# **Iota Report – Falvey Group**

Iota is card game where a finite number of unique cards are placed on a board until the number of cards in the deck is exhausted. Because each card in the deck is unique, there are several instances where only one card can fill a certain position. This is particularly noticeable as the game continues and more cards are placed on the board.

## **Strategy**

To win in Iota you must play smart, taking advantage of some intrinsic conditions forced in each game is paramount to being able to beat your friends/family/rivals on the board. Understanding and being able to identify these conditions will be key to developing a strategy. Some key situations we’ve identified that can cause trouble for players are:

* A line is close to being completed but surrounding cards block you from placing a card down to complete it.
* There are multiple cards you can place but none of them complete a set (and receive double points).
* The card(s) you place might create a series of opportunities for your rival.

From this, it’s important to understand that winning the game requires you to not only play good hands that earn a high score, but also to try and ensure the player you’re competing against can’t earn more points than you on their turn. In a nutshell, you always need to be at least one step ahead of your opponent, knowing when to play a card and when not to.

## **Processes**

To implement some of these conditions into our strategy of the program will require some computer-generated foresight, a comprehensive data structure that that can be queried easily on the state of the board as it changes and/or as the program decides on which cards to play.

To do this, we can take advantage of brute force methods. Iota is a game of finite cards so looking at every possible scenario on the board and assessing each possible move on a predetermined set of criteria is feasible (as long as we don’t look too many steps ahead).

**Current state of our strategy**

Our system currently assesses all available positions on a board (free spots next to a card) and checks to see if any cards in our hand can be placed. If a card can be placed, we jump to that position and see if any other cards can be played with that one. This repeats until no more cards can be played.

Every single combination of cards is assessed and evaluated to see which will return the highest score. Before this is done though, we compare all the possible moves against themselves to see if they can be combined to achieve an even higher score.

**Moving forward**

Our strategy is fairly simple as it stands, currently assessing what it can do to achieve a high score without taking into account any moves the opponent might take. Some improvements we have built the program to accommodate in the future are:

* Reassessing the board after we have decided what cards we want to play so that we can check what opportunities that move opens for our opponent.
* Assessing the probability of an opponent possessing a certain card against the board and the deck.

These improvements can be used to refine our evaluation criteria on which hand we should play on each turn.