# Pockets Pack Levels (1-30) | Flow Free Wiki | FandomProject Summary

*Flow free is a single player mobile game developed by Big Duck, an American studio. Players are faced with a grid of different-colored pairs of dots. The goal is to connect the right dots while filling up every square on the grid. We connect enough dots so that the entire grid has a color and all cells are filled. While there are more complex versions of this game (with bridges and blockades) we wish to focus on the very basic iteration of flow free that was derived from the Numberlink logical puzzle.*

# Propositions

* girdConversion(fileName): This converts the text version of the flow free puzzle into a grid of coordinates that we can use to identify each cell and endpoints. It returns a 2D list of tuples with unique (i,j) values know as coordinates. In addition, it returns a dictionary of all colors in the puzzle and their respective endpoints.
* Connections(coordinate): Finds all possible neighbours of a cell by shifting its coordinates up, down, left or right by one.

# Constraints

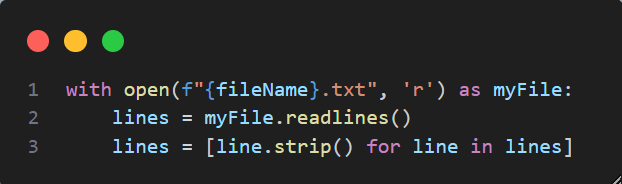
- “the cell comes from nowhere if it doesn’t have a colour”  
- “if the cell has a colour and comes from somewhere, then that source has the same colour”  
- “a cell can come from at most one other place”  
- “of all the neighbours that can come from a particular cell, at most one is true”

# Model Exploration

## Fixing the bug with improper grid conversion

*When we first implemented the grid size function, we ran into issues of improper sizes. It seemed as though in some cases, every row except the first one was longer or they were the same size in other grids.*

*This immediately initiated us to look the rows we were attempting to convert individually. As a result, we discovered that the newline character at each row but the last was being considered a cell in our gird. In order to solve this problem, we decided to use .strip() on each row.*

**

# Jape Proof Ideas

*List the ideas you have to build sequents & proofs that relate to your project.*

*Have not started, yet..*

# Requested Feedback

*Provide 2-3 questions you’d like the TA’s and other students to comment on.*

# First-Order Extension

*Describe how you might extend your model to a predicate logic setting, including how both the propositions and constraints would be updated.* ***There is no need to implement this extension!***

Have not started yet!

# Useful Notation

*Feel free to copy/paste the symbols here and remove this section before submitting.*