T2 New Mechanic

Customizing Flappy Bird

The Tutorial 2 assignment requires you to implement a new feature into flappy bird. You must implement the following.

Multiplayer Mode

A new mode that allows two players to cooperate / compete side-by-side.

- There must be two birds, each with a distinct aesthetic (so players know which bird belongs to which player).
- One bird must be controlled by mouse click, while the other is controlled via the spacebar (KeyCode.Space)
- The game continues until both players are knocked out. A player that is knocked out is left behind.
- The two birds may not collide with each other at all and may not knock one another out.
- Each bird must contribute to scoring in some way (when a bird passes through a pipe safely, it must affect the score in some way).

Remember to playtest / debug your game thoroughly!

Hint #0: A Second Component?

Should you find yourself creating a second “Bird.cs” component (“Bird2.cs”, perhaps), you’re cooking spaghetti. Can you think of a way to re-use your existing Bird.cs component with some small tweaks? What happens in the inspector window if you write “public int player_number;” at the top of the Bird.cs component? Can you use that new variable to change how each bird works?

Hint #1: Creating the second bird

What are prefabs useful for? Does your bird gameobject have a prefab? You may learn more about prefabs here.

Hint #2: Changing a Bird's color

In the Inspector, take a look at the SpriteRenderer component on your bird gameobject. Do any of these properties seem relevant to making your second bird look different? You may learn more about the SpriteRenderer component here.

Hint #3: Changing the failure condition

In single-player flappy bird, the Bird component calls the GameControl singleton to end the game (GameControl.BirdDied()). Could the GameControl singleton be altered to require 2 calls (one from each bird) before ending the game?

Hint #4: Double Knockout?

When a bird touches the ground, or a pipe, it gets "knocked out". If both birds are "knocked out", the game should end. If a bird is "knocked out" by a pipe, when it falls to the ground, does it get "knocked out" again?

Hint #5: Preventing Friendly Fire

Collision layers may be used to tell Unity "gameobjects on these two layers should not collide with one another". How can this feature of Unity be used to implement your multiplayer mode? Be careful to make your changes with respect to the 2D physics system, rather than the 3D one.