description of the game and how to play it.
Details of how to submit and any modification to the above submission details will be posted by the TAs towards the submission deadline.
This assignment will take time to complete. Start early.
All error scenarios must be gracefully handled (Games crashing during testing will be penalised).
Plagiarism in any form shall not be tolerated (MOSS will be used) and a straight F grade for the course will be given.