**Assignment #2: Where’s Waldorf?**

The Waldorf octuplets have been trapped within an interdimensional cube. Can you find where all the Waldorf’s are hiding?

In this assignment students must create 8 cameras within Unity, 1 for each room where Waldorf is hiding. All cameras MUST be placed within the “starting\_point” GameObject of each room. Each camera will have specifications provided that MUST be followed to find Waldorf.

**Objective:**

* Ensure students are familiar with using and navigating the Unity tool
* Provide basic understanding of input and cameras with Unity
* Familiarize students with create scripts in Unity

**Required Resources:**

* Waldorf Unity Project (Unity 2018.2.3f1)

**Project Structure:**

* There are 8 total rooms
  + Each room has:
    - “room\_geometry”: fixed geometry for the stage
      * CANNOT alter for final project
      * You may enable/disable objects for visibility
    - “hiding\_spot”: the location of Waldorf in the room w/ some props
      * CANNOT alter for final project
    - “starting\_point”: this is the starting location for the room camera
      * Camera object MUST be created here for each room
  + Rooms may NOT be altered for final submission
  + Be careful to not mess up your room layout (undo is your friend)
* Main Camera: the default view when entering the game (do not alter)
* Directional Light: the default light (do not alter)
* Starting Scripts (No changes to these scripts are allowed):
  + GameCamera: base camera type all other cameras should inherit from
    - See class for function description
  + MainCamera: simple camera control attached to Main Camera object
    - For reference on other camera setups
  + FindRoomController: controls turning Waldorfs on/off in the Find Room
  + IsoRoomController: controls hiding the wall based on camera in Iso Room
  + SpaceRoomController: controls moving the object in Space Room
  + WaldorfController: controls moving Waldorf by path in Follow Room

**The Cameras:**

* Each camera should:
  + Reset position, rotation, and default properties when selected
  + Start in “starting\_point” in each room
  + Perform the required specifications ONLY
  + Should all inherit from GameCamera
* Camera #1: “pz\_camera” - the camera for “P&Z Room”
  + Provide a movement speed property
  + Controls
    - Pan Left: “A” key is down
    - Pan Right: “D” key is down
    - Zoom Forward: “W” key is down
    - Zoom Back: “S” key is down
    - Pan Up: “E” key is down
    - Pan Down: “Q” key is down
* Camera #2: “yaw\_camera” - the camera for “Yaw Room”
  + Provide a movement speed property
  + Controls
    - Yaw Left/Right: with Mouse Axis X
* Camera #3: “roll\_camera” - the camera for “Roll Room”
  + Provide a movement speed property
  + Field of View = 50
  + Controls
    - Roll Right: Mouse wheel down
    - Roll Left: Mouse wheel up
* Camera #4: “pitch\_camera” - the camera for “Pitch Room”
  + Provide a movement speed property
  + Field of View = 30
  + Controls
    - Pitch Up: Mouse Axis Y forward (away from user)
    - Pitch Down: Mouse Axis & back (towards user)
* Camera #5: “find\_camera” - the camera for “Find Room”
  + Controls
    - Look at Waldorf: “Fire1” button / left mouse
      * Must look at the currently visible Waldorf with each click
        + Can use tag=Waldorf or reference FindRoomController
      * Must not rotate the Camera on the z-axis
      * Field of View should change based on visible Waldorf
        + See FOV in FindRoomController
* Camera #6: “iso\_camera” - the camera for “Iso Room”
  + Default camera is perspective projection
  + Solid background color – student’s choice of color
  + Controls
    - Set Perspective Camera: “Fire1” button / left mouse
    - Set Orthographic Camera: “Fire2” button / right mouse
      * Orthographic size = 3
* Camera #7: “follow\_camera” - the camera for “Follow Room”
  + Default camera is not returned to “starting\_point”
  + Provide a follow offset vector
  + Controls
    - FPS view: “Fire1” button / left mouse
      * Camera must follow the position and view of Waldorf’s head
      * Camera must not have clipping or show Waldorf’s head
    - Follow view: “Fire2” button / right mouse
      * Camera must follow Waldorf in an over the shoulder view
        + Using the offset vector
      * Camera must face the same direction as Waldorf
      * Camera must be behind Waldorf with no clipping issues
* Camera #8: “space\_camera” - the camera for “Space Room”
  + Provide a movement speed for the camera
  + Provide a minimum distance to keep between the object and the camera
  + Provide a customizable key to trigger
  + Provide a time to update the colors
  + Hold Mode
    - Do not move the camera
    - Use default color for the background
  + Follow Mode
    - Move the camera towards the “hiding\_spot” object
    - Should move at the movement speed
    - Must never exceed the minimum distance
    - Should alternate background colors over time when moving
  + Controls
    - Follow mode: Custom key is down
* Camera Switches
  + MUST provide a method for me to switch between cameras
    - 1-8 on the top of the keyboard: each of the room cameras
    - 0: the default camera
  + May use any method to achieve this result
    - I recommend using some sort of Game Controller

**Submission:**

* Completed Unity Project
  + Only “Assets”, “Packages”, and “Project Settings” folders
  + Naming convention for project folder: gat240\_studentid\_2
  + Submit as zip: gat240\_studentid\_2.zip

**Rubrics:**

* **Each Room – 10% each (80% total)**
  + Follow the general specifications
  + Follow each cameras specific specifications
* **Camera Switches – 10%**
  + Can switch to all camera correctly
* **Cleanliness – 10%**
  + Clean project structure
  + Clean code structure
  + Followed naming conventions
  + Explained complicated code

**Notes:**

Remember you will get some points for just making the camera and adding it to the proper location. So, even if you can’t figure out a camera, do try your best!