

**CS4555 Introduction to 3D Game Programming**  
**Fall 2018**  
**HW1 – Getting Familiar with Unity via a Simple FPS (5%)**

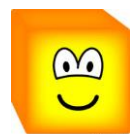
- Submission deadline: 9/25 Tuesday, 9AM
- What to submit:
  - Submit only script files. File names should be self-explanatory. Each script must have a short description.
  - Create and upload your game description document (.docx) to CSNS. The document should provide 1) team members, 2) contributions of each member, 3) used game assets (enumerate all the game assets), 4) game features, 5) steps to run and test your game, and 6) any known bugs.

Each team must present the HW1 on 9/25 (Tue).

Your reference or inspirational game is Goleneye 007 FPS. Watch a gameplay clip (00:27-01:03, 01:50-01:56) at [https://www.youtube.com/watch?v=dW\\_ElQJ6eRA](https://www.youtube.com/watch?v=dW_ElQJ6eRA).

Add the following features to the toy FPS introduced in the class.

- ✓ Static environment:
  - Make a room similar to the environment of the inspiration game (watch 00:27-01:03) with walls and large crates where an enemy shooter or the player can hide.
  - Apply visually appealing texture to each object in the room.
  - Use new 6-sided Skybox textures.
  - Add to the scene static objects for pick-up and hot spots.
    - Find and import at least three new static models.
    - Each object must have visually appealing textures.
    - Make two hotspots in the room. When the player reaches a hotspot, transport the player to a random location and continue the play.
- ✓ Player:
  - Limit the player's up and down rotation further.
  - Change (toggle) the player's weapon (gun) upon the right click. Provide different visual effects (e.g different bullet holes) for weapons. Make the bullet hole match the environment visually.
  - The player's health shall be decreased after getting any attack.
  - The player's health shall be increased after acquiring any pick-up objects. The game should remove the static pick-up objects.
- ✓ NPCs:
  - Do NOT use any humanoid-like model for NPCs. Stick to use simple shapes like a cube. Use a texture for each NPC prefab (e.g. scaled cube). The selected texture should be able to indicate the front face of the character.
  - Create three different NPCs
    - Enemy 1 - Modify your current WanderingAI enemy to move and attack the player. When this enemy detects the player in its vicinity, the enemy must look at (turn to) the player and move towards the player while shooting fireballs. The player can kill this enemy by shooting two times. Do NOT use any built-in Pathfinding library for moves.



- Enemy 2 – This is another enemy type. This enemy will always hide behind a crater. When this enemy detects the player in its vicinity, the enemy must look at (turn to) the player and shoots the player.
- Bystander – Create a bystander character. This character doesn't do anything. This character runs away if it sees the player (i.e the player is in this character's line of sight). Do NOT use any built-in Pathfinding library for moves.

✓ Scene:

- Start the game with 5 NPCs.
- Spawn 1~3 NPCs when any NPC gets destroyed. The total number of NPCs in the scene should not exceed 10. Control spawn speed so that the scene is not occupied by 10 NPCs all the time. The type of a spawned character should be random.
- Up to 3 enemies can attack the player at a time.
- Handling collision among enemies is not required.
- Provide a keystroke for game "reset". The game will restart with the initial setting for the player, enemies, and the environment. Do NOT use Unity's scene control functions (such as LoadScene) for this.
- When the player's health is 0, stop the game. The program should stop processing player moves, player shooting, and enemy's attack. Player's rotation and enemy's moves must be allowed.