Student name	Overall Score	Grade
Challinor, Jole (963402)	98	

Assessment Criteria

	Scale	Score
Design	20	20
Testing	10	10
Main menu screen	5	5
Player ship implementation	15	15
Shot implementation	15	15
Score implementation	10	10
Gameplay	10	8
End of game menu screen	5	5
Programming style / best practice / comments	10	10
Total	100	98

feedback

Overall: This is an excellent submission, demonstrating suitable programming skills for this module (that will be useful for other modules as well). The report is extensively detailed and evidenced. Also, all programming tasks have been performed as expected with a magnitude of creativity demonstrated in the results.

Design, develop and test an application to specification: All nouns and verbs have been identified in the specification and placed into a table to identify Classes and Methods. All classes are present in the UML diagram and match those found in the code. All attributes and methods are present in the UML diagram and match those found in the code. All aggregation and composition relationships have been clearly identified.

Testing: A testing strategy has been developed identifying how testing will take place, when testing will take place, where testing will take place, who will do the testing and what types of test will be performed. A testing table has been created that covers all of the minimum required features identified in the specification. All tests have been carried out and evidence has been provided where possible.

Game Functionality: The main menu contains a title or similar. The main menu contains a "Start Game" option that detects keyboard input and has full functionality. The main menu contains a "Quit Game" option that detects keyboard input and has full functionality. Ship is moving the right way (up and down). The enemies are moving the right way (from right to left). There is collision detection between ship and the enemies and working properly. Enemies are reaching the end of the screen and the proper action is taken. Shots are implemented and moving the right way. There is a good implenetation of a collision detection between shot and enemy and it is working properly.

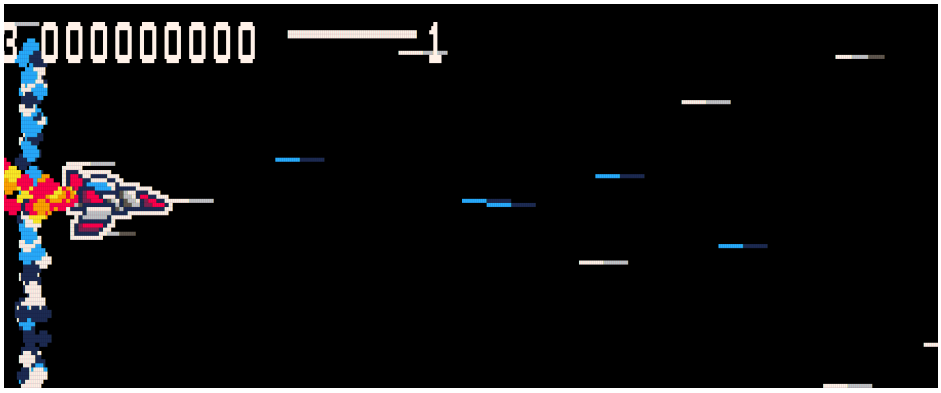
Score: A large score is displayed using the ASCIIRenderer as in the specification screenshot. A player will score 1 point if the ball exits the screen behind the opponent player.

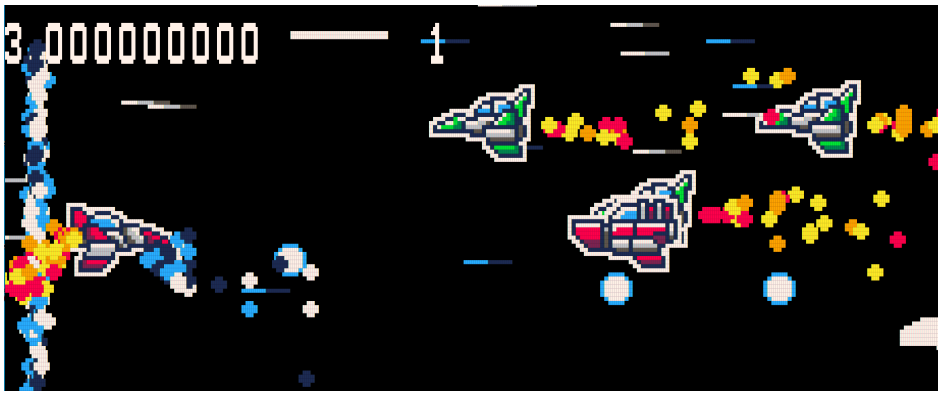
Gameplay: At the start of each round all of the game objects are re-set to their initial state. There speed of the game is fine, but the game due some high speed and unexpected behavior of the enemies. The enemies are implemented, but showing in pattern instead of randomly. The game is won by the first player to reach a given number of points.

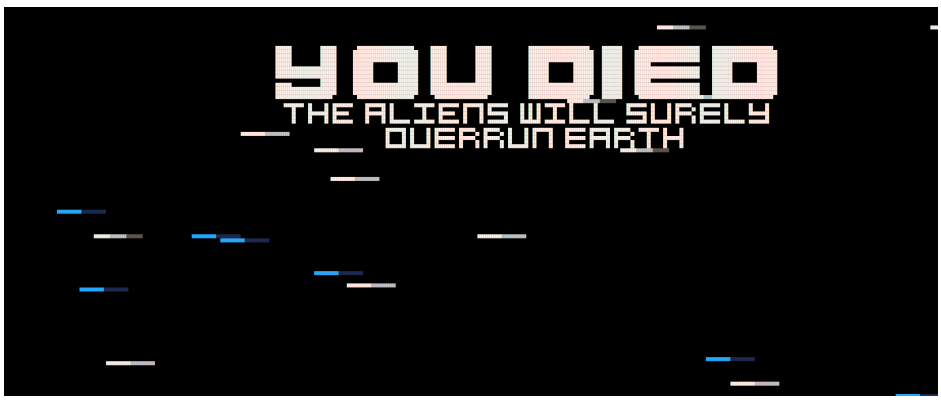
End of Game: The win screen clearly displays which player has won all of the time. The win screen allows the player to return to the main menu.

Programming Style / best practices / comments: A good programming style has been used consistently throughout the project. Good programming best practices have been used consistently throughout the project. Excellent use of comments throughout the project.


SHOOTY
SHOOTY
WAR
5
START
QUIT


3 0000000000
1


3 0000000000
1


YOU DIED
THE ALIENS WILL SURELY
OVERRUN EARTH