COMP20050 - Group 47 - HLA

Student Numbers: 22441636, 22450456, 22715709

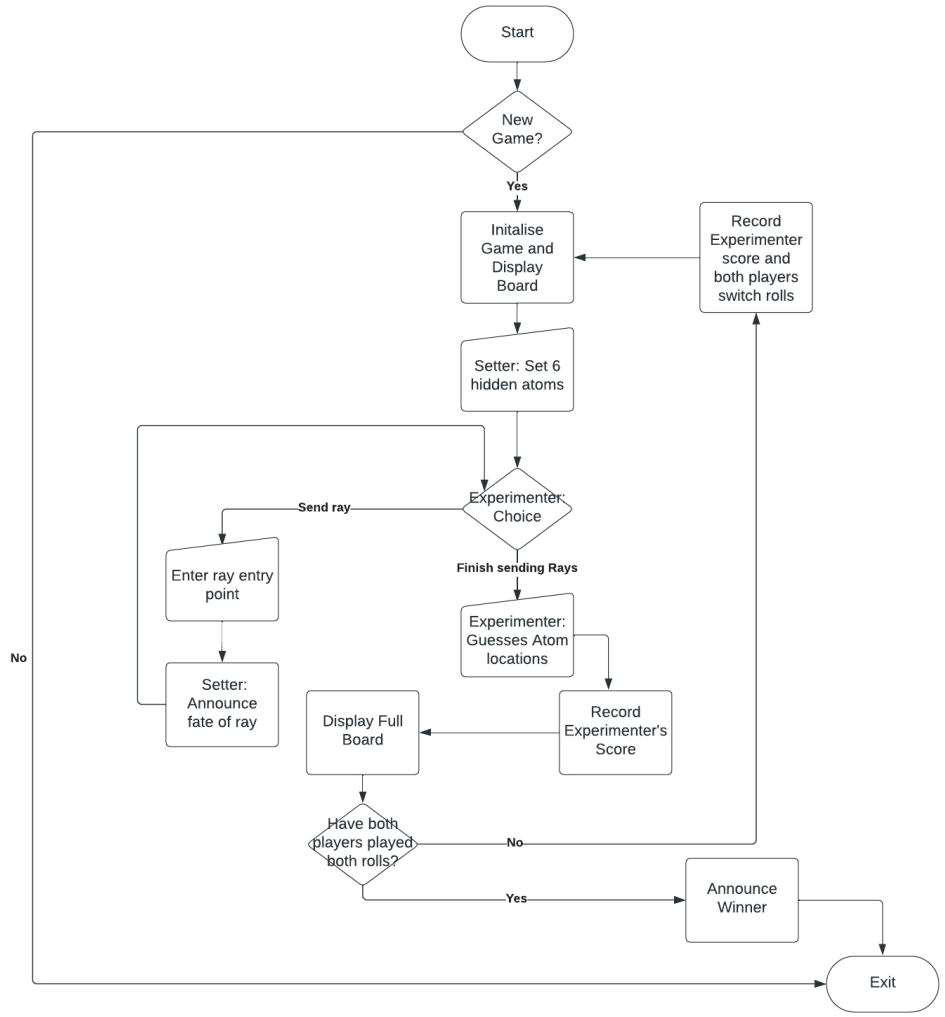
Game Requirements:

The game will consist of two players who take turns playing the role of 'setter' and 'experimenter'. The setter should be able to place 6 'atoms' onto a hexagonal grid in which the experimenter will send rays into to try and locate these atoms. The experimenter shall notify whether the ray hit an atom or whether it has exited the grid by placing a marker as to where 1)it entered and 2)where it exited (both of the same colour). Roll will flip once the first experimenter has sent the number of rays they wish and have guessed where they believe the atoms to be. At the end of the second game, the system will notify a winner based on a points system that tracks number of rays sent and also incorrectly guessed atoms.

Software Architecture:

The software architecture we will employ is the MVC (model, view, controller) idea as it will allow us to create a software system which captures changes in the model, displays those changes to the user and then allows the user to make more changes with the controller. In our case, the system will model the board in a data structure and store all necessary logic, display the board to the user and allow the user to input changes which are then fed back to our model.

Software System Flow Chart:



MVC Architecture:

Model:

Our model will handle all representation of states and also logic which can be carried out through various methods. The model will also have various key classes which will be used to represent items in the context of the Black Box game such as the board itself, atom, circle of influence, ray, raymarker. The main board class will be central and it will use all other classes to represent the game itself.

View:

Our view will be constantly updated to reflect changes in the model. Through our view component, we will be able to represent various points of the game, from showing an initialised board, updates made to the board and finally an end score.

Controller:

The controller will allow the user to input changes which will then be fed back to the model which will use the changes to update the current state of the game. The controller component will also be able to authenticate user input and ensure no incorrect inputs.