Assignment 0 – Pong

* Implement a basic AI for either Player 1 or 2 (or both!).

Assignment 1 – Flappy Bird

* Influence the generation of pipes so as to bring about more complicated level generation.
* Give the player a medal for their performance, along with their score.
* Implement a pause feature, just in case life gets in the way of jumping through pipes!

Assignment 2 – Breakout

* Add a powerup to the game that spawns two extra Balls.
* Grow and shrink the Paddle when the player gains enough points or loses a life.
* Add a locked Brick that will only open when the player collects a second new powerup, a key, which should only spawn when such a Brick exists and randomly as per the Ball powerup.

Assignment 3 – Match 3

* Implement time addition on matches, such that scoring a match extends the timer by 1 second per tile in a match.
* Ensure Level 1 starts just with simple flat blocks (the first of each color in the sprite sheet), with later levels generating the blocks with patterns on them (like the triangle, cross, etc.). These should be worth more points, at your discretion.
* Creat random shiny versions of blocks that will destroy an entire row on match, granting points for each block in the row.
* Only allow swapping when it results in a match. If there are no matches available to perform, reset the board.

Assignment 4 – Super Mario Bros.

* Program it such that when the player is dropped into the level, they’re always done so above solid ground.
* In LevelMaker.lua, generate a random-colored key and lock block (taken from keys\_and\_locks.png in the graphics folder of the distro). The key should unlock the block when the player collides with it, triggering the block to disappear.
* Once the lock has disappeared, trigger a goal post to spawn at the end of the level. Goal posts can be found in flags.png; feel free to use whichever one you’d like! Note that the flag and the pole are separated, so you’ll have to spawn a GameObject for each segment of the flag and one for the flag itself.
* When the player touches this goal post, we should regenerate the level, spawn the player at the beginning of it again (this can all be done via just reloading PlayState), and make it a little longer than it was before. You’ll need to introduce params to the PlayState:enter function that keeps track of the current level and persists the player’s score for this to work properly.

Assignment 5 – Legend of Zelda

* Implement hearts that sometimes drop from enemies at random, which will heal the player for a full heart when picked up (consumed).
* Add pots to the game world (from the tile sheet) at random that the player can pick up, at which point their animation will change to reflect them carrying the pot (shown in the character sprite sheets). The player should not be able to swing their sword when in this state.
* When carrying a pot, the player should be able to throw the pot. When thrown, the pot will travel in a straight line based on where the player is looking. When it collides with a wall, travels more than four tiles, or collides with an enemy, it should disappear. When it collides with an enemy, it should do 1 point of damage to that enemy as well.

Assignment 6 – Angry Birds

* Implement it such that when the player presses the space bar after they’ve launched an Alien (and it hasn’t hit anything yet), split the Alien into three Aliens that all behave just like the base Alien.

Assignment 7 – Pokémon

* Implement a Menu that appears during the player Pokémon’s level up that shows, for each stat, ‘X + Y = Z’, where X is the starting stat, Y is the amount it’s increased for this level, and Z is the resultant sum. This Menu should appear right after the “Level Up” dialogue that appears at the end of a victory where the player has indeed leveled up.

Assignment 8 – Helicopter Game 3D

* Add Gems to the game that spawn in much the same way as Coins, though more rarely so. Gems should be worth 5 coins when collected and despawn when off the left edge of the screen.
* Fix the bug whereby the scroll speed of planes, coins, and buildings doesn’t reset when the game is restarted via the space bar.

Assignment 9 – Dreadhalls

* Spawn holes in the floor of the maze that the player can fall through (but not too many; just three or four per maze is probably sufficient, depending on maze size).
* When the player falls through any holes, transition to a “Game Over” screen similar to the Title Screen, implemented as a separate scene. When the player presses “Enter” in the “Game Over” scene, they should be brought back to the title.
* Add a Text label to the Play scene that keeps track of which maze they’re in, incrementing each time they progress to the next maze. This can be implemented as a static variable, but it should be reset to 0 if they get a Game Over.

Assignment 10 – Portal

* Create your own level in a new scene using ProBuilder and ProGrids!
* Ensure that the level has an FPSController to navigate with in the scene.
* Ensure that there is an object or region with a trigger at the very end that will trigger the end of the level (some zone with an invisible BoxCollider will work).
* When the level ends, display “You Won!” on the screen with a Text object.