CMIS 3150: Game Programming & Development I
CSC4820/6820: Interactive Computer Graphics

Project 2

In this project, you will develop a simple 3D first-person shooter game. You will learn how to implement first-person player control, ranged weapon/projectile, collectable objects, game AI/NPC navigation, UI, sound effects, and visual effects.

Deadline: 11:59 pm, 10/28/2024

Requirements:

1. This is an individual project.

2. You must use Unity and C#. The project must involve programming.

3. Player character
    a. You must use first-person control.

    b. The main character should be able to do the following:
        i. walk (WASD keys or the arrow keys),
        ii. run (SHIFT key + WASD or arrow keys)
        iii. jump (SPACE key)
        iv. switch weapon (Q or E key)
        v. fire weapon (left mouse button)
        vi. look around (mouse motion)

    c. The character should have three lives.

4. Weapons
    a. There are two types of weapons in this game: gun and grenade.
    b. To kill an enemy with a gun, you need to hit the enemy three times.
    c. To kill an enemy with the grenade, you only need to hit it once.
    d. You do not have any weapons at the start of the game. You will need to find them and pick them up by walking or running over them.
        i. There is only one gun in the game. The gun has six bullets in it. There is additional ammunition (bullets) in the game world for the player to pick them up.
        ii. There are multiple grenades in the game for pickup.
    e. You can use the 3D weapon models from the Unity Asset Store or other websites. The 3D models do not need to be precise.
    f. The weapons can be attached to the avatar from the beginning but are invisible. When you pick up a weapon, the attached weapon becomes visible.

5. Game world and gameplay

a. There are three treasure boxes. Each treasure box is protected by one or more armed guards. You decide how many guards are needed for each box.

b. The player's goal is to collect all three treasure boxes and eliminate all the guards.

c. Each treasure box has an invisible "protection zone" (a circle). A guard patrols randomly within the protection zone.

   Once the player is inside the "protection zone" of a treasure box, the guard (NPC) who protects the treasure box will chase the player and shoot at the character.

   Once the character is outside the "protection zone," the guard will return to patrol.

d. One of the treasure boxes must be inside a house. The player can only enter the house through a door. There is at least one guard outside the house and one guard inside the house. The player must enter the house to get the treasure box.

e. If the player steals a treasure box, the guard protecting the box will pursue the character wherever the player goes, even outside its protection zone. The guard will shoot at the player during the chase.

f. The player has three lives. If the player is hit once, the player loses one life. If the player loses three lives, the game is over.

g. The player can collect health kits to restore lives. One health kit can restore one life.

h. After the player collects all three treasure boxes, the remaining guards will stop chasing the player but try to flee from the player. Now the player's goal is to hunt and eliminate the guards.

i. The player wins the game when all three treasure boxes are collected and all the guards are eliminated.

j. Only one level is required.

6. Sound effects are required. Sound effects should be played in the following events:
   a. Shooting the gun
   b. Shooting or throwing the grenade
   c. Grenade explosion
   d. Hitting a guard
   e. You are hit

f. Collecting a treasure box
g. Background music
h. The sounds of the player running and walking are optional but nice to have.

7. A Graphical User Interface (GUI) is required. The following information should be displayed in the GUI:
    a. The number of lives left
    b. Weapons you have collected
    c. The number of bullets left
    d. The number of kills

8. Visual effects are optional but nice to have.

9. Lighting: You can add multiple point lights or sunlight in Unity Editor. Experiment with multiple lights.

10. You must build a standalone player of your game for Windows so I can play it without having to open your project in Unity. If you use a Mac, you can create a Mac version. But Windows is preferred.
    a. For instructions on how to create standalone players, visit https://docs.unity3d.com/Manual/PublishingBuilds.html .
    b. You should test your standalone player before submitting it. Make sure you include all the necessary files, including data files. If I cannot run your game, I may deduct points.

11. Create a short gameplay video with at least one successful playthrough and one failure. You may record your voices to explain your design and gameplay. You can record separate videos and edit them together.
    a. You can easily record gameplay on Windows 10 or 11 (see this tutorial) and Mac (see this page). I use OBS Studio to record my lectures, and I find it easy to use and reliable.
    b. You may submit the video file to iCollege or upload the video to YouTube (or another cloud server) and submit a link.

12. Grading guideline
    a. Game assets (max 50 points). Game assets mean the entire Unity project of this game. If you only submit the assets but not a standalone player or gameplay video, I assume your game is unfinished. You will get at most 50 points, depending on how much work you have done.
        i. Without a playable standalone player, it's difficult for me to check what you have implemented. You will need to write a report detailing what you have implemented.
    b. Standalone player (40 points).
    c. Gameplay demo video (10 points)
    d. Game assets + Standalone player + Gameplay video = 100 points

13. Grading guideline
    a. Game assets only (max 50 points).
        i. If you only submit assets without a standalone player or gameplay video, I assume your game is unfinished. You will get at most 50 points, depending on how much work you have done.
        ii. Without a playable standalone player, it's difficult for me to check what you have implemented. You will need to write a report detailing what you have implemented.
    b. Game assets + Standalone player (max 90 points)
        i. If you fail to submit a gameplay video, assuming your standalone player works, you will lose 10 points.
        ii. For the grading of individual game features, see C.
    c. Game assets + Standalone player + Gameplay video (max 100 points)

       Assuming you submit everything and the standalone player works, the individual game features are graded as below.
        i. First-person or third-person player control (10 points)
        ii. Can the player character collect weapons, ammunition, and health kits? (10 points)
        iii. Can the player shoot the gun and throw the grenades to damage and kill the enemies? (20 points)
        iv. Can the guards patrol the protected zones? (10 points)
        v. Can the guards chase the player? (10 points)
        vi. Can the guards shoot at the player? (5 points)
        vii. Can the guards flee from the player? (5 points)
        viii. Sound effects (10 points)
        ix. UI (10 points)
        x. Other features (10 points)

14. Deliverables. Your final submission should includes the following:
    a. A zip file that contains the standalone application (executable file and related files and data folder)
    b. Gameplay video or a file that shows the link to your gameplay video
    c. A zip file that contains your entire Unity project
        i. All the game assets and the source code
        ii. Please remove unused and unnecessary packages and assets from your Unity project before submission.
        iii. If the file is too big, you can put the zip file on a cloud drive (such as Google Drive) and submit a link to it. Make sure your file is shared as "anyone with the link can view" so I can download it.
    d. Upload the files to the iCollege > Assignments, under the folder "Project 2."