

Games Programming

Assessment Item 1

Libraries Used

- SDL 2.0
- SDL_Mixer
- SDL_ttf
- SDL_Image

Controls

Navigate Up – W

Navigate Left – A

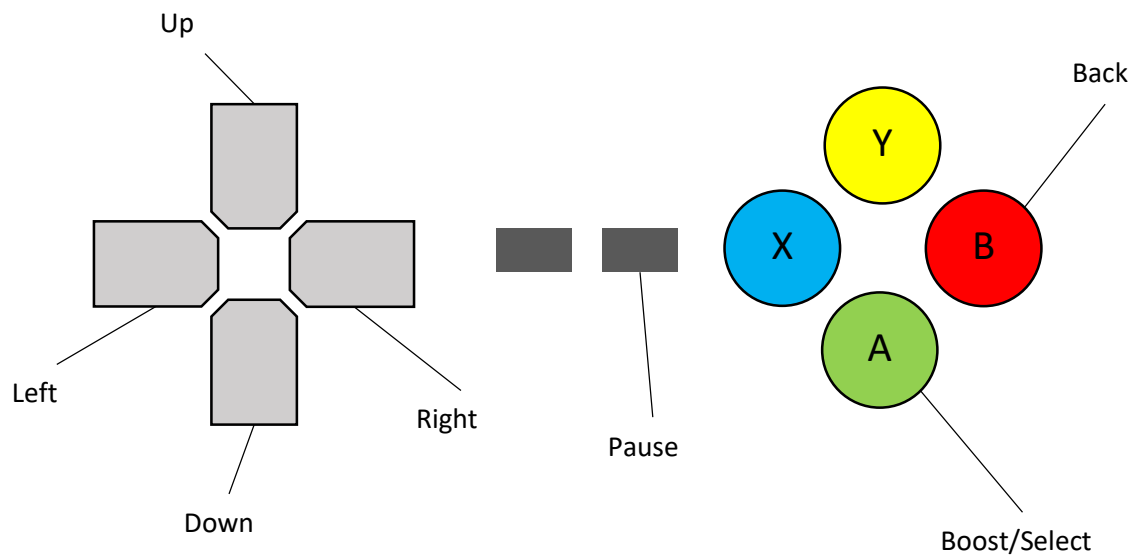
Navigate Down – S

Navigate Right – D

Select Option/Use Boost – Space/Return

Pause Game/Menu Back – Escape

Activate Debug Rendering – Tab



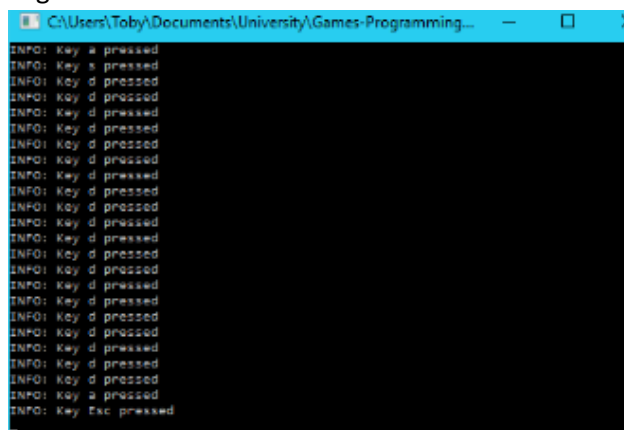
Features

- Menu System
 - Splash Screen.

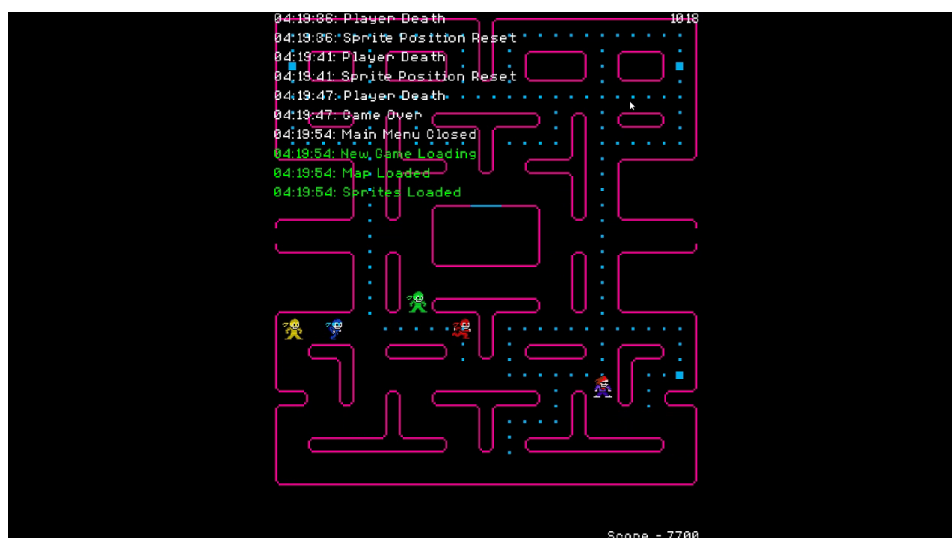


- Options Menu.
 - Music Volume adjustment.
 - Full screen Toggle.
 - Menu can be controlled with the keyboard or controller.
 - Elements are highlighted when they are selected.

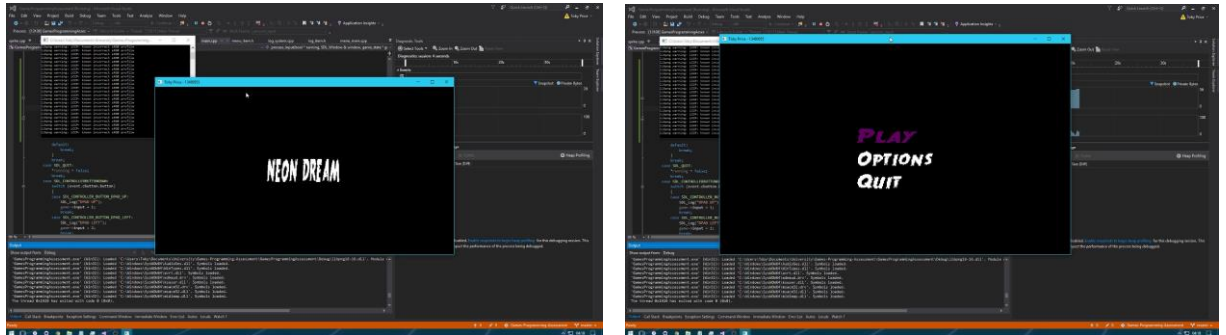
- Logging
 - Logs information to console.



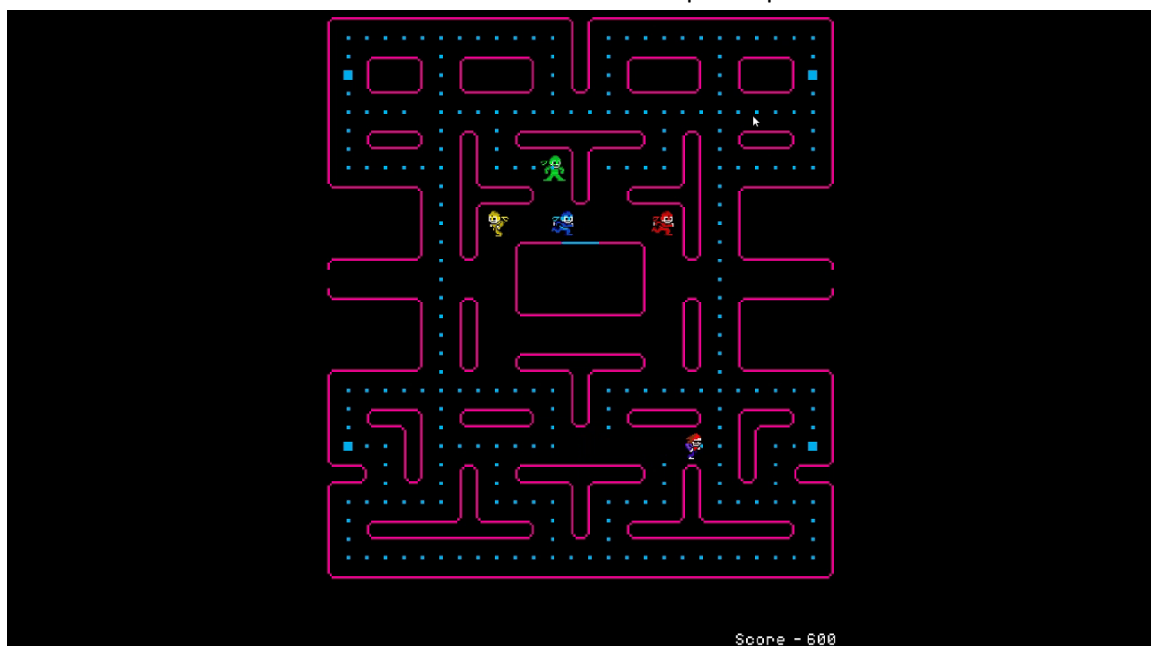
- Logs information to screen.
 - Score is shown on screen.
 - Log uses colours on screen to indicate priority.
 - Frame Counter is show in top right of screen.



- Window and Renderer
 - Window is created at the centre of the screen with half the width and half the height of the Desktop.
 - Window is resizable.
 - Window can be put in and out of Full screen through options menu.
 - Aspect Ratio remains constant regardless of the size of the window.



- Sprites
 - Multiple sprites are rendered to the screen.
 - Multiple sprites move, including the player controlled sprite.
 - Sprites have transparency.
 - Player and enemy sprites have animation sequences.
 - These animations are tied to each step the sprites take.



- Physics

- Player and enemy sprites can collide, causing the player to lose a life.
- Once the player has lost 3 lives the game ends.
- Player and pellet sprites can collide, causing the pellet to be deleted and the player score to increase by 100.
- Simulation pauses when the pause button is toggled



- Control

- The player can be controlled by keyboard.
- The player can be controlled using a Controller D-pad.
- The player increases its speed temporarily when a button is pressed.

- Audio

- There is background music during both the menu and gameplay.
- When the player collides with an enemy or a pellet a sound is played.
- Music volume can be controlled using the options menu.
- Background music is paused when the game is paused.

Development Process

During the development of this game I initially found myself trying to design an Object Oriented approach to the programming. To begin with this worked quite well as was helpful in allowing the code to be both easier to read and more flexible. However, towards the end of development where time became more of a factor, this Object Oriented approach began to slip and therefore there are lot more seemingly random variables strewn throughout the code. In order to prevent this from happening in the future more emphasis would need to be placed on designing a structure to the code before implementing it. Adding features as I went resulted in a lot of functions that are very specific to the task they complete, for example a menu item class and the log item are very similar in that they are both classes with a texture and text as their primary input. The two could have been very easily combined into a single "render text" class. This would make the code a lot more flexible in future situations where text would need to be rendered to screen.