Games Engineering Coursework

Doors of Perception

Roderick Ewles 40330977

Git hub: https://github.com/40330977/Doors_of_Perception

1 Introduction

Overview

The inspirations for the game, in terms of mechanics, were Super Mario World and Risk of Rain. Both of these games rely on power ups or items that the player unlocks to change how they interact with the world. However the twist mechanically would be the inability to harm the enemies, forcing the player to try to chain their abilities to get around the enemies rather than killing them.

The inspirations for the game's theme and story are Aldous Huxley's "The Doors of Perception" and "Heaven and Hell", as well as Jeremy Narby's "The Cosmic Serpent". The idea being that the player has an altered state of perception which has lead to them being locked up as mad. However they aren't mad they just see things differently (i.e. perceiving themselves and medical staff as abstract shapes). While they are physically restrained they are being led on a journey to free their mind by two snake "spirit guides".

Scope & Content

The Game is a platformer with power ups that allows the player to change the attributes of their character. These powers must be used in varying combinations to complete the levels. The player will be able to change their speed and jump height as well as growing and shrinking. Aside from the snakes the game is supposed to be abstract and symbolic. The AI is simple but can be challenging. Essentially it attempts to follow the player wherever they go. The AI movement is such that the power ups can be used (with some challenge) to get around the enemies. There are several routes through the levels to encourage different strategies and power ups.

2 GDD Changes

The first major change to the gdd mechanically is collecting the power ups. Instead the player can use all four from the start. This was due to time and the fact that the game concept had to be changed. In addition to this all of the enemies in this implementation are hostile; some enemies were going to be added in the background and non-hostile so that their dialog could inform the player of the story but I ran out of time.

Another change was to simplify the UI so there is now no indication of which power ups are active. Instead an audio cue is given. This leads to less clutter on the screen, and it was found during

testing that the player knows what power ups are active. This is due to the fact that they are either visual or change the player's interaction. A health system was also scrapped as the games setting evolved from an Alice in Wonderland inspired game to a more abstract representation of similar themes. The player is being caught when they bump into enemies was now going to represent the main characters mental journey being disrupted by the mental health staff who don't understand him. Also rather than wonder land the platforms are now supposed to represent the main character's fractured stream of consciousness.

3 Software Design (State & Sequence Models at the end of the report)

Player Functionality

The Player functions were developed first. The increased speed and low gravity are implemented in the player physics component. So long as the relevant key or controller button is held down and movement is attempted the new interaction will occur. If a power up unlock system an external Boolean variable would be used to toggle them active or not.

Shrink and grow were more challenging. Re-scaling the shape was ok; changing the size of the player's collider was more challenging. Currently the code for this is in the level, if I had more time I would have tried to make this a separate component that can be added to the player.

Enemy AI

The enemy AI is simple but it can become quite challenging as it hunts the player through the level. Essentially the y direction changes on a timer or when the move the enemy is trying to make is invalid. The x direction always points towards the player. If the enemy falls off the map it is spawned in at the top of the map with the same x coordinate as the player. This has the effect of keeping the player under pressure. This AI is difficult to move around using the power ups but, with practice, it is possible. It is also possible to use the layout of a level to "trap" the AI.
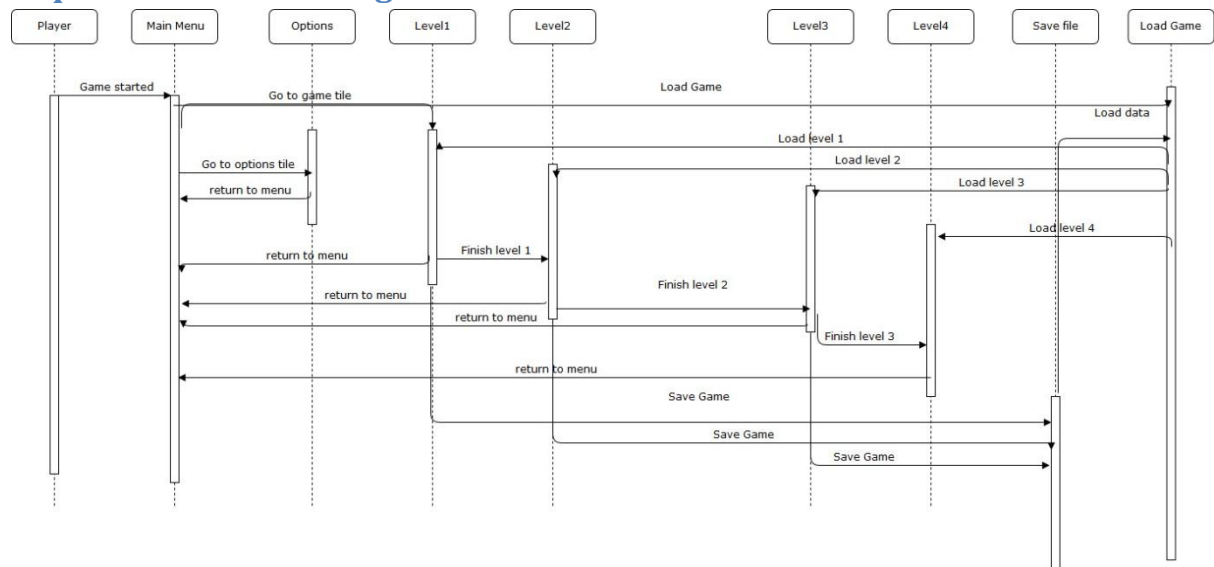
Save Game, Controller & Key Binding

Save game functionality works by writing out to a .txt file. This is set either at new game, setting it to level 1 to overwrite previous saves. At the end of each level the save file is updated before the scene is changed.

There is a controller class that stores the controller commands. It has functions for each button and an update function that allows input to be taken. Currently the controller controls
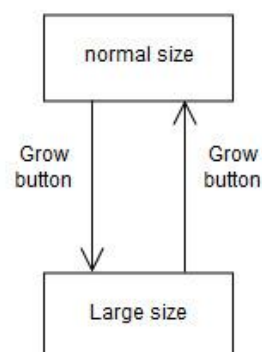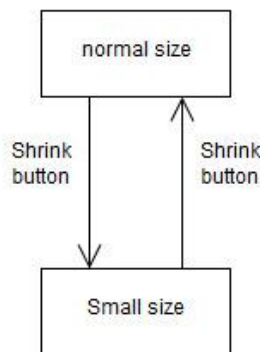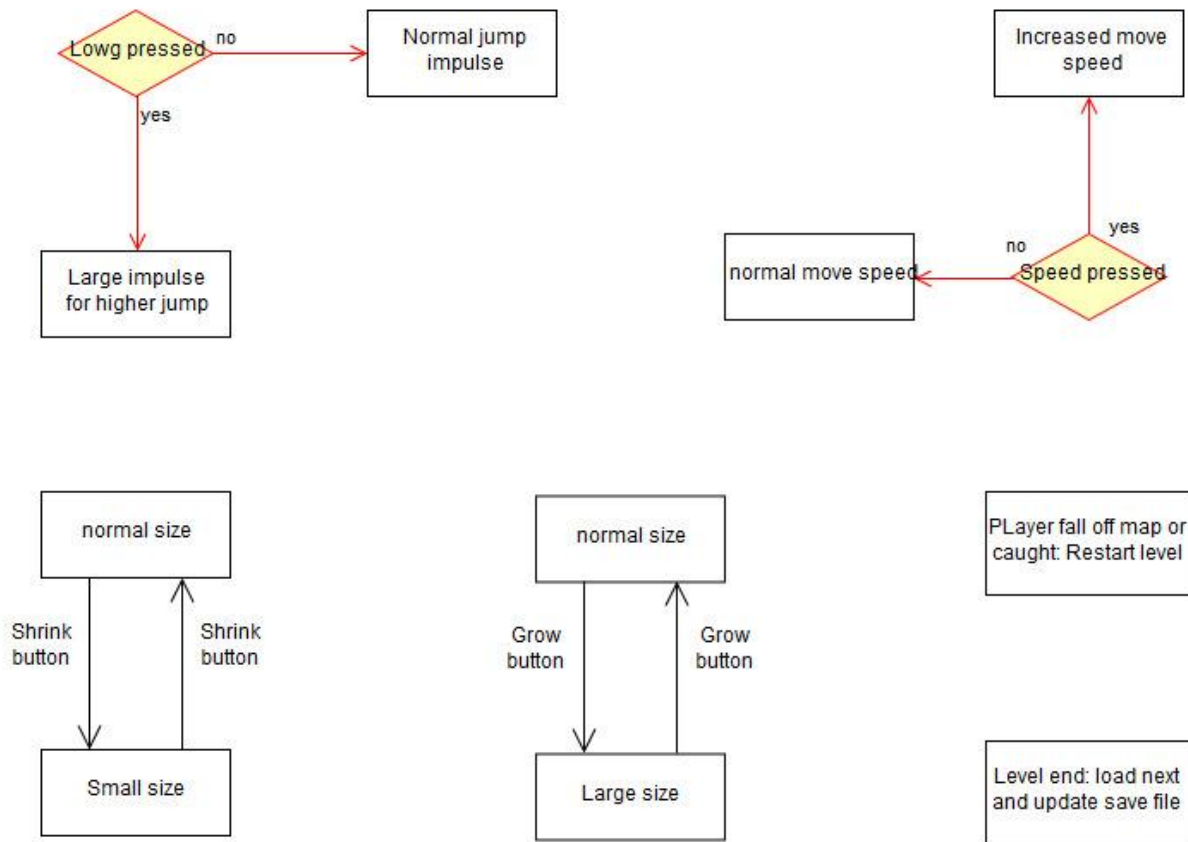
cannot be remapped as the triggers are used for the shrink and grow commands which act on a scale. I think that with time this could definitely be achieved.

Key binding is a little cumbersome but it does function. This is done in the kremap screen which is triggered by a tile to the left of the instructions in the options scene. To remap a key enter and the desired key to remap must be pressed at the same time. Following this the new key must be pressed. Currently kremap only allows one key to be remapped per entry to the scene.
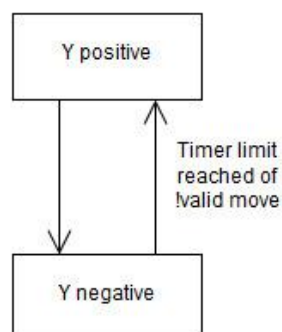
Sequence and State Diagrams

Player States

Lowg pressed → no → Normal jump impulse

Lowg pressed → yes → Large impulse for higher jump

Increased move speed

Speed pressed → yes → Increased move speed

Speed pressed → no → normal move speed

normal size → Shrink button → Small size

Small size → Shrink button → normal size

normal size → Grow button → Large size

Large size → Grow button → normal size

PLayer fall off map or caught: Restart level

Level end: load next and update save file

AI

Y positive → Y negative

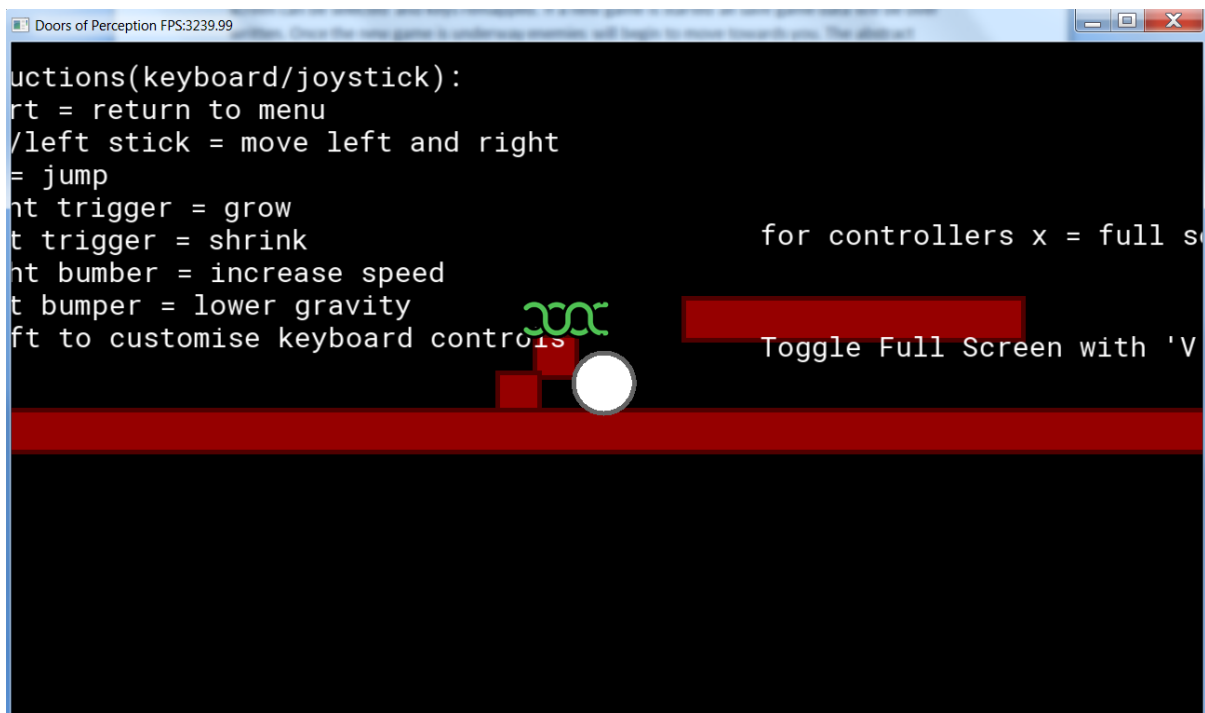Y negative → Timer limit reached of !valid move → Y positive

X direction: Always move towards Player

If fall off map:
Y = 5
X = player X

4 Implementation

The game has a start screen. At this point a lot of loading is done. Following this the player can take control of the player character in the main menu. The main menu leads to options, new game or load game. The two snakes give instructions as to where to go. In the options screen full screen can be selected and keys remapped. If a new game is started all save game data will be over written. Once the new game is underway enemies will begin to move towards you. The abstract shapes "attacking" you are supposed to be the mental health workers attempts at breaking your train of thought. The snakes are "spirit guides" like those suggested in book "The Cosmic Serpent". The circle represents the player character and the platforms in red represent the player characters stream of consciousness. At the final level there are no broken platforms, this is supposed to represent the player having attained a higher state of consciousness despite their physical imprisonment.

Doors of Perception FPS:3239.99

uctions(keyboard/joystick):
rt = return to menu
/left stick = move left and right
= jump
nt trigger = grow
t trigger = shrink
nt bumber = increase speed
t bumper = lower gravity
ft to customise keyboard controls

for controllers x = full s

Toggle Full Screen with 'V

Doors of Perception FPS:1811.39

< Your body may be trapped but
you can unlock your mind!

5 Evaluation

1 Comparison against original concept

Compared to the initial concept there have been considerable changes. This is in part due to the fact that a few weeks into the project I became a team of one. It was also due to the idea of the game evolving as the project continued. The basic mechanics behind the game play however, are roughly the same as the initial pitch.

2 Comparison against other games / inspiration

While mechanically slightly different, the game is a platformer and shares many features with both the inspirations and other platformers. In comparison to the inspirations, there is quite a disparity in quality between it and the likes of the Mario games. The levels are shorter, but challenging (for me anyway). This is mainly due to how the AI hunts the player. It may also be due to the fact that the main mechanic of the game is that the player can change how they interact with the world at will.

3 Overall Quality

The objectives of the game are reasonably apparent, and the snakes will prompt the user (although if more time were available more speech would have been implemented). The game is fairly useable, although, there are improvements I would like to make. The user experience seems to be positive, although again I would like to carry out more testing to ensure the correct experience is being conveyed to the player.

The power ups need to be balanced somehow. For grow and shrink a short cool down achieves this. The low gravity is balanced by the enemy AI re-spawn location. The increase upward impulse allows you to reach platforms you couldn't before. The trade off is that you are potentially closer to the enemy spawn point which can result in less response time for the player. The speed power up is balanced by the camera lag. If the player wants they can have super speed all of the time, the danger is quite quickly they will end up off of the screen. The snakes, which also lag behind the player but not as much, can almost be used to get around this but the likelihood is that the player will die if they speed too much.

4 Improvements

The first improvement would be more dialogue from the snakes and enemies. In both cases this could flesh out the story. The snakes could also give advice at certain points in levels or if it is detected that the player is dying in a certain way a lot. This leads to the next improvement which would be more and bigger levels.

Another would be more power ups like the ability to stop time for enemies, mirror (make lots of confusing copies of the player) or blink/ teleport. This ties in with the levels improvement as more power ups would allow an increasing complexity of challenges to be presented to the player. More sound track tracks would also be good to add variation.

Finally, ensuring that there are no bugs, allowing the player to remap all keys at once and controller remapping should be implemented.

5 Assets/ References

Assets

Original soundtrack was produced using FL Studio

Other sounds: https://www.bfxr.net/

https://sfbgames.com/chiptone/

Snake artwork: https://www.pngrepo.com/svg/8208/snake

References

Course lectures & tutorials

https://www.sfml-dev.org/

Stack Overflow

Gamasutra

Doors of Perception   Aldous Huxley

Heaven & Hell   Aldous Huxley

The Cosmic Serpent: DNA and the Origins of Knowledge   Jeremy Narby