IN2026 Games Technology Coursework

Joshua Lian 210042251

GitHub Link :  
https://github.com/Joshua153123/GamesTechCW.git

**Part I : Start Screen**

Upon launching the game, players are greeted with a start screen that displays the game's title along with a prompt to "Press any key to start!" This approach helps in pacing the start of the game, allowing players to begin playing when they feel prepared.

Initially, I considered incorporating the start screen functionality directly into an existing method called createGUI. However, upon further evaluation, I realized that this approach was not feasible. To ensure a more effective implementation, I decided to introduce a new method named displayStartScreen. This method is specifically designed to manage the start screen's display and operation, ensuring that it functions seamlessly within the game's structure.

The first thing I created is a flag startScreenActive in Asteroids header file. 

Then a method call displayStartScreen. The DisplayStartScreen() method in the Asteroids game sets up the start screen by adding a transparent border around the display and creating a centered label that instructs players to "Press any key to start." It then marks the startScreenActive flag as true to indicate the start screen is currently active.

A screen shot of a computer program

Description automatically generated

In the Start() method, I make a conditional call on the displayStartScreen method, only if the flag is true. I put the rest initialising code in the else condition, this make sure that the game would not initialise before I pressed any key. A screen shot of a computer program

Description automatically generated

In the OnKeyPressed, and OnSpecialKeyPressed method, I put the rest initialising game code here. I also turn the flag to false, and remove the “Press any key to start” label.

A computer screen with green and white text

Description automatically generated

Part2 : High Score Table  
  
The objective of the high score table feature is to maintain and display a list of the top five scores in a game, providing recognition and incentive for players to surpass previous records. Alternatives include expanding the list to include more scores, integrating online leaderboards for broader competition

Implementation:  
  
First I create a highscoreManager header and cpp file. A screenshot of a computer program

Description automatically generated

A computer screen shot of a program code

Description automatically generated

The HighScoreManager in the header and CPP files is designed to manage the top scores within a game. In the header file (HighScoreManager.h), the class is defined with methods to add a score, retrieve the high scores, and store a maximum number of scores (defaulting to 5). The CPP file (highScoreManager.cpp) implements these methods. The AddScore method intelligently adds a score only if it is high enough to be in the top scores and maintains the list in descending order. This setup ensures that at any point, the game can display the highest scores achieved, promoting competition and tracking player achievements efficiently.