Practical Gaming 2022

# Name of Student Paul Kerins

# T Number T00188667

# Name of Project PaulKerinsPG

# Gameplay

Describe how to play the game here, specify keys/mouse etc. what needs to be done to unlock further features etc.. i.e. a walkthrough which covers all of what is to be seen to be marked.

The game is played as an action-adventure game where there are pick up objects and defeat enemies.

The Character has 6 different movements.

W Key makes the character go forward

S Key makes the character go backwards

A Key makes the character go Left

D Key makes the character go right

Space Button make the character Jump

The left Shift key along with W Key makes the character sprint

The character has 2 different attacks

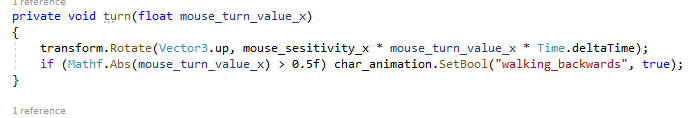
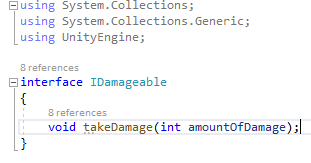
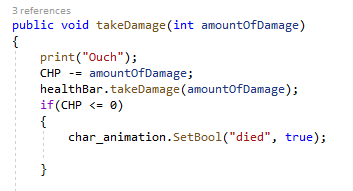
Left mouse click does a sword attack

Right mouse click does a punch attack.

In the game you must go through the level and kill all the enemies in the area and go through the door of the cabin in the woods to win the game.

# Coding

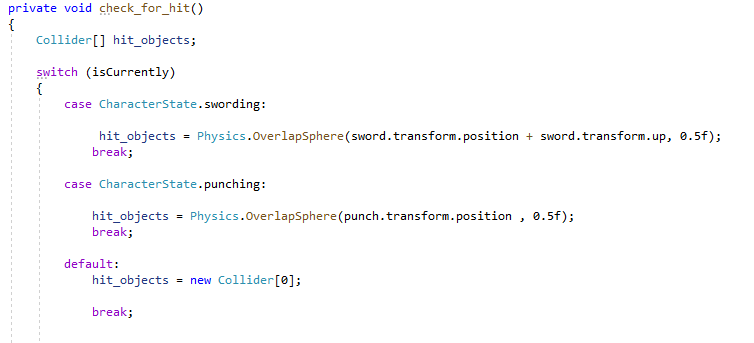
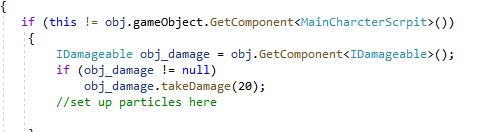
Under each of the following headings, please describe the concept, why is it or isn’t it useful/needed, where do you implement in you project, you may provide screenshots or cut and past code segments etc..

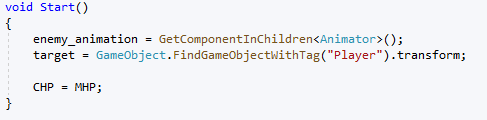
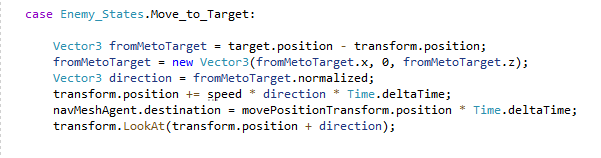
* Frame Rate Independence
  + Allows characters and camera to move in real time and makes them move smoothly
  + Frame rate independent is very useful in games because it makes the character or object move in real life time
  + I use frame rate independence when I have my characters moving and turning
  + 
* Interfaces
  + Allows other scripts to call on the method once they are all linked to the interface
  + Interfaces are useful when calling methods for a load of different object
  + I have an IDamageable interface for all the characters that can take or deal damage
  + 
* Inheritance
  + Inheritance is reusing classes in a script that is written and the objects are declared in another class without modifying them in
  + Inheritance is very useful in games as it allows you to not repeat the same code in different scripts
  + I used Inheritance in the MainCharacterScript when using the iDamageable TakeDamage Class
  + 

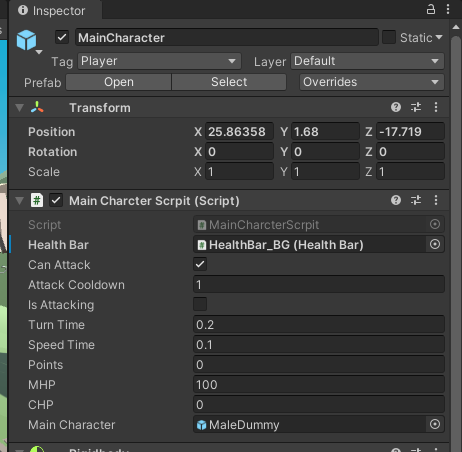
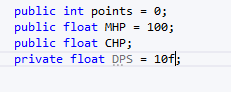
* Case pattern
  + Case patterns are very useful in games as it allows the object to change from one state to another and makes the code look much neater
  + I use case patterns for my characters

I have the enemy characters have idle, moving, attacking and died cases

And for the main character I have a sword and punch state

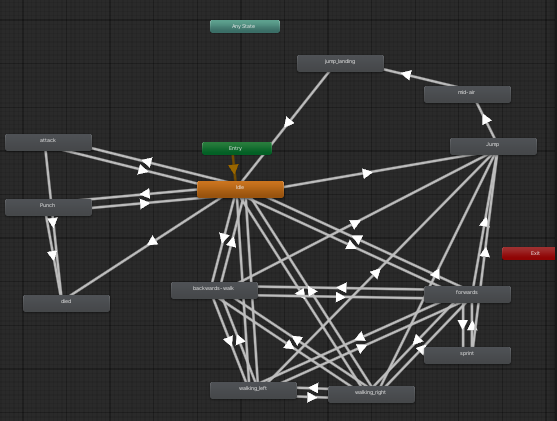
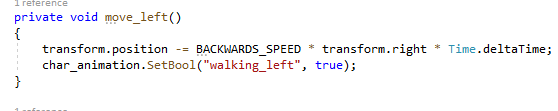
* + 
* Observer Pattern
  + //Manager
* Polymorphism
  + Polymorphism is described that you can access objects of different types through the same interface.
  + I use polymorphism in my project with the interface IDamageable for the game objects to check to see if the objects are damageable
  + 

* Communication between scripts/game objects
  + In my game the enemy script communicates with the game object target which I set to the player so the enemy will know who to attack
  + 
  + 

* Instantiation and Prefabs
  + Prefabs are objects made already with certain items attached to them that can be dragged onto the scene and are already to be used
  + For my project I have prefabs made for the enemy’s and main character with the scripts attached
  + 
  + 
* Magic Numbers
  + Magic numbers are set values that don’t change
  + I use magic numbers in my project when I have DPS for each characters own DPS
  + 

* Model Animation
  + Animations allow the models to preform movements.
  + In the game I have two different animators for the main character and the enemy.

The animators hold the Booleans and the transitions for the characters animations.

* + 
  + Within the scripts I have code that switches the status of the Bools to activate the animations at appropriate times
  + 
* Self made models and or animations
  + Self-made models and animations are models and animations made by the user themselves in another software and exported to unity
  + 
  + A person in a garment

    Description automatically generated with medium confidence
  + I made my own sword in Maya for the character to use
* Interactions between objects/scripts
  + Through out the game I have objects that interact with scripts with using colliders to see what objects the player, enemy or objects have collided with.
  + An example of this would be at the end I have a black door with a collider so when the character hits off it, it will bring you to the end screen.
  + A picture containing sky, building, outdoor, house

    Description automatically generated
  + Text

    Description automatically generated
* Propper code placement
  + Propper code placement is writing code in the right format and within the relevant scripts
  + Propper code placement can be useful as it makes the code much easier to read and allows the code to be followed much clearer
  + I used it in my scripts to make it much easier to follow what each method does with nested if statements
  + A screenshot of a computer

    Description automatically generated with medium confidence

* Code repetition
  + Code repletion is the use of code to keep iterating through the code till the wanted outcome occurs
  + This is very useful in games as it can be used in various scenarios
  + I used it in my project to check to see if my characters were hit and to see if they had no health left or if they did. If they had none left the character would then die
  + Text

    Description automatically generated
* Feature 1
  + An Extra Feature that I put into my game was UI’s such as Health Bar

And Game menus

* + A picture containing text, resort

    Description automatically generated
  + A picture containing text

    Description automatically generated
* Feature 2
* Feature 3