

# CE218 Assignment

---

*By George Cording*

## Instructions

In order to run the game the user must simply run the Game.class in the core package, In the game the user has several controls Up, left and right accelerate, turn left and right respectively. The Space key is the shoot key; this can be tapped or held down. This game features five different kinds of player ships and pressing the numbers 1 – 5 will change to the respective ship. Initially the player is ship 1 a light ship with relatively average firing speed.

Pressing the R key will restart the game. Due to the design of the game as it is Graphics weren't needed however pressing I will alternate between normal mode and Image mode, where the background and, when the game is restarted, initial asteroids are Images. Due to these Images not looking well next to the rest of the game they were left out of the normal version and are mostly in as proof of the ability to load images.

Pressing C and restarting, either by changing ships or by pressing the R key, will activate chaos mode in which bullets will break out of asteroids when they die. This is both fun to watch and shows how much the game can render.

The objective of the game is as standard asteroids with the added threat of enemy ships. There are three types of enemy ships, ones that remain stationary and shoot at the player from a distance, ones that actively seek out the player and ones that will ignore the player until the player gets close to them. There are three types of items in the game that must be shot by the player to be activated, a green one provides bonus points, blue is an extra life and white is temporary invincibility.

## Design choices

The software used in the game is fairly standard, in order to keep the game simple little outside of java was used. In terms of game design, the main part of this game is that there are different types of ships, both player and enemy ships. These function differently both from each other and depending on what lives they have. The other main feature of this game is that all ships have a size category that determines their performance. Light, Medium and Large. These ships change in performance depending on the number of lives the player has. Initially at 5 lives the Light ship will have the best performance, with the highest speed and turn rate however as the more lives are lost the light ship gains less performance than the medium and heavy ship and is eventually out classed by them with the Heavy ship having the best performance at 0 lives. The art style of the game is one that I initially stumbled upon, not liking how the fill polygon method looked I noticed a draw polygon method which when contrast with a black background looked nice and allowed for easy manipulation of the ship sizes as they lost lives.

## Tuned parameters

The game has many stats to calculate. With a changing performance being a large part of the game many variables had to be calculated in relation to other variables. For example the turn rate and acceleration:

Lives	Light Acceleration	Medium Acceleration	Heavy Acceleration	Light Turn Rate	Medium Turn Rate	Heavy Turn Rate
5	1.00	0.625	0.5	2	1.65	0.8
4	1.05	0.775	0.7	2.1	1.85	1.2
3	1.10	0.925	0.9	2.2	2.05	1.6
2	1.15	1.075	1.1	2.3	2.25	2.0
1	1.20	1.225	1.3	2.4	2.45	2.4
0	1.25	1.375	1.5	2.5	2.65	3

The turn rate is also multiplied by PI. This table shows the numbers calculated for the difference in ship stats over the number of lives, If a ship has more than 5 lives it is considered to have 5 lives. This is calculated as an equation for each with the exception of a Heavy ship at 0 lives getting an extra 0.2PI for turn rate, this is to give it the best turn rate at no lives whilst still having the worst at one.

Other things to take into account are the shot speeds of the 5 player ships, the first ship (Named Fish Tail in the class name) is a light ship with a shot delay of 20, this is an average number for the game as it is relatively fast but still requires some aiming. The second ship (Named Archie) is a heavy ship focused around a shot delay of 0.01, giving it a shot speed so fast it is almost a beam. The one drawback is that it has a tiny shot distance. Whilst a normal bullet has a distance relative to the screen height, so as to ensure that it will loop round and potentially hit the player if they remain stationary, Archie's shots are little more than the length of its ship.

The Third ship (Named Bessi) is another Heavy ship has a more sensible shot delay of 10 meaning that although it is hard to control at high lives it can shoot rapidly with a normal range. The Fourth ship (Named Buster) is a Medium ship with a shot delay of 20, this being a reasonable amount. Finally Krampt the fifth ship has the slowest shot delay of 200, It makes up for this by producing a shotgun blast as it shoots, turning itself in the process.

## Appraisal

I feel that the project was an enjoyable experience, The look of the game is nice however a bit more game play testing would help me tweak some variables and make the game more comfortable. I believe I have routed out most of the bugs and haven't found any game breaking bugs in the final version. The initial design of the game was to have easily addable ships, with little required to make new player ships and, with a bit of effort for the AI different enemy ships can be added as well.

At a point in the development I had planned for a menu system that would allow the player to alter things such as screen resolution, the active player ship or the different modes, This was scrapped however due to the work arounds in the form of keys being easier to implement. I am strangely proud of the look of the game, not being very good at art I was not expecting to make a good looking

game. However due to the ability to construct visual objects using matrices I was able to make simple artwork look good with a somewhat retro theme to it.