**Unit Testing**

**Shao Liang**

short line

**Module**: Environment Objects Geometry

**Classes:**

* buildField.cs, fieldSettingWide.cs

**Expect Outcome:**

* Environment objects should be built along the straight path between starting collider and ending collider.
* Obstacles should be in a reasonable position

**Test Case:**

* By observation in game scene.
* Ending collider is set to different positions via Inspector > Transform > Position parameters in unity framework. Building blocks (Wallblock) are able to built at straight from starting collider to exactly where ending collider located.
* Over 1000 obstacle are being instantiated and all within correct position (not totally outside of the tunnel, leaving it unseen by the player/ not totally inside the tunnel, detach from the wall)

**Module:** Environment Instantiation

**Classes:**

* onEnter.cs, onExit.cs

**Expected Outcome:**

* field segment should be instantiate attached to the previous segment next segment should be instantiated when entering the start collider current segment should be destroyed when entering the end collider

**Test Case:**

* when player entering the start collider, print in console "player entered, build next section" via onEnter.cs
* when player entering the end collider, print in console "player left, destroy section" via onExit.cs
* field prefab is instantiated without Environmental objects (only start collider and endcollider is included in the prefab)
* by observation, field segment is instantiated with its start collider attached to the end collider of the previous segment, prompts are correctly triggered at the right timing.

**Module:** Environment Instantiation

**Classes:**

* difficultySettings.cs, environmentMovement.cs, pauseOnStart.cs

**Expected Outcome:**

* object with environmentMovement.cs as a component will move along z-axis with manually set speed
* object should not move for a period of time right after game starts.
* object should accelerate as game progress, and stop acceleration at a manually set final speed.

**Test Case:**

* current speed is printed in console via difficultySettings.cs, by observation, speed is accelerating properly, and speed will not increase when reaching speedCap
* environmentMovement is attached to a simple cube as tested object

By observation in game scene, cube is moving along z-axis.

**Module:** Environment Instantiation

**Classes:**

* EndGame.cs

**Expected Outcome:**

* when player hits the wall, obstacle, or enemy objects, game is over

**Test Case:**

* simple cube object is instantiated and tagged as "Wall", "Enemy", or "Enemy\_Active", make it collide with player in a simple game scene. Scene is ended properly.

**Module:** Score Display

**Classes:**

* scoreKeeper.cs, KillCountDisplay.cs, ScoreDisplay.cs, TimeDisplay.cs

**Expected Outcome:**

* after game overed, score, kill count, time survived should be displayed
* high score, highest kill count, longest time survived should be displayed
* if new record is made, highest should be the same as for the previous round

**Test Case:**

* independent from the game play, score and high score (kill count, time corresponsive) are assigned a value in scoreKeeper.cs via Start() method, and Update() the data and immediately end the game to the end scene where everything is displayed.
* different scenarios are tested and score show up correctly, high score update correctly. (kill count, time corresponsive)