Assignment 2 – Player Movement Prototype
Due: 11:00pm – 18 Sep, 2019
Weighting 20%

Using Unity with C#, prototype a 2D platformer game level.

(This should be based on your design from Assignment 1 – if you wish to change you design please talk to your tutor during the tutorial)

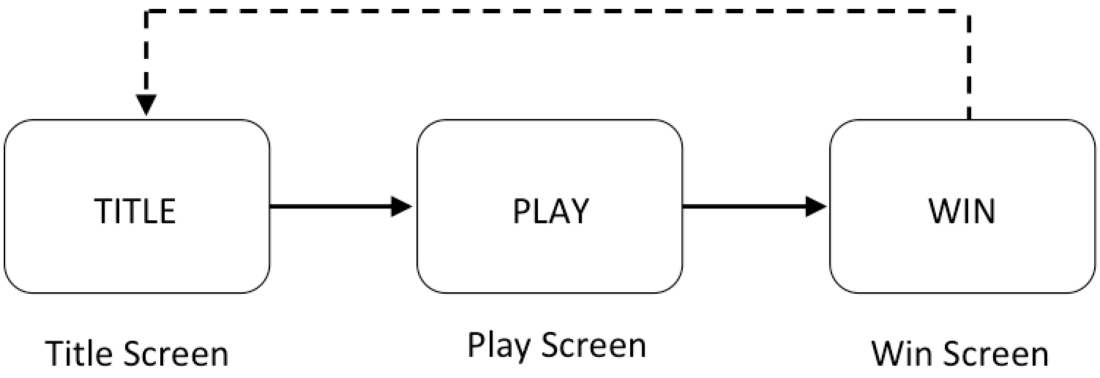
As a minimum your prototype must have the following 3 screens:

1. A **title screen** with a button to start play.

2. A **play screen** that the player should be able to navigate through from beginning to end. It should take at least 10 seconds to reach the end of the level. The level must restart or go to a checkpoint if the player cannot reach the end of the level (for instance if they fall off a platform into the sea)

   NOTE: A Checkpoint is just some point in the game at which a player may restart play if their character is killed or the player loses for some other reason.

3. Once the player reaches the end of the level the game should exit to a **win screen. (**The player should also be able to navigate from the **win screen** back to the **title screen**.)



You may either work as an individual or as a pair (two) on assignment 2 and assignment 3. (Note the requirements/conditions for working as a pair at the end of this assignment)

Submission Requirements

- Please note - Penalties will apply for failure to follow the correct submission process.
- All the required files for this assignment, including the cover sheet, report, movie, and Unity project should be placed into a folder that contains your name. For example place everything in a folder called “KeithNesbitt_assign2”. This folder should then be zipped into a single file that has the same name. For example, the zipped file might be called “KeithNesbitt_assign2.zip”. (Use both names if you are working in a pair. eg.KeithNesbitt_BobBrown_assign2.zip)
- This single zip file must be submitted on blackboard using the correct Assignment 2 link. (Only one student needs to submit if you are working as a pair).
- *A standard ZIP file must be submitted – do NOT use other compression formats such as rar. If I cannot unzip your submission in the lab I will not mark your assignment.*

Demonstration of Game

You must be available to demonstrate your working prototype in the lab on the Tutorial following the submission. If you do not demonstrate your game, the assignment will not be marked. The demonstration will be about 3-4 mins – you might be best to use the movie of your game to do this presentation.

(*Depending on unfolding COVID events* - presentation requirements may change - you will be advised of any changes in lectures)

Submission Content (to be included in your zip file)

1. **Cover Sheet**

   A standard, signed, assignment cover sheet must be submitted. This must be signed correctly and in pdf format. It must be submitted electronically. (For a pair - a group cover sheet with both signatures must be included).

2. Report

Your submission must include a brief report (1-4 pages) for your game in **pdf** format. This report must include:

- On every page you must include page numbers and your name and student number (or both names and numbers if working as a pair)
- A list of your specific contributions in terms of code (**what code you wrote**).
- You can use other code sources but you must include a **list of code references** for any reused code that is not yours. If you are found to reuse code from another source that is not referenced this will be considered plagiarism and appropriate action will be taken.
- A list of specific contributions in terms of graphics and other media (Just **what images, animations, sounds did you create**?)
- You can use other asset sources but you must include **a list of asset references t**hat clearly identifies the source of all images, sounds etc. that you

borrowed or modified. If you are found to reuse assets from another source that is not referenced this will be considered plagiarism and appropriate action will be taken.

3. Unity project:

Include your Unity project. This will be a folder/directory on your harddisk that has the same name you have given your project. Your marker should be able to open this Unity project and build and run the game. If you are not sure how to find this directory please talk to your tutor before the assignment is due.

4. A 3-5 Minute Movie of Gameplay

- In a folder called "Movie" – You should submit a short movie of your game. This can be made using screen capture software. Make sure you use a format that runs on the machines in the lab (test it).
- The movie should run for 3-5 minutes and include all the relevant game play, screens etc of your game. It should showcase your game. This movie can also be used in the final demonstration of the game in your tutorial and will be used to help mark your assignment.

Submission Checklist

Please note that marks may be deducted if you do not follow requirements. So please check the following points - failure to comply with any of these requirements may result in loss of marks.

- ❑ You have submitted on time.
- ❑ You have submitted on blackboard using the Assignment 2 link
- ❑ You have submitted a single ZIP file with the correct name (no other format!)
- ❑ You have included in this a folder a cover sheet, report in pdf, movie and your Unity project folder.
- ❑ The game you have submitted will run in the lab (have you tested it).
- ❑ You have included all the required content in your report.
- ❑ The report is in a nice format (see Appendix B – How to write a good report)
- ❑ You have included a 3-5 minute movie of the game that will run in the lab (have you tested it).
- ❑ You are available in the tutorial the week of submission to demonstrate your prototype (failure to do this will result in 0 marks).

Working as a Pair (two students)

You may complete this assignment and the next as a pair if you wish (this is not compulsory). Since you may need to combine ideas from the first assignment into a single game you should discuss this with your tutor.

Note if you have special requirements, that involve any adjustment to your program you must work as an individual.

Students who wish to work in pairs will select their own partners and make their own arrangements as to how the work is to be divided. Each student in a pair will receive the same mark for the assignment, regardless of how the workload was distributed. Note that it will be difficult to change this working arrangement for the third assignment so you are advised to choose your partner wisely.

Each assignment will be marked according to the same marking scheme, regardless of whether it is completed by one student or by a pair.

In exceptional circumstances, where one member of pair has contributed significantly more than the other, either student may make a case to the course coordinator for the students not to be given the same mark. Such a case must be supported by evidence, such as a detailed journal describing what work was done when, and by whom. If the coordinator is persuaded by the evidence, the students in the pair will be given marks that reflect their individual contributions to the assignment.

Appendix A - Assignment Marking Criteria

In general the marking criteria will consider:

- Degree of effort evident in implementation and testing of the game.
- Individuality and originality
- Look and feel of the game world (aesthetic quality).
- Evidence of applied project management skills and good programming practices.
- Effective and efficient response to the project brief.
- Presentation, structure and communication of report and demo material.

A more detailed breakdown for marks is provided below.

	Total Marks
Marks (60)	
Late Penalty (10% per day)	
TOTAL (20)	

Production Requirements	Max (10)	Mark
Demonstration (In tut the following Monday) Note that failure to provide this demonstration may mean you lose additional marks as the marker may not be able to make judgements for game content etc.	6	
Report • code contributions included • graphics, etc. contributions included • reference to reused code sources • reference to reused graphic, etc sources	4	

Production penalties (deductions for incorrect process)

Submission process is correct • correctly named zip file • signed hard copy cover sheet submitted • report /cover sheet s in correct pdf format • correct folder names	- 4	

Game Content & Experience	Max (50)	Mark
Can start the game, play through the level to the end, as specified. Is it possible for the player to get stuck? Does it meet requirements. Title screen, play screen and win screen etc.	15	
Individuality and originality. Gameplay - is it fun, is it different. Story and character	7	
Look and feel of the gameworld Is it consistent? Is it interesting?	8	
Degree of effort evident in graphics, sounds etc.	10	
Degree of effort evident in coding etc.	10	

Appendix B – How to write a good report

Your submission must include a brief report (1-4 pages) for your game in **pdf** format. This report must include:

- On every page you must include page numbers and your name and student number (or both names and numbers if working as a pair)
- A list of your specific contributions in terms of code (**what code you wrote**).
- You can use other code sources but you must include a **list of code references** for any reused code that is not yours. If you are found to reuse code from another source that is not referenced this will be considered plagiarism and appropriate action will be taken.
- A list of specific contributions in terms of graphics and other media (Just **what images, animations, sounds did you create**?)
- You can use other asset sources but you must include **a list of asset references t**hat clearly identifies the source of all images, sounds etc. that you borrowed or modified. If you are found to reuse assets from another source that is not referenced this will be considered plagiarism and appropriate action will be taken.

Basic Layout

1. Include a title (no need for a title page – but a title is good)
2. Include a header/footer on every page that has – page numbers, your name and student number.
3. A brief table of contents is optional – in this case probably not required.
4. The layout of the document should certainly have headings that match the requirements: eg.
    a. Code I created
    b. Code I reused/referenced
    c. Assets I created
    d. Assets I sourced / references
5. Don't forget that you will need to create a pdf of this for submission. You will likely be including a lot of images (eg of the assets in the game) – if your pdf gets really big you can probably just save it using the "reduced size" option for pdf files.

Report Content

Code I created

Make sure you don't leave out the detail – the code you created should provide a precise list of elements that you created eg.

1. Implement the Character walk left /right (player.cs – in Update() function)
2. Implemented the main character run left/right (player.cs – in Update() function)
3. Implemented the Characters jump functionality (player.cs – in Update() function)
4. Implemented the camera follow function (cameraFollow.cs)
5. Implemented the change of state associated with animation swapping (player.cs)

Note: Be precise and list the actual tasks and where in code you implemented them (ideally both the class and function(s). - If you keep track of these as you do them then it will be very easy to do this report – each of these is a task you did when you worked on the game.

Note: Do not write "I did everything except some bits listed below – so why don't you go and work it out" – unless you want a low grade ☺

Code I referenced

This should be a careful reference to the sources (books, papers, online) you used to help implement your prototype. There is no penalty for using sources to help you.

In some cases you may completely use an idea, function, class that was taken from online. In this case list it precisely.

1. Enemy class movement sourced from the textbook – Chapter 3 page - Introduction to Game Design, Prototyping, and Development: From Concept to Playable Game with Unity and C#, 1st Edition, by Jeremy Gibson (Author). Publisher: Addison-Wesley Professional; 1 edition (July 21, 2014)

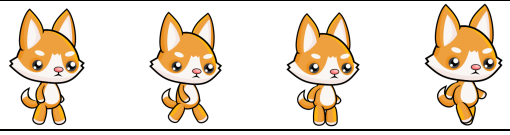
2. In other cases you may have partially used ideas from somewhere that you integrated into your solution – you can reference back to the code you wrote – eg. Task 3 character Jumping - I integrated some ideas for my character jumping movement from this online tutorial - https://www.youtube.com/watch?v=Tpak3yIkS5M

Note: Be precise and list the actual tasks and where in code you implemented them (ideally both the class and function(s). - If you keep track of these as you do them then it will be very easy to do this report – each of these is a task you did when you worked on the game.

Note: Do not write "I got some from the internet or I did some of the AI using code I found on stackoverflow– so why don't you go and work it out just where I got what and how I used it - try searching the world wide web and you'll work it out because I couldn't really be bothered in keeping track myself " – unless you want a low grade ☺

Assets I created

Make sure you don't leave out the detail – the assets you created should provide a precise list of elements that you created – you should also include pictures of the visual assets eg.

1. Created the sprite image of the main the Character (using photoshop)	
2. Created the character walk animation sprites	

Actually I didn't actually make these they come from

http://opengameart.org/content/cat-dog-free-sprites

Assets I sourced

Again this should be a careful and precise reference to the sources (stores, online sources) you used to get any assets for your prototype. If you edit an asset you got from somewhere (eg adjust colours, apply filters etc – you must still reference it here – just make sure you also include a brief description of the edits you made. Again visual assets should include a picture. Don't use vague references – be precise about where you got them on the website – don't make me search a website full of thousands of pictures to find it!

1. Enemy sprite "enemyFloating.png was downloaded from (the colours where edited to make them darker using photoshop)	

2. the games background music – "Violin Concerto in D major, Op. 61" was sourced from https://musopen.org/music/composer/ludwig-van-beethoven/

3. the interface button sounds were used "Interface Sound 40" from http://www.pacdv.com/sounds/interface_sounds.html