Session 11 Start

Core features in the game are all the features in the minimum viable product (the basic game that is enjoyable to play without all the extra additions).

Before we start implementing the core features in the game, we need to know what we want to add, or at least a simple idea to expand on.

We need to think about:

* What are the main features we want in the game?
* What are these features going to do?
* What is the algorithm for these features?
* How can I implement this?
* Can I write this?
* What do I need to write it?
* Repeat for all the features then start writing/programming.

So, for example, lets think about your game. You have said that you want a game where there are objects that move along the screen and the player must somehow jump to each one before he reaches the end. There are some simple features that is already stated in this small sentence, make sure to read each word carefully as it can indicate what you need.

You have said that you want a game where there are objects that move along the screen and the player must somehow jump to each one before he reaches the end.

*Objects that move along the screen*: We need an algorithm to move these objects in the world  
*Must somehow jump to each one:* Again, we need to decide an algorithm for a player, can be move left/right, what about just up/down? It is up to you to decide this  
*Reaches the end:* What happens when the player reaches an end? Does he take damage, lower score, die? How can we implement these ends in the game?

We are going to focus on the first feature that we have found (Objects that move along the screen) and we are going to follow the steps that we stated above. We have already discovered the main features and now it is time to find out what this feature is going to do.

Below is a good table to get you started on designing the implementation of a feature into a program or game.

|  |  |
| --- | --- |
| **Feature Name**  *This is not required; it just helps you to structure your planning* | Object Movement |
| **What does it do?**  *Provide a description on everything this feature needs to be able to do, make sure that each task is unique to this feature and that it can’t be divided further into other features.* | * Move an object from the right-hand side of the screen to the left * Destroy the object when it has completed this |
| **What is an algorithm for the feature?**  *What are the steps the computer needs to take to be able to do this?* | * Create a variable for the start and end points and for the object and for speed * Teleport object to start point * Move the object each frame towards the end position based on the speed * Delete the object when it has reached the destination |
| **How can I implement this?**  *What functions and methods do I need to know, what objects do I need to put in the scene?* | I will need a:  Start Position Gameobject End Position Gameobject  ObjectController Gameobject  ObjectMovement Script  Using a transform and vectors moveto function  *Don’t forget Unity’s documentation!* |
| **Do I have the ability to create this?** | I know how to create objects and components. I can use the documentation to learn the required methods. The algorithms are not complicated, and I understand them.  I can create this. |

A basic implementation of this can be (please see script in github):

|  |
| --- |
| **ObjectController.cs** |
| // Get the start and end positions  public GameObject startPosition;  public GameObject endPosition;  // How fast we want the object to move  public float moveSpeed= 1f;  // A list of all the objects to move  private List<GameObject> \_objectsToMove;  // Start is called before the first frame update  void Start()  {    // Initialise the list  \_objectsToMove = new List<GameObject>();  // Find all the objects that we need to move  // These objects have the MovingObject tag (Don't forget to  // add that if you are using this script!)  GameObject[] currentMovingObjects = GameObject.FindGameObjectsWithTag("MovingObject"); // Get all current moving objects in the scene    // Loop through the array and add it to the list  for (int i = 0; i < currentMovingObjects.Length; i++) {  // Add object to the list  \_objectsToMove.Add(currentMovingObjects[i]);  // Set the start position of the object  currentMovingObjects[i].transform.position = startPosition.transform.position;  }  }  // Update is called once per frame  void Update()  {  // Move each object towards the end position  foreach(GameObject go in \_objectsToMove)  {  // See <https://docs.unity3d.com/ScriptReference/Vector3.MoveTowards.html>  go.transform.position = Vector3.MoveTowards(go.transform.position, endPosition.transform.position, moveSpeed \* Time.deltaTime);  }  // If the object is at the end position, remove it from the list and destroy it  foreach (GameObject go in \_objectsToMove)  {  if(Vector3.Distance(go.transform.position,endPosition.transform.position) <= 0.001f) {  \_objectsToMove.Remove(go);  Destroy(go);  }  }  } |

This script does work, however there are some flaws with the script:

* It does not reset the objects positions if an object is added to a list after the game has started
* There is no stagger time so all the objects are stuck together and move towards the end position in one big lump
* It could be optimised because accessing object.transform.position uses a lot of words and memory

You would notice that these flaws are not something that is immediately discovered when you create the algorithm, but something you would discover when you create the script. These errors and flaws are not entirely obvious sometimes.

Try it!

Write a paragraph describing the core features of your game.

|  |
| --- |
| The core features of my game include, speed, jumping and quick thinking. Speed; This is a core feature because speed will be needed to be able to quickly move from platform to platform. Jumping; umping will be a core feature because we will need to jump from platform to platform, coming in with the first core feature, speed, being used for air movement. Our 3rd core feature will be quick thinking, this will be needed because over a small amount of time the platform will disappear making the player fall and lose, however, using the quick thinking feature the player will use this to once again jump platform to patform without losing. |

Analyse your paragraph and find the main features

|  |  |
| --- | --- |
| 1 |  |
| 2 |  |
| 3 |  |
| 4 |  |
| 5 |  |
| 6 |  |
| 7 |  |
| 8 |  |

Create the tables for each of those features

**Feature 1**

|  |  |
| --- | --- |
| **Feature Name**  *This is not required; it just helps you to structure your planning* |  |
| **What does it do?**  *Provide a description on everything this feature needs to be able to do, make sure that each task is unique to this feature and that it can’t be divided further into other features.* |  |
| **What is an algorithm for the feature?**  *What are the steps the computer needs to take to be able to do this?* |  |
| **How can I implement this?**  *What functions and methods do I need to know, what objects do I need to put in the scene?* |  |
| **Do I have the ability to create this?** |  |

**Feature 2**

|  |  |
| --- | --- |
| **Feature Name**  *This is not required; it just helps you to structure your planning* |  |
| **What does it do?**  *Provide a description on everything this feature needs to be able to do, make sure that each task is unique to this feature and that it can’t be divided further into other features.* |  |
| **What is an algorithm for the feature?**  *What are the steps the computer needs to take to be able to do this?* |  |
| **How can I implement this?**  *What functions and methods do I need to know, what objects do I need to put in the scene?* |  |
| **Do I have the ability to create this?** |  |

**Feature 3**

|  |  |
| --- | --- |
| **Feature Name**  *This is not required; it just helps you to structure your planning* |  |
| **What does it do?**  *Provide a description on everything this feature needs to be able to do, make sure that each task is unique to this feature and that it can’t be divided further into other features.* |  |
| **What is an algorithm for the feature?**  *What are the steps the computer needs to take to be able to do this?* |  |
| **How can I implement this?**  *What functions and methods do I need to know, what objects do I need to put in the scene?* |  |
| **Do I have the ability to create this?** |  |

**Feature 4**

|  |  |
| --- | --- |
| **Feature Name**  *This is not required; it just helps you to structure your planning* |  |
| **What does it do?**  *Provide a description on everything this feature needs to be able to do, make sure that each task is unique to this feature and that it can’t be divided further into other features.* |  |
| **What is an algorithm for the feature?**  *What are the steps the computer needs to take to be able to do this?* |  |
| **How can I implement this?**  *What functions and methods do I need to know, what objects do I need to put in the scene?* |  |
| **Do I have the ability to create this?** |  |

**Feature 5**

|  |  |
| --- | --- |
| **Feature Name**  *This is not required; it just helps you to structure your planning* |  |
| **What does it do?**  *Provide a description on everything this feature needs to be able to do, make sure that each task is unique to this feature and that it can’t be divided further into other features.* |  |
| **What is an algorithm for the feature?**  *What are the steps the computer needs to take to be able to do this?* |  |
| **How can I implement this?**  *What functions and methods do I need to know, what objects do I need to put in the scene?* |  |
| **Do I have the ability to create this?** |  |

**Feature 6**

|  |  |
| --- | --- |
| **Feature Name**  *This is not required; it just helps you to structure your planning* |  |
| **What does it do?**  *Provide a description on everything this feature needs to be able to do, make sure that each task is unique to this feature and that it can’t be divided further into other features.* |  |
| **What is an algorithm for the feature?**  *What are the steps the computer needs to take to be able to do this?* |  |
| **How can I implement this?**  *What functions and methods do I need to know, what objects do I need to put in the scene?* |  |
| **Do I have the ability to create this?** |  |

**Feature 7**

|  |  |
| --- | --- |
| **Feature Name**  *This is not required; it just helps you to structure your planning* |  |
| **What does it do?**  *Provide a description on everything this feature needs to be able to do, make sure that each task is unique to this feature and that it can’t be divided further into other features.* |  |
| **What is an algorithm for the feature?**  *What are the steps the computer needs to take to be able to do this?* |  |
| **How can I implement this?**  *What functions and methods do I need to know, what objects do I need to put in the scene?* |  |
| **Do I have the ability to create this?** |  |

**Feature 8**

|  |  |
| --- | --- |
| **Feature Name**  *This is not required; it just helps you to structure your planning* |  |
| **What does it do?**  *Provide a description on everything this feature needs to be able to do, make sure that each task is unique to this feature and that it can’t be divided further into other features.* |  |
| **What is an algorithm for the feature?**  *What are the steps the computer needs to take to be able to do this?* |  |
| **How can I implement this?**  *What functions and methods do I need to know, what objects do I need to put in the scene?* |  |
| **Do I have the ability to create this?** |  |

Great job! Now we know what we want to make, we can start creating it for the game! If you are unsure of how to implement these features, just ask me on Friday!